CODES, GRAPHS AND GRAPH BASED CODES

THÈSE N^{O} 3816 (2007)

PRÉSENTÉE LE 22 JUIN 2007

À LA FACULTÉ DES SCIENCES DE BASE
LABORATOIRE DE MATHÉMATIQUE ALGORITHMIQUE
PROGRAMME DOCTORAL EN MATHÉMATIQUES

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Andrew BROWN

M.Sc. in Mathematics, Oxford University, Oxford, Royaume-Uni
et de nationalité britannique

acceptée sur proposition du jury:

Prof. S. Morgenthaler, président du jury
Prof. M. A. Shokrollahi, directeur de thèse
Prof. D. Augot, rapporteur
Prof. E. Bayer Fluckiger, rapporteur
Prof. D. MacKay, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2007

Abstract

This work is concerned with codes, graphs and their links. Graph based codes have recently become very prominent in both information theory literature and practical applications. While most research has centered around their performance under iterative decoding, another line of study has focused on more combinatorial aspects such as their weight distribution. This is the angle we explore in the first part of this thesis, investigating the trade-off between rate and relative distance. More precisely, we show, using a probabilistic argument, that there exist graph-based codes approaching the asymptotic Gilbert-Varshamov bound, and that are encodable in time $O(n^{1+\epsilon})$ for any $\epsilon > 0$, where n is the block length.

The second part is concerned with more practical issues, more specifically the erasure channel. Although the codes mentioned above have been shown to perform very well in this setting, this nonetheless requires their lengths to be quite large. When short blocks are required, certain algebraic constructions become viable solutions. In particular Reed-Solomon (RS-) codes are used in a wide range of applications. However, there do not appear to be any practical uses of the more general Algebraic-Geometric (AG-) codes, despite numerous advantages. We explore in this work the use of very short AG-codes for transmissions over the erasure channel. We present their advantages over RS-codes in terms of the encoder/decoder running times, and evaluate the drawbacks by designing an efficient algorithm for computing the error probabilities of AG-codes. The work was done as part of an industrial collaboration with specific transmission problems in mind, and we include some practical data to illustrate the theoretical improvements.

Graphs and codes can be related in different ways, and a graph being a good *expander* often yields a code with certain desirable properties. In the third part we deal with graph products and their expansion properties. Just as the *derandomized squaring* operation essentially takes the square of a graph and removes some edges according to a second graph, we introduce the *derandomized tensoring* operation which removes edges from the tensor product of two graphs according to a third graph. We obtain a bound on the expansion of the product in terms of the expansions of the constituent graphs. We also apply the same ideas to a code product, leading to the *derandomized code concatenation* operation and its analysis.

Keywords: Repeat-Accumulate code, Gilbert-Varshamov bound, Reed-Solomon code, Algebraic-Geometric code, erasure channel, expander graph, derandomized squaring, derandomized tensoring, code concatenation.

Résumé

Ce travail concerne les codes, les graphes et leurs liens. Les constructions de codes à partir de graphes ont récemment pris beaucoup d'importance, tant dans les publications de théorie de l'information que dans les applications pratiques. Alors que la recherche s'est majoritairement centrée sur leur performance dans le décodage itératif, une autre direction s'est plutôt focalisée sur des aspects plus combinatoires, tels que leur distribution de poids. C'est cette approche que nous explorons dans la première partie de cette thèse, en étudiant le compromis entre rendement et distance minimale. Plus précisément, nous montrons, suite à un argument probabiliste, qu'il existe de tels codes approchant la borne asymptotique de Gilbert-Varshamov, et pour lesquels il existe un algorithme d'encodage avec temps de parcours $O(n^{1+\epsilon})$ pour tout $\epsilon > 0$, où n représente la longueur de bloc.

La seconde partie concerne des problèmes plus pratiques, plus spécifiquement le canal à effacement. Bien que les codes mentionnés ci-dessus aient de très bonnes performances dans ce cadre, leur longueur doit néanmoins être assez grande. Lorsque des blocs courts sont nécessaires, certaines constructions algébriques deviennent des solutions viables. En particulier, les codes de Reed-Solomon (RS) sont utilisés dans une grande panoplie d'applications. Il n'y a cependant apparemment aucune utilisation pratique des codes Algébriques-Géométriques (AG) pourtant plus généraux, et ceci en dépit de nombreux avantages. Nous explorons dans ce travail l'utilisation de codes AG très courts pour la transmission sur le canal à effacement. Nous présentons leurs avantages sur les codes RS, en termes des temps de parcours de l'encodeur et du décodeur, puis évaluons leurs inconvénients en concevant un algorithme efficace pour calculer les probabilités d'erreur des codes AG. Ce travail a été réalisé dans le cadre d'une collaboration industrielle, motivé par des problèmes de transmission spécifiques, et nous incluons également des données pratiques pour illustrer les gains théoriques.

Il existe plusieurs façons d'établir la relation entre les codes et les graphes, et un graphe qui est un bon *expanseur* mène souvent à un code avec certaines propriétés souhaitables. Dans la troisième partie nous nous intéressons aux produits de graphes et leurs propriétés d'expansion. Tout comme l'opération du *carré dérandomisé* prend le carré d'un graphe et lui retire des arêtes selon un deuxième graphe, nous introduisons le *produit tensoriel dérandomisé*, qui enlève des arêtes du produit tensoriel de deux graphes selon un troisième graphe. Nous obtenons une borne sur l'expansion du produit en fonction de l'expansion des graphes utilisés. Nous adaptons également ces idées à un produit de codes, menant ainsi à la *concaténation de codes dérandomisée* et son analyse.

Mots-clés: Code Repeat-Accumulate, borne de Gilbert-Varshamov, code de Reed-Solomon, code Algébrique-Géométrique, canal à effacement, graphe expanseur, carré dérandomisé, produit tensoriel dérandomisé, concaténation de code.

Acknowledgments

I would like to first of all thank my advisor Prof. Amin Shokrollahi for providing direction and inspiration throughout this thesis. Working with him has been not only a privilege, but also a rewarding experience on many levels. I am thankful for the opportunity to have learned so much from him.

I am also very grateful to Prof. Eva Bayer Fluckiger first of all for introducing me to Prof. Shokrollahi but also for her help and guidance when I arrived at EPFL.

It was an honor to work with Prof. Mike Luby throughout my time at Digital Fountain. The project I was involved in was both very gratifying and enjoyable. I am also grateful to Dr. Emina Soljanin for supervising me during my internship at Bell Labs.

I express my thanks to Prof. Eva Bayer Fluckiger, Prof. Daniel Augot, Prof. David MacKay, and Prof. Stephan Morgenthaler for accepting to be part of the Jury.

I thank Gérard Maze for his ideas on group algebras which were very helpful in Chapter 4. My thanks also go to Ivan Suarez for answering numerous algebraic questions while we shared an office. Likewise, I am grateful for the many discussions I had with Lorenz Minder on topics too abundant to mention. I am also thankful to Natascha Fontana for her help with all administrative issues, and to the other members of ALGO/LMA who made my time here more enjoyable: Christina, Payam, Giovanni, Zeno, Mahdi and Bertrand.

This work was funded in part by the Swiss National Science Foundation (under grant 200021-101495/2), which is greatly appreciated.

Boven alles zou ik graag Kim willen bedanken voor haar steun en toeverlaat.

Finally I would like to thank my parents firstly for proofreading the English, but mainly for their invaluable support throughout the years.

Contents

1 Introduction **1**

2 Coding Theory Background **4**

2.1 Introduction 4

2.2 Error Correcting Codes 4

2.3 The Gilbert-Varshamov Bound 6

3 Repeat-Accumulate Codes that Approach the Gilbert-Varshamov Bound **7**

3.1 Introduction 7

3.2 Background 8

3.2.1 Ensembles of Codes 8

3.2.2 Standard Bounds for the Binomial Function 8

3.3 Random Codes and the Gilbert-Varshamov Bound 9

3.4 RA Codes that Approach the GV Bound 11

3.4.1 Code Construction 11

3.4.2 Input/Output Weight Distribution 12

3.4.3 Proof Outline 17

3.4.4 Case 1: Large γ, Any α. 19

3.4.5 Case 2: Small α, Small γ. 21

3.4.6 Case 3: Small γ, $\alpha \geq \hat{A}$. 31

3.4.7 Case 4: Any α, $\hat{\Gamma_\ell} \leq \gamma \leq \hat{\Gamma_u}$. 34

3.4.8 Conclusion 44

4 Short Algebraic-Geometric Codes for Transmission over the Erasure Channel **46**

4.1 Introduction 46

4.2 The Erasure Channel 47

4.3 Algebraic-Geometric Codes . . . 48
4.3.1 Reed-Solomon Codes . . . 48
4.3.2 Algebraic-Geometric Codes . . . 49
4.4 The Specific Codes . . . 50
4.5 Computing the Error Probabilities . . . 51
4.5.1 Reduction to an Abelian Group Problem . . . 52
4.5.2 The Group Algebra $\mathbb{C}[G]$ . . . 54
4.5.3 Efficiently Computing the Polynomial $p_S(x)$ . . . 56
4.5.4 The Final Algorithm . . . 58
4.5.5 The Error Probabilities for our Specific Codes . . . 58
4.6 Interleaved Vector-Matrix Multiplication . . . 61
4.6.1 The Regular Representation . . . 61
4.6.2 Interleaving . . . 62
4.6.3 Encoding Time . . . 64
4.6.4 Decoding Time . . . 64
4.7 Implementations . . . 65
4.7.1 Encoding Bit Rates . . . 65
4.7.2 Decoding Bit Rates . . . 67
4.8 Conclusion . . . 70

5 Expander graphs **72**
5.1 Introduction . . . 72
5.2 Background . . . 73
5.3 Expander Graphs . . . 78
5.4 Random Walks . . . 79
5.5 Families of Expander Graphs . . . 81
5.6 Graph Products and Operations . . . 82
5.6.1 Edge Labelings . . . 82
5.6.2 Graph Squaring . . . 83
5.6.3 Graph Tensoring . . . 84
5.6.4 The Zig-Zag Product . . . 85
5.6.5 Derandomized Squaring . . . 87
5.6.6 Projection . . . 87

5.6.7 De-Projection . . . 88
5.7 The Spectrum of Biregular Bipartite Graphs . . . 90
5.7.1 Notation . . . 90
5.7.2 Transition Matrix . . . 91
5.7.3 Eigenvalues and Eigenvectors . . . 93
5.7.4 Random Walks . . . 93
5.7.5 The Second Eigenvalue . . . 94
5.7.6 Results We Will Need . . . 94
5.7.7 Convergence of Random Walks . . . 96
5.7.8 The Expander Mixing Lemma . . . 99

6 Derandomization Through Expander Graphs 102
6.1 Introduction . . . 102
6.2 Background . . . 103
6.3 Derandomized Squaring . . . 104
6.3.1 Introduction . . . 104
6.3.2 $A \textcircled{S} C$ as a Projection . . . 105
6.3.3 Bounding the Second Eigenvalue . . . 107
6.4 Derandomized Tensoring . . . 111
6.4.1 The Product . . . 111
6.4.2 Notation . . . 113
6.4.3 Definitions . . . 113
6.4.4 Proof Outline . . . 119
6.5 Derandomized Code Concatenation . . . 121
6.5.1 Introduction . . . 121
6.5.2 Definitions . . . 121
6.5.3 The Rate of $\mathcal{C}_1 \diamondsuit_H \mathcal{C}_2$ . . . 122
6.5.4 The Relative Distance of $\mathcal{C}_1 \diamondsuit_H \mathcal{C}_2$ . . . 123

A The Extension of the Binomial Function 128

B Proof of Theorem 6.11 137

C Proofs 159

Chapter 1

Introduction

The aim of coding theory is to provide methods of transmitting information in a reliable way over unreliable communication channels. Data sent through these channels may get *corrupted*, and the role of coding theory is to pre-process the sent data in such a way that it can be recovered from the corrupted data received. The pre-processing is referred to as *encoding*, while the recovery is referred to as *decoding*.

Encoding involves adding redundant information to the message before it is sent. This means that more information must be transmitted than would be on a reliable channel. How much redundancy is needed depends on how "bad" the channel is, i.e., how much corruption it adds. This leads to the natural question of what is the smallest amount of redundancy we can get away with for a given channel. The answer was given in Shannon's 1948 paper "A mathematical theory of communication", which laid down the basis for all digital communication. However, although his proof guarantees the existence of coding schemes that achieve the limits given in the paper, it gives no clue as to how such codes can be constructed.

It has henceforth been a major aim of coding theory to construct codes whose structural properties ensure reliable transmission, using as little redundancy as possible. We will consider only *block codes*, in which data is divided into pieces which are processed independently. The *length* of a code describes how much data is sent in each block. The *minimum distance* of a code can be important in assessing its error correction ability, in the sense that it being large guarantees a minimum adeptness to correct errors. The *rate* measures how much real information a block contains. These last two parameters pull against each other (improving one tends to worsen the other), and it is a fundamental problem in coding theory to find the best trade-off between the two. Another important issue to consider is the *complexity* of the code, referring to the running times of the encoding and decoding algorithms. Even when codes are studied only as combinatorial objects it is an interesting property to possess efficient algorithms, and it becomes essential in the context of data transmission.

Different tools have been developed to construct such codes. One major tool is *algebraic*, whereby known results from often rather abstract fields have been applied to obtain codes that can be proved to meet certain requirements. Although mathematically pleasing, there are aspects of these more traditional codes that can be improved upon. Graph theory is another such tool, whereby graphs with certain desirable properties can lead to codes with very effective decoding algorithms that work particularly well on common transmission channels.

Low Density Parity Check (LDPC-) codes are graph-based constructions that have attracted a lot of attention in recent years, due to their impressive performance under iterative decoding. Although first invented in 1963

by Gallager, they were later independently rediscovered in different flavors by Tanner [83], MacKay [48], Luby et al. [47]. They were shown to contain sequences that approach the capacity of a given symmetric channel, with very fast encoding and decoding algorithms. A different research direction has been the study of more combinatorial properties such as the weight distribution of these codes, mostly to obtain bounds on their performance under Maximum Likelihood decoding. This is the aspect we will consider in the first part of this work. More precisely we will construct in **Chapter 3** ensembles of graph-based codes that approach the Gilbert-Varshamov (GV-) bound with high probability, and that can be encoded in near linear time (essentially $O(n^{1+\epsilon})$ for any $\epsilon > 0$).

The second part of our work involves more practical applications. While the graph-based codes like those mentioned above do indeed have excellent performance, this is conditioned on their lengths being reasonably large. There are however applications requiring very short blocks for which *algebraic* codes have distinct advantages. The most ubiquitous are Reed-Solomon (RS-) codes, which are widely used in diverse applications. On the other hand, practical uses of the more general Algebraic-Geometric (AG-) codes are almost nonexistent. This is despite the fact that AG-codes have remarkable properties in that they enable the construction of codes with excellent rate/distance trade-off (in some cases beating the asymptotic Gilbert-Varshamov bound).

RS-codes have the drawback that their length is bounded by the size of the field on which they are constructed. This means first of all RS-codes cannot be studied asymptotically, but even for finite lengths, long codes require large fields. AG-codes do not have this restriction, an advantage that can be interpreted in two different ways. The most straightforward is that for a given field size one can construct longer codes, so that bigger pieces of data can be protected in each block. On the other hand, for a given n, an AG-code of length n will require a smaller field than an RS-code, which in turn means that the encoding and decoding algorithms can be made to run faster. This second interpretation becomes very relevant for applications that require short blocks (i.e., anything that needs to be decoded in real time). Furthermore, this is exactly the situation in which these algebraic codes can still outperform graph-based codes.

We explore in **Chapter 4** the use of very short AG-codes for transmissions over the erasure channel. We present their advantages over RS-codes in terms of encoder/decoder running time, and also quantify their drawbacks by developing an efficient algorithm to compute the error probabilities of the short AG-codes we consider. The contents of this chapter were motivated by existing practical needs, and we use a specific transmission problem to obtain some data illustrating the theoretical speed-ups. The work was done in collaboration with the company Digital Fountain and the codes presented are being used in some of their commercial products. It is interesting that although AG-codes are best known for their asymptotic properties, it is for these very short lengths that they appear to offer the best prospects for practical exploitation.

The third part of our work deals with the topic of expander graphs. Graphs and codes can be related in different ways. With the LDPC codes mentioned above the link was provided by the *Tanner graph* of the code. A different relationship can be established by taking an $[n, k]$-code with generator matrix G, and looking at the Cayley graph of $\mathbb{F}_2^k$ with respect to the columns of G. In both cases, the graph being a good *expander* guarantees that the corresponding code will be good.

We will be concerned in the last two chapters with graph products and their expansion properties. Rozenman and Vadhan introduced a modified version of the graph squaring product called *derandomized squaring* [65]. This led to a graph of smaller degree, at the cost of slightly worse expansion. We extend these ideas to another graph product (the tensor product) and a code product (code concatenation). After introducing expander graphs and some useful tools in **Chapter 5**, we describe and analyze our products in **Chapter 6**. More precisely we obtain a bound on the expansion of the derandomized tensor product (measured by the second

eigenvalue), as a function of the second eigenvalues of the constituent graphs.

Chapter 2

Coding Theory Background

2.1 Introduction

In this chapter we review the basic notions of coding theory that will be used in subsequent chapters. We give the standard definitions from the area of block codes before presenting the Gilbert-Varshamov bound which features prominently in Chapter 3.

2.2 Error Correcting Codes

All the following material can be found in standard textbooks (for example [89][51][41]), and will therefore not be expanded upon.

Definition 2.1. We have:

• An (n, M) *block code* $\mathcal{C}$ over an alphabet Σ is a subset of Σ^n of size M. n is referred to as the *length* of the code. All our codes will be block-codes, and we refer to them simply as *codes*.

• An $[n, k]$ *linear code* $\mathcal{C}$ over a finite field $\mathbb{F}_q$ is a subspace of $\mathbb{F}_q^n$ of dimension k. n and k are respectively referred to as the *length* and *dimension* of the linear code.

In this work we will deal exclusively with linear codes, so we assume from now on that all codes are linear. An $[n, k]$-code over $\mathbb{F}_q$ can also be referred to as an $[n, k]_q$-code.

Definition 2.2. Let $\mathcal{C}$ be an $[n, k]_q$-code.

• The *rate* of $\mathcal{C}$ is defined as $R(\mathcal{C}) = \frac{k}{n}$.

• A matrix $G \in \mathbb{F}_q^{k\times n}$ whose rows form a basis of $\mathcal{C}$ is called a *generator matrix* for $\mathcal{C}$.

• A matrix $H \in \mathbb{F}_q^{(n-k)\times n}$ for which $\mathcal{C} = \text{rker}(H)$ is called a *parity check matrix* for $\mathcal{C}$ (rker(H) denotes the right kernel of H).

Notice that if G is a generator matrix and H a parity check matrix for $\mathcal{C}$ then

$$\mathcal{C} \;=\; \big\{Gu \;\big|\; u \in \mathbb{F}_q^k\big\} \;=\; \big\{c \in \mathbb{F}_q^n \;\big|\; Hc = 0\big\}.$$

Definition 2.3. Let $\mathcal{C}$ be an $[n,k]_q$-code.

• The *hamming weight* of a vector $x \in \mathbb{F}_q^n$ is the number of non-zero components in x:

$$\text{wgt}(x) = \Big|\big\{i \mid x_i \neq 0\big\}\Big|.$$

The hamming weight of a vector will simply be referred to as its *weight*.

• The *hamming distance* between two vectors $x, y \in \mathbb{F}_q^n$ is the number of components in which they differ:

$$d(x,y) = \text{wgt}(x-y).$$

• The *zero codeword* is the zero vector in $\mathbb{F}_q^n$. It is always an element of the code.

• The *ball* around $x \in \mathbb{F}_q^n$ of radius r is defined as

$$B_r(x) = \big\{y \in \mathbb{F}_q^n \;\big|\; d(x,y) \leq r\big\}.$$

• The *minimum distance* of $\mathcal{C}$ is defined as

$$d_{\min}(\mathcal{C}) = \min\big\{d(x,y) \mid x,y \in \mathcal{C},\ x \neq y)\big\}.$$

Since we are assuming $\mathcal{C}$ to be linear, $d_{\min}(\mathcal{C})$ is also equal to the smallest hamming weight of a non-zero codeword.

• The *relative distance* of $\mathcal{C}$ is defined as

$$\delta(\mathcal{C}) = \frac{d_{\min}(\mathcal{C})}{n}.$$

• The *weight distribution* of $\mathcal{C}$ is the histogram of the weights of all the codewords. More formally it consists of the integers $A_0, \ldots, A_n$ where A_i is the number of codewords of weight i.

An $[n,k]_q$-code of minimum distance d can also be referred to as an $[n,k,d]_q$-code.

Although codes are interesting combinatorial objects in themselves, to study them in the context of reliable data transmission it is important to consider the *encoding* and *decoding* procedures.

Definition 2.4. Let $\mathcal{C}$ be an $[n,k]_q$-code. An *encoding function* is an injective map

$$E : \mathbb{F}_q^k \hookrightarrow \mathbb{F}_q^n$$

with $\text{Im}(E) = \mathcal{C}$.

A *Family* of codes is a sequence of codes of increasing length. Because we often do not know beforehand the length of the code we will need, it will be convenient and elegant to construct families in which all codes have a set of desired properties. Furthermore, we will be interested in *asymptotic* properties of codes, which require us to work with families.

Definition 2.5. A *family of codes* over $\mathbb{F}_q$ is a sequence $\{\mathcal{C}_i\}_{i \in \mathbb{N}^*}$, where $\mathcal{C}_i$ is an $[n_i, k_i, d_i]_q$ code, and

$$\lim_{i \to \infty} n_i = \infty.$$

The *rate* R and *relative distance* δ of the family are defined as

$$R = \lim_{i \to \infty} \frac{k_i}{n_i} \qquad \text{and} \qquad \delta = \lim_{i \to \infty} \frac{d_i}{n_i},$$

if these limits exist (and are said to be undefined otherwise).

2.3 The Gilbert-Varshamov Bound

The rate and minimum distance are fundamental parameters of a code. In the context of data transmission it is desirable to have both large minimum distance and large rate. A big minimum distance often means that more corruption in the transmission can be overcome, whereas a code of larger rate will require less redundant bits and therefore less bandwidth in the transmission. However these two parameters pull against each other, in the sense that increasing one of them tends to decrease the other one. This leads to the natural question of finding the best possible trade-off between the two.

One of the fundamental problems of coding theory is to compute the following function:

$$A_q(n,d) = \max\left\{k \;\middle|\; \text{there exists an } [n,k,d]_q\text{-code}\right\}. \tag{2.1}$$

This is a difficult problem, and for each field size q, the values of $A_q(n,d)$ are known only for small n and d. There are however many upper and lower bounds on $A_q(n,d)$ (see for example chapter 5 of [89]).

Another major question in coding theory concerns the *asymptotic* version of this problem, namely determining for which pairs $R, \delta \in [0,1]$ there exist families of codes of rate R and relative distance δ. Formally we define an asymptotic version of (2.1)

$$\alpha_q(\delta) = \limsup_{n\to\infty} \frac{A_q(n, \lfloor n\delta \rfloor)}{n}, \tag{2.2}$$

and are concerned with evaluating this function. $\alpha_q(\delta)$ is not known for any values of δ other than 0 and 1, but again there are many upper and lower bounds. In particular, the *Gilbert-Varshamov bound* described below will be important to us.

We will need the following function:

Definition 2.6. The *q-ary entropy function* $h_q : [0, \frac{q-1}{q}] \to [0,1]$ defined as

$$h_q(x) = \begin{cases} 0 & \text{if } x = 0 \\ -x\log_q\left(\frac{x}{q-1}\right) - \left(1-x\right)\log_q\left(1-x\right) & \text{if } 0 < x \leq \frac{q-1}{q}. \end{cases}$$

Theorem 2.7. ***The asymptotic Gilbert-Varshamov bound.***
For any $\delta < \frac{q-1}{q}$, we have

$$\alpha_q(\delta) \geq 1 - h_q(\delta). \tag{2.3}$$

Proof: This is a standard result. See for example [89], Theorem 5.1.9. ■

Notice that this is equivalent to saying that given a field $\mathbb{F}_q$, for any $\delta < \frac{q-1}{q}$ and $R < 1 - h(\delta)$ there exists a family of codes with rate R and relative distance $\geq \delta$.

In the next chapter we will be interested in the bound (2.3) of Theorem 2.7 for the binary case ($q = 2$). Although its proof is very simple, it has been conjectured that this bound is tight for $q = 2$. Perhaps surprisingly, almost all families of binary codes approach this bound asymptotically. It is however an open problem to find *explicit* constructions that do so.

Chapter 3

Repeat-Accumulate Codes that Approach the Gilbert-Varshamov Bound

3.1 Introduction

Graph based codes have attracted a lot of attention in recent years. For the most part, their renaissance has been due to the fact that they allow for fast encoding and decoding algorithms with which suitably designed codes approach the capacity of a given memoryless symmetric channel [48] [75] [35] [58] [21] [46] [62].

A different line of research has concentrated on the weight distribution of graph based codes (see, e.g., [42]). Mostly, these results are used to obtain bounds on the performance of the Maximum-Likelihood decoder for the codes in question. In this chapter, we study a special class of graph based codes and show that they contain sequences which approach the Gilbert-Varshamov (GV) bound. This bound says that for any $\delta < 1/2$ and any $R < 1 - h(\delta)$, there is a family of codes with relative distance $\geq \delta$ and rate R (where h is the binary entropy function).

The codes that we concentrate on are the Repeat-Accumulate Codes [22]. These have generator matrices of the form $G = M \cdot A$, where M is a matrix in which the columns are constructed independently at random to have approximately the same weight W, and A is the *accumulator* matrix, i.e., the upper triangular matrix having ones on and above the main diagonal. We will show, using a probabilistic argument, that there exist codes from this class that approach the Gilbert-Varshamov bound, if W is not too small. More precisely, if n and k denote the block length and the dimension of the code respectively, then we show that for any $y > 0$, if $W = \theta\left(k^y\right)$ then for any $\delta < nh^{-1}(1 - R)$ the probability that a code chosen from this ensemble has minimum distance $\leq n\delta$ converges to zero as n tends to infinity.

One of the applications of this result is that there are codes that approach the Gilbert-Varshamov bound and have fast encoding algorithms. This result in itself is not new, (see, e.g., Section 11.1 of [89]) but the derivation is interesting and the fact that the codes are Repeat-Accumulate codes with a simple combinatorial structure may suggest that there are asymptotically very good explicit Repeat-Accumulate codes.

After establishing some background we describe the construction and show that the corresponding codes approach the GV-bound with high probability. This is essentially done in two parts. We first obtain an expression for the probability that $\delta < nh^{-1}(1 - R)$, and then show that this expression converges to zero as n tends to infinity. The second part is unfortunately rather technical, but can be broken up into different cases

which we treat separately.

3.2 Background

3.2.1 Ensembles of Codes

This chapter deals with codes constructed using a *random component*, so we start by formalizing this concept. An *ensemble* $\mathbb{E}$ of codes is a finite set of codes with a probability distribution assigning non-zero probabilities to the codes. Choosing a code from $\mathbb{E}$ is equivalent to sampling from this distribution. We also suppose that all codes in a given ensemble have the same length (called the *length of the ensemble*). When we refer to the probability that an ensemble $\mathbb{E}$ has a certain property, we mean the probability that a code sampled from $\mathbb{E}$ has this property. So, for example,

$$\Pr\big[\mathbb{E} \text{ has rate } \geq R\big]$$

refers to the probability that a code sampled from $\mathbb{E}$ has rate at least R. Likewise if we say that the ensemble $\mathbb{E}$ has a certain property we mean that all codes in $\mathbb{E}$ have this property.

Recall that a *family of codes* is a sequence $\mathcal{C}_1, \mathcal{C}_2, \ldots$, where $\mathcal{C}_i$ is an $[n_i, k_i, d_i]_q$ code, and

$$\lim_{i\to\infty} n_i = \infty.$$

The *rate* R and *relative distance* δ of the family are defined as

$$R = \lim_{i\to\infty} \frac{k_i}{n_i}, \qquad \text{and} \qquad \delta = \lim_{i\to\infty} \frac{d_i}{n_i},$$

if these limits exist (and are undefined otherwise).

We can also have *families of ensembles* $\mathbb{E}_1, \mathbb{E}_2, \ldots$, where $\mathbb{E}_i$ has length n_i and

$$\lim_{i\to\infty} n_i = \infty.$$

The family is said to have a certain property P *with high probability* if

$$\lim_{i\to\infty} \Pr\Big[\mathbb{E}_i \text{ has property } P\Big] = 1.$$

3.2.2 Standard Bounds for the Binomial Function

We start by recalling the definition of the binary entropy function:

Definition 3.1. The *binary entropy function* $h : [0, \frac{1}{2}] \to [0, 1]$ is defined as

$$h(x) = \begin{cases} 0 & \text{if } x = 0 \\ -x \cdot \log_2(x) - (1-x) \cdot \log_2(1-x) & \text{otherwise.} \end{cases}$$

Unless specified otherwise, all logarithms in this chapter will have base 2, so $\log(x) = \log_2(x)$. The following standard results will be used throughout the chapter:

Theorem 3.2. *Let h denote the binary entropy function. For any $n \in \mathbb{N}$ and $\lambda \in \mathbb{R}$ with $0 \leq \lambda \leq \frac{1}{2}$, we have:*

$$\sum_{i=0}^{\lfloor \lambda n \rfloor} \binom{n}{i} \leq 2^{nh(\lambda)}, \tag{3.1}$$

and

$$\lim_{n \to \infty} \frac{\log\left(\sum_{i=0}^{\lambda n} \binom{n}{i}\right)}{n} = h(\lambda). \tag{3.2}$$

Proof: See [89], Theorem 1.4.5. ■

When $a, b \in \mathbb{R}_{\geq 0}$, we will use the following notational convention:

$$\sum_{i=a}^{b} f(i) = \sum_{i=\lceil a \rceil}^{\lfloor b \rfloor} f(i).$$

3.3 Random Codes and the Gilbert-Varshamov Bound

Uniformly random binary linear codes are produced by picking the entries of a $k \times n$ generator matrix uniformly at random. More formally, for any $n \in \mathbb{N}^*$ and $0 < R < 1$, we call $\mathcal{C}_{\text{rand}}(n, R)$ the ensemble of uniformly random binary linear codes of length n and of design rate R. The procedure of sampling from this ensemble can be described by the following algorithm:

Algorithm: UNIFORM-RANDOM-LINEAR(n, R)
1: Set $k \leftarrow \lceil nR \rceil$
2: Choose a matrix G uniformly at random from $\mathbb{F}_2^{k \times n}$.
3: Let $\mathcal{C} \leftarrow \{u \cdot G \mid u \in \mathbb{F}_2^k\}$ be the code whose generator matrix is G.
4: **return** $\mathcal{C}$.

Notice that this is equivalent to picking each entry of G independently and uniformly from $\mathbb{F}_2$. A code in $\mathcal{C}_{\text{rand}}(n, R)$ will have length n, but its rate will not necessarily be R (for example G could be the zero matrix with probability 2^{-kn}). R is referred to as the *design rate* of the ensemble.

When we speak of "random codes" without further specification we actually mean "uniformly random codes". We will sometimes abuse notation by referring to the family of ensembles

$$\left\{\mathcal{C}_{\text{rand}}(n, R)\right\}_{n \in \mathbb{N}^*}$$

simply as $\mathcal{C}_{\text{rand}}(n, R)$.

We recall the asymptotic Gilbert-Varshamov bound for binary codes:

Theorem 3.3. ***The asymptotic Gilbert-Varshamov (GV) bound.***
For any $\delta < \frac{1}{2}$ and $R < 1 - h(\delta)$, there exists a family of binary codes with rate R and relative distance $\geq \delta$.

Notice that this is not saying we can find families with (δ, R) *on* the $R = 1 - h(\delta)$ curve, but *arbitrarily close* to it. It turns out that random binary codes (the family of ensembles $\{\mathcal{C}_{\text{rand}}(n, R)\}_{n \in \mathbb{N}^*}$) approach this bound with high probability:

Theorem 3.4. *For any $\delta < \frac{1}{2}$ and $R < 1 - h(\delta)$, if $k = \lceil nR \rceil$ then $\mathcal{C}_{rand}(n, R)$ is an $[n, k, \geq n\delta]$-code with high probability.*

Proof: Let $\mathcal{C}$ be a code sampled from $\mathcal{C}_{\text{rand}}(n, R)$, and let d_{min} be the minimum distance of $\mathcal{C}$. We will show that the probability that there is a non-zero codeword in the closed ball $\text{B}(0, n\delta)$ converges to zero as n gets large. This will imply first of all that $d_{\text{min}} \geq n\delta$ (with high probability), and secondly that the kernel of the generator matrix G of $\mathcal{C}$ consists only of the zero vector, and therefore that G has full rank, which means that $\mathcal{C}$ has dimension k (with high probability). Let

$$\epsilon_1 = 1 - h(\delta) - R.$$

Since $R < 1 - h(\delta)$, we have $\epsilon_1 > 0$. Now the volume of $\text{B}(0, n\delta)$ is

$$\text{Vol}(n\delta, n) = \sum_{i=0}^{\lfloor n\delta \rfloor} \binom{n}{i}.$$

From (3.1) of Theorem 3.2, we have

$$\text{Vol}(n\delta, n) \leq 2^{n \cdot h\left(\frac{\lfloor n\delta \rfloor}{n}\right)} \leq 2^{n \cdot h(\delta)}.$$

For a fixed non-zero message vector $u \in \mathbb{F}_2^k$, the corresponding codeword $c = uG$ is uniformly distributed over $\mathbb{F}_2^n$. So the probability that c is in $\text{B}(0, n\delta)$ is

$$P = \Pr\Big[c \in \text{B}(0, n\delta)\Big] = \frac{\text{Vol}(n\delta, n)}{2^n} \leq 2^{n \cdot (h(\delta) - 1)}.$$

Recall that k was defined as $k = \lceil nR \rceil$. Now let $\epsilon_2 = k - nR$, so that $0 \leq \epsilon_2 < 1$. We have $k = nR + \epsilon_2$. By making n large enough we can ensure that $\frac{\epsilon_2}{n}$ is as small as we like. In particular there is an N for which

$$n \geq N \implies \frac{\epsilon_2}{n} \leq \frac{\epsilon_1}{2}. \tag{3.3}$$

Since there are 2^k message vectors, by the union bound we can deduce that if $n \geq N$ then

$$\begin{aligned}
\Pr\Big[\exists c \in \mathcal{C} : c \neq 0 \text{ and } c \in \mathrm{B}(0, n\delta)\Big] &\leq 2^k \cdot P \\
&\leq 2^k \cdot 2^{n\cdot(h(\delta)-1)} \\
&= 2^{nR+\epsilon_2} \cdot 2^{n\cdot(h(\delta)-1)} \\
&= 2^{n\cdot(R+h(\delta)+\epsilon_2/n-1)} \\
&\leq 2^{n\cdot(R+h(\delta)+\epsilon_1/2-1)} && \text{(using (3.3))} \\
&= 2^{-\frac{\epsilon_1}{2}\cdot n} && \text{(since } \epsilon_1 = 1-h(\delta)-R\text{).}
\end{aligned}$$

We can therefore deduce that with high probability, $B(0, n\delta)$ does not contain a non-zero codeword. ■

We see in this proof of Theorem 3.4 that $\mathcal{C}_{\text{rand}}(n, R)$ approaches the GV bound with a probability that converges to 1 *exponentially* fast as n tends to infinity.

3.4 RA Codes that Approach the GV Bound

3.4.1 Code Construction

Our idea is to construct a code in which the distances between successive columns of the $k \times n$ generator matrix G are approximately the same. We construct each column of G by taking the previous column, picking W components uniformly at random *with repetition* from $\{1, \ldots, k\}$, and each time flipping the corresponding bit. Notice that the distance between successive columns could be less than W if a component got picked more than once (though this happens with very low probability). Ensuring that the distance is exactly W would require the flipped components to be picked *without repetition*, which makes the analysis substantially more complicated.

We will show instead that picking them with repetition suffices to obtain families that approach the GV-bound. Indeed, asymptotically the probability of getting any repetitions converges to zero. We start by expressing this construction as a Repeat-Accumulate (RA) code.

The *accumulator matrix* is a square matrix with ones on and above the diagonal, and zeros everywhere else:

Definition 3.5. The $n \times n$ *accumulator matrix* A_n is defined as

$$\big(A_n\big)_{ij} = \begin{cases} 1 & \text{if } i \leq j \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

When the dimensions are clear from the context we will write A instead of A_n.

Definition 3.6. For any $n \in \mathbb{N}^*$, $0 < y < 1$ and $0 < R < 1$, we call $\mathcal{C}_{\text{RA}}(n, R, y)$ the ensemble whose sampling procedure is the following:

Algorithm: GOOD-RA(n, R, y)

Initialize a $k \times n$ matrix M to the all zero matrix
Set $k \leftarrow \lceil nR \rceil$
for $j = 1, \ldots, n$ **do**
  for $a = 1, \ldots, \lfloor k^y \rfloor$ **do**
    pick $i \in \{1, \ldots, k\}$ uniformly at random
    Set $M_{ij} \leftarrow M_{ij}$ XOR 1
  end for
end for
Let $G \leftarrow M \cdot A_n$
Let $\mathcal{C} \leftarrow \{u \cdot G \mid u \in \mathbb{F}_2^k\}$ be the code whose generator matrix is G.
return $\mathcal{C}$.

Informally, we construct a random matrix M as follows: each column is constructed independently by picking $\lfloor k^y \rfloor$ entries from $\{1, \ldots, k\}$ uniformly at random with repetition (and $k = \lceil nR \rceil$). Each component picked an even number of times is set to 0, each component picked an odd number of times is set to 1. This matrix M is then multiplied by the accumulator matrix to obtain the generator matrix of our code.

Note that for $i = 2, \ldots, n$, column i of M is the difference between columns $i-1$ and i of G, and so the weights of the columns of M represent the distances between successive columns of G.

The expected number of ones in M is at most $n \cdot k^y = O(n^{1+y})$ (assuming the rate k/n is constant). So multiplication by M can be done in sub-quadratic time. If $u \in \mathbb{F}_2^k$ is a message vector, the encoding process (i.e., computing the codeword $c = u \cdot G$) can be decomposed into two stages:

1. Compute $v = u \cdot M$. This requires $O(n^{1+y})$ operations.
2. Compute $c = v \cdot A$. This requires $O(n)$ operations.

So the whole encoding process is sub-quadratic $O(n^{1+y})$.

As above, we will abuse notation by referring to the family of ensembles

$$\left\{\mathcal{C}_{\text{RA}}(n, R, y)\right\}_{n \in \mathbb{N}^*}$$

simply as $\mathcal{C}_{\text{RA}}(n, R, y)$.

Our aim is to show that $\mathcal{C}_{\text{RA}}(n, R, y)$ approaches the asymptotic Gilbert-Varshamov bound. More formally we want to show that for any $\delta < \frac{1}{2}$ and $R < 1 - h(\delta)$, a code chosen from $\mathcal{C}_{\text{RA}}(n, R, y)$ will be an $[n, nR, \geq n\delta]$-code with high probability (with a probability that converges to 1 as n tends to infinity).

3.4.2 Input/Output Weight Distribution

Our goal in this section is to get an expression upper bounding the probability

$$\Pr\left[d_{\min}\left(\mathcal{C}_{\text{RA}}(n, R, y)\right) \leq n\delta\right] \tag{3.5}$$

as a function of n, R, y and δ (see Theorem 3.14). We will then use this in the next section to show that when $R < 1 - h(\delta)$ this probability will converge to zero as n tends to infinity. Our approach to obtaining this upper bound is to compute the *Input/Output weight distribution* of the generator matrix of $\mathcal{C}_{\text{RA}}(n, R, y)$.

Suppose we have values $n \in \mathbb{N}^*$, $0 < R < 1$ and $0 < y < 1$. Set $k = \lceil nR \rceil$. We consider the following experiment:

1. Sample a code (along with its $k \times n$ generator matrix $G = MA$) from $\mathcal{C}_{RA}(n, R, y)$.
2. Sample a message vector u uniformly at random from $\mathbb{F}_2^k$.
3. Compute $v = uM$.
4. Compute $c = vA$ (so $c = uMA = uG$ is the encoding of u).

To each $u \in \mathbb{F}_2^k$ there corresponds a distribution $\mathcal{D}_v$ on v. We now make two observations. Firstly, the distribution is the same for all u's of a given weight, i.e., if $\text{wgt}(u) = \text{wgt}(u')$ then $\mathcal{D}_u = \mathcal{D}_{u'}$. Secondly, for a fixed $u \in \mathbb{F}_2^k$ the probabilities are the same for two v's of a given weight (since each component of v is independent of the others), so if $\text{wgt}(v) = \text{wgt}(v')$ then $\Pr_{\mathcal{D}_u}(v) = \Pr_{\mathcal{D}_u}(v')$.

Definition 3.7. We define the $k \times n$ matrix $\overline{M}$ as follows:

$$\overline{M}_{w\ell} = \Pr\left[\text{wgt}(uM) = \ell \;\middle|\; \text{wgt}(u) = w\right]. \tag{3.6}$$

Notice that the probability in (3.6) involves two different sources of randomness: On the one hand the random construction of M (described in Definition 3.6), and on the other hand the choice of the message vector u (picked uniformly at random).

Definition 3.8. We define the $n \times n$ matrix $\overline{A}$ as follows:

$$\overline{A}_{\ell d} = \Pr\left[\text{wgt}(vA) = d \;\middle|\; \text{wgt}(v) = \ell\right]. \tag{3.7}$$

Because the matrix A is not random, the probability in (3.7) has a single source of randomness, namely the choice of v. We call $\overline{M}$ and $\overline{A}$ the *input/output weight distributions* (IOWD) of the matrices M and A.

Lemma 3.9. *Let $\mathcal{C}$ be a code sampled from $\mathcal{C}_{RA}(n, R, y)$, and let $k = \lceil nR \rceil$. Then*

$$\Pr\left[d_{\min}(\mathcal{C}) \leq n\delta\right] \leq \sum_{d=1}^{n\delta} \sum_{w=1}^{k} \binom{k}{w} \cdot \sum_{\ell=1}^{n} \overline{M}_{w\ell} \cdot \overline{A}_{\ell d}. \tag{3.8}$$

Proof: If we let G be the generator matrix of $\mathcal{C}$, then G is a random matrix whose IOWD is the $k \times n$ matrix $\overline{B}$ defined as

$$\overline{B}_{wd} = \Pr\left[\text{wgt}(uG) = d \;\middle|\; \text{wgt}(u) = w\right].$$

As in (3.6), there are two sources of randomness for this probability: the construction of G (sampling from the ensemble), and the uniform choice of u. For the rest of this proof we suppose that we have a vector u chosen uniformly at random from $\mathbb{F}_2^k$, and we let

$$v = uM, \qquad \text{and} \qquad c = vA, \tag{3.9}$$

so that $c = uG$ is the codeword obtained from u. We have:

$$\overline{B}_{wd} = \sum_{\ell=0}^{n} \Pr\left[\text{wgt}(v) = \ell \;\middle|\; \text{wgt}(u) = w\right] \cdot \Pr\left[\text{wgt}(c) = d \;\middle|\; \text{wgt}(v) = \ell\right],$$

which using Definitions 3.7 and 3.8 leads to

$$\overline{B}_{wd} = \sum_{\ell=0}^{n} \overline{M}_{w\ell} \cdot \overline{A}_{\ell d}.$$

Let W_d be the probability that a codeword picked uniformly at random has weight d. This is equal to the probability that a message vector u picked uniformly from $\mathbb{F}_2^k$ gets encoded to a codeword of weight d, which gives us

$$W_d \quad = \quad \sum_{w=0}^{k} \frac{\binom{k}{w}}{2^k} \cdot \overline{B}_{wd} \quad = \quad \frac{1}{2^k} \sum_{w=0}^{k} \binom{k}{w} \sum_{\ell=0}^{n} \overline{M}_{w\ell} \cdot \overline{A}_{\ell d}. \tag{3.10}$$

So since there are at most 2^k codewords in $\mathcal{C}$, by the union bound the probability that there exists a codeword of weight d is at most $2^k \cdot W_d$:

$$\Pr\left[\exists c \in \mathcal{C} : \text{wgt}(c) = d\right] \leq \sum_{w=0}^{k} \binom{k}{w} \sum_{\ell=0}^{n} \overline{M}_{w\ell} \cdot \overline{A}_{\ell d}. \tag{3.11}$$

Since

$$\Pr\left[d_{\min}(\mathcal{C}) \leq n\delta\right] \leq \sum_{d=1}^{n\delta} P\left[\exists c \in \mathcal{C} : \text{wgt}(c) = d\right], \tag{3.12}$$

we obtain

$$\Pr\left[d_{\min}(\mathcal{C}) \leq n\delta\right] \leq \sum_{d=1}^{n\delta} \sum_{w=0}^{k} \binom{k}{w} \cdot \sum_{\ell=0}^{n} \overline{M}_{w\ell} \cdot \overline{A}_{\ell d}. \tag{3.13}$$

Because all the terms are non-negative, this inequality still holds if we start the sums at $w = 1$ and $\ell = 1$:

$$\Pr\left[d_{\min}(\mathcal{C}) \leq n\delta\right] \leq \sum_{d=1}^{n\delta} \sum_{w=1}^{k} \binom{k}{w} \cdot \sum_{\ell=1}^{n} \overline{M}_{w\ell} \cdot \overline{A}_{\ell d}, \tag{3.14}$$

which is the required result. ■

So to get the bound on (3.5) we are looking for, we need expressions for $\overline{M}_{w\ell}$ and $\overline{A}_{\ell d}$. The IOWD of the accumulator matrix is given in [22] without proof, so we include a proof below.

Theorem 3.10.

$$\overline{A}_{\ell d} = \frac{\binom{n-d}{\lfloor \ell/2 \rfloor}\binom{d-1}{\lceil \ell/2 \rceil - 1}}{\binom{n}{\ell}}. \tag{3.15}$$

Proof: We would like to count how many vectors $v \in \mathbb{F}_2^n$ of weight ℓ have the property that $c = v \cdot A$ has weight d. Let $s_1 < \ldots < s_\ell \in \{1, \ldots, n\}$ be the ℓ indices such that $v_{s_i} = 1$. For convenience we also define $s_0 = 1$. For all $j = 0, \ldots, \ell - 1$ we define $S_j = \{s_j, \ldots, s_{j+1} - 1\}$, and $S_\ell = \{s_\ell, \ldots, n\}$. Notice that S_0 is

the only set that may be empty, all other sets will contain at least one element. $S_0, \ldots, S_\ell$ form a partition of $\{1, \ldots, n\}$. We call S_j an *even set* when j is even (including $j = 0$), and an *odd set* otherwise.

If ℓ is even then there are $\ell/2 + 1$ even sets, and $\ell/2$ odd sets. If ℓ is odd, there are $(\ell + 1)/2$ even sets, and $(\ell + 1)/2$ odd sets. So in both cases there are $\lfloor \ell/2 \rfloor + 1$ even sets, and $\lceil \ell/2 \rceil$ odd sets.

By looking closely at the accumulator matrix, we can see that:

$$c_i = \begin{cases} 0 & \text{if} \quad i \in S_j \ \text{ where } S_j \text{ is an even set} \\ 1 & \text{if} \quad i \in S_j \ \text{ where } S_j \text{ is an odd set.} \end{cases}$$

Observe that by deciding on the size of each set S_j we are uniquely determining the values $s_1, \ldots, s_\ell$, and therefore the vector v. So to count how many vectors v lead to c having weight d we need to count how many ways we can construct $S_0, \ldots, S_\ell$ such that the odd sets contain a total of d elements, and the even sets a total of $(n - d)$ elements.

Our problem is now reduced to one of balls and bins: we need to place d ones (balls) into $\lceil \ell/2 \rceil$ odd sets (bins), and $(n - d)$ zeros into $\lfloor \ell/2 \rfloor + 1$ even sets. We recall that in general for $a \geq b$ there are $\binom{a-1}{b-1}$ ways of placing a balls into b bins in such a way that no bin is empty (we write out the a elements one after the other and pick $(b - 1)$ dividing lines in between two elements).

The number of ways of putting the $(n - d)$ zeros into the $\lfloor \ell/2 \rfloor + 1$ even sets is $\binom{n-d}{\lfloor \ell/2 \rfloor}$ (we have $(n - d)$ instead of $(n - d - 1)$ because we also allow S_0 to be empty). Likewise, the number of ways of putting the d ones into the $\lceil \ell/2 \rceil$ odd sets is $\binom{d-1}{\lceil \ell/2 \rceil - 1}$.

So the total number of ways of placing the ones and zeros into these sets is

$$\binom{n-d}{\lfloor \ell/2 \rfloor} \cdot \binom{d-1}{\lceil \ell/2 \rceil - 1}, \tag{3.16}$$

and this is therefore the number of vectors v of weight ℓ that lead to a codeword c of weight d. Since the total number of vectors v of weight ℓ is $\binom{n}{\ell}$, the result follows. ■

Theorem 3.11.

$$\overline{M}_{w\ell} = \binom{n}{\ell} \cdot \left(P_w\right)^{\ell} \cdot \left(1 - P_w\right)^{n-\ell}, \tag{3.17}$$

where P_w denotes the probability that a fixed entry of v is equal to 1, given that the weight of u is w:

$$P_w = \frac{1}{2} - \frac{1}{2} \cdot \left(1 - \frac{2w}{k}\right)^{\lfloor k^y \rfloor}, \quad \textit{where} \ \ k = \lceil nR \rceil . \tag{3.18}$$

Proof: Let $u \in \mathbb{F}_2^k$ be a message vector of weight w. First note that a fixed entry v_i of v depends only on u and column i of M, which is generated independently of all other columns. $v_i = 0$ if and only if among the $\lfloor k^y \rfloor$ components chosen from $\{1, \ldots, k\}$ to construct column i, an even number are in $\text{supp}(u)$. So the distribution on the possible values of v depends on $\text{wgt}(u)$, but not on u itself (all u's of weight w lead to the same distribution).

Each time a component is chosen, it will hit $\text{supp}(u)$ with probability $\frac{w}{k}$. So

$$\begin{aligned}\Pr\Big[v_i = 0\Big] &= \textstyle\sum_{i=0}^{\lfloor k^y \rfloor/2} \binom{\lfloor k^y \rfloor}{2i} \cdot \big(\frac{w}{k}\big)^{2i} \cdot \big(1 - \frac{w}{k}\big)^{\lfloor k^y \rfloor - 2i} \\ &= \tfrac{1}{2}\Big[\big((1 - \tfrac{w}{k}) + \tfrac{w}{k}\big)^{\lfloor k^y \rfloor} \quad + \quad \big((1 - \tfrac{w}{k}) - \tfrac{w}{k}\big)^{\lfloor k^y \rfloor}\Big] \\ &= \tfrac{1}{2} + \tfrac{1}{2}\left(1 - \tfrac{2w}{k}\right)^{\lfloor k^y \rfloor}.\end{aligned}$$

We therefore see from the definition of P_w in (3.18) that

$$P_w = 1 - \Pr\Big[v_i = 0\Big] = \Pr\Big[v_i = 1\Big]. \tag{3.19}$$

Since the components v_i are independent, constructing $v = (v_1, \ldots, v_n)$ consists of n Bernoulli trials, where $v_i = 1$ with probability P_w, so (3.17) follows. ■

Now the binomial function $\binom{a}{b}$ is a map

$$\binom{\cdot}{\cdot} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}.$$

We extend it to be defined over all non negative real numbers:

$$\binom{\cdot}{\cdot} : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}.$$

This is done using the gamma function (which is an extension of the factorial function to real numbers), the details are given in Appendix A. This extension has all the expected properties, in particular the following bound on $\binom{a}{b}$ still holds:

Proposition 3.12. *For any $a, b \in \mathbb{R}$ with $1 \leq b \leq a$ we have*

$$\binom{a}{b} \leq 2^{a \cdot h(b/a)}.$$

Proof: See Appendix A. ■

We also have the following proposition:

Proposition 3.13. *For any $n, \ell \in \mathbb{N}^*$, $0 < \delta < \frac{1}{2}$ with $1 \leq \ell \leq 2n\delta$, letting $\overline{\delta} = 1 - \delta$ we have*

$$\sum_{d=1}^{n\delta} \binom{n-d}{\lfloor \ell/2 \rfloor} \binom{d-1}{\lceil \ell/2 \rceil - 1} \leq n\delta \cdot \binom{n\delta}{\ell/2} \binom{n\overline{\delta}}{\ell/2}. \tag{3.20}$$

Proof: See Appendix A. ■

Throughout we will set $\overline{\delta} = 1 - \delta$. We are now ready to compute the bound we set out to find at the beginning of this section:

Theorem 3.14. *If $\mathcal{C}$ is a code sampled from the ensemble $\mathcal{C}_{RA}(n, R, y)$, then*

$$\Pr\Big[d_{\min}(\mathcal{C}) \le n\delta\Big] \;\le\; n\delta \cdot \sum_{w=1}^{nR}\sum_{\ell=1}^{2n\delta} \overbrace{\binom{nR}{w}}^{s(n,w)} \cdot \overbrace{P_w^{\ell}\cdot\big(1-P_w\big)^{n-\ell}}^{f(n,\ell,w)} \cdot \overbrace{\binom{n\delta}{\ell/2}\binom{n\overline{\delta}}{\ell/2}}^{g(n,\ell)}, \tag{3.21}$$

where

$$P_w = \frac{1}{2} - \frac{1}{2}\cdot\left(1-\frac{2w}{nR}\right)^{\lfloor (nR)^y \rfloor}. \tag{3.22}$$

Proof: Let $d_{\min} = d_{\min}(\mathcal{C})$. We saw in Lemma 3.9 that

$$\Pr\Big[d_{\min} \le n\delta\Big] \le \sum_{d=1}^{n\delta}\sum_{w=1}^{nR}\binom{nR}{w}\cdot\sum_{\ell=1}^{n}\overline{M}_{w\ell}\cdot\overline{A}_{\ell d}.$$

Now plugging in the expressions we computed in Theorems 3.10 and 3.11 for $\overline{A}_{\ell d}$ and $\overline{M}_{w\ell}$ (and moving the sums around), we obtain

$$\Pr\Big[d_{\min} \le n\delta\Big] \le n\delta\cdot\sum_{w=1}^{nR}\sum_{\ell=1}^{n}\binom{nR}{w}\cdot P_w^{\ell}\cdot\big(1-P_w\big)^{n-\ell}\cdot\sum_{d=1}^{n\delta}\binom{n-d}{\lfloor \ell/2\rfloor}\binom{d-1}{\lceil \ell/2\rceil - 1}, \tag{3.23}$$

where P_w is defined in (3.18). Therefore applying Proposition 3.13 leads to:

$$\Pr\Big[d_{\min} \le n\delta\Big] \;\le\; n\delta\cdot\sum_{w=1}^{nR}\sum_{\ell=1}^{n}\binom{nR}{w}\cdot P_w^{\ell}\cdot\big(1-P_w\big)^{n-\ell}\cdot\binom{n\delta}{\ell/2}\binom{n\overline{\delta}}{\ell/2}.$$

Finally notice that since

$$\frac{\ell}{2} > n\delta \implies \binom{n\delta}{\ell/2} = 0,$$

in our sum we only need to consider values of ℓ up to $2n\delta$, and therefore the required result (3.21) follows. ■

3.4.3 Proof Outline

Our aim is to show that $\mathcal{C}_{RA}(n, R, y)$ approaches the Gilbert-Varshamov bound as n tends to infinity. So if we let h denote the binary entropy function, then we want to show that when $R < 1 - h(\delta)$, the probability in (3.21) tends to 0 as n tends to infinity. Indeed, in this case (3.21) is an upper bound on the probability that the GV-bound is *not* achieved.

Let

$$m(n,\ell,w) = \overbrace{\binom{nR}{w}}^{s(n,w)}\cdot\overbrace{P_w^{\ell}\cdot\big(1-P_w\big)^{n-\ell}}^{f(n,\ell,w)}\cdot\overbrace{\binom{n\delta}{\ell/2}\binom{n\overline{\delta}}{\ell/2}}^{g(n,\ell)}$$

be the term inside the double sum in (3.21). Our goal is to prove the following theorem:

Theorem 3.15. *Suppose we are given $0 < R$, $0 < y < 1$ and $0 < \delta < \frac{1}{2}$ with $R < 1 - h(\delta)$. Then there are $N, \tau > 0$ (depending only on R, δ and y) for which*

$$n \geq N \implies \forall \ell = 1, \ldots, \lfloor 2n\delta \rfloor, \forall w = 1, \ldots, \lfloor nR \rfloor \ : \ m(n, \ell, w) \leq \exp\big(-\tau \cdot n^y\big). \tag{3.24}$$

From this theorem we deduce that each term $m(n, \ell, w)$ in the double sum (3.21) is superpolynomially small in n, and since there is only a polynomial number of terms, the whole sum will converge to 0 as n tends to infinity.

Outline of the Proof: The proof of Theorem 3.15 below is very long and technical. $m(n, \ell, w)$ is a complicated expression, and which ones of its terms dominate for large n depends on the sizes of ℓ and w relative to n. To measure these we define

$$\alpha = \frac{\ell}{n}, \qquad \gamma = \frac{w}{n^{1-y}}. \tag{3.25}$$

Notice that for the values of ℓ and w that interest us we have $\alpha \in [\frac{1}{n}, 2\delta]$ and $\gamma \in [\frac{1}{n^{1-y}}, R \cdot n^y]$. We will show that there are constants $\hat{A}, \hat{\Gamma_\ell}$ and $\hat{\Gamma_u}$ (depending only on R, δ and y) that enable us to divide the proof into four cases:

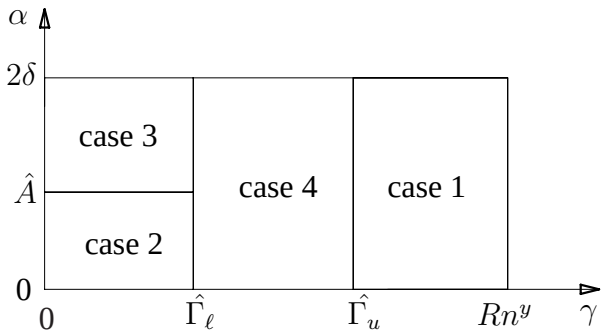
• **Case 1:** $\gamma \geq \hat{\Gamma_u}$, and any α. For each n large enough this will cover all pairs (ℓ, w) with $\ell = 1, \ldots, \lfloor 2\delta n \rfloor$ and $w = \left\lceil \hat{\Gamma_u} \cdot n^{1-y} \right\rceil, \ldots, \lfloor nR \rfloor$.

• **Case 2:** $\gamma \leq \hat{\Gamma_\ell}$, $\alpha \leq \hat{A}$. For each n large enough this will cover all pairs (ℓ, w) with $\ell = 1, \ldots, \left\lfloor \hat{A}n \right\rfloor$ and $w = 1, \ldots, \left\lfloor \hat{\Gamma_\ell} \cdot n^{1-y} \right\rfloor$.

• **Case 3:** $\gamma \leq \hat{\Gamma_\ell}$, $\alpha \geq \hat{A}$. For each n large enough this will cover all pairs (ℓ, w) with $\ell = \left\lceil \hat{A}n \right\rceil, \ldots, \lfloor 2\delta n \rfloor$ and $w = 1, \ldots, \left\lfloor \hat{\Gamma_\ell} \cdot n^{1-y} \right\rfloor$.

• **Case 4:** $\hat{\Gamma_\ell} \leq \gamma \leq \hat{\Gamma_u}$, and any α. For each n large enough this will cover all pairs (ℓ, w) with $\ell = 1, \ldots, \lfloor 2\delta n \rfloor$ and $w = \left\lceil \hat{\Gamma_u} \cdot n^{1-y} \right\rceil, \ldots, \left\lfloor \hat{\Gamma_\ell} \cdot n^{1-y} \right\rfloor$.

The following diagram illustrates the splitting of the problem into our four cases:



3.4.4 Case 1: Large γ, Any α.

Outline: Recall that we view the encoding as a two stage process. Given a codeword $u \in \mathbb{F}_2^k$ we first compute $v = u \cdot M$, and then the codeword $c = v \cdot A$. The idea in Case 1 is that as γ gets large, $P_w = \frac{1}{2} - \frac{1}{2} \cdot \left(1 - \frac{2w}{nR}\right)^{(nR)^y}$ will get close to $\frac{1}{2}$. Recall that P_w represents the probability that a fixed entry v_i of v is equal to 1, see (3.19). So P_w being close to $\frac{1}{2}$ means that v is close to being a (uniform) random vector. Since the codeword can be expressed as $c = v \cdot A$ and A is bijective, this means that c is also close to being a (uniform) random vector. So as γ gets large our code resembles a uniform random code, we therefore proceed in a similar way to the proof of Theorem 3.4.

The next lemma formalizes the idea that P_w gets close to $\frac{1}{2}$ when γ gets large.

Lemma 3.16. *For any $\epsilon_1 > 0$, there are N_1, Γ_1 with*

$$N \geq N_1, \gamma \geq \Gamma_1 \implies \left| P_w - \frac{1}{2} \right| \leq \epsilon_1. \tag{3.26}$$

Proof: See Appendix C. ■

Lemma 3.17. *For any $\epsilon_5 > 0$, there are N_5, Γ_5 with*

$$n \geq N_5, \gamma \geq \Gamma_5 \implies \left(P_w\right)^{\ell} \cdot \left(1 - P_w\right)^{n-\ell} \leq 2^{-n(1-\epsilon_5)}.$$

Proof: From Lemma 3.16 we see that for any ϵ_1, if n and γ are large enough then

$$\left(P_w\right)^{\ell} \cdot \left(1 - P_w\right)^{n-\ell} \leq \left(\frac{1}{2} + \epsilon_1\right)^n. \tag{3.27}$$

So given ϵ_5, by choosing ϵ_1 so that $\frac{1}{2} + \epsilon_1 = 2^{\epsilon_5 - 1}$, the right hand side of (3.27) becomes $2^{-n(1-\epsilon_5)}$ and so the result follows. ■

Recall that $\overline{\delta}$ is defined as $\overline{\delta} = 1 - \delta$.

Lemma 3.18. $\forall n \in \mathbb{N}^*, \delta < \frac{1}{2}, \ell = 0, \ldots, \lfloor 2n\delta \rfloor$:

$$\delta \cdot h\left(\frac{\ell}{2n\delta}\right) + \overline{\delta} \cdot h\left(\frac{\ell}{2n\overline{\delta}}\right) \leq h(\delta). \tag{3.28}$$

Proof: First note that the derivative of the binary entropy function h is

$$\frac{d}{dx} h(x) = \log_2\left(\frac{1}{x} - 1\right). \tag{3.29}$$

Let

$$q(x) = \delta \cdot h\left(\frac{x}{\delta}\right) + \overline{\delta} \cdot h\left(\frac{x}{\overline{\delta}}\right). \tag{3.30}$$

We want to upper bound the function $q(x)$ over the range $0 \leq x \leq \delta$ (identifying x with $\frac{\ell}{2n}$). We have

$$\frac{d}{dx}q(x) = \log_2\left(\frac{\delta}{x} - 1\right) + \log_2\left(\frac{\overline{\delta}}{x} - 1\right) = \log_2\left(\frac{\delta\overline{\delta}}{x^2} - \frac{\delta + \overline{\delta}}{x} + 1\right), \tag{3.31}$$

so that

$$\frac{d}{dx}q(x) = 0 \iff \frac{\delta\overline{\delta}}{x^2} - \frac{1}{x} + 1 = 1 \iff x = \delta\overline{\delta}. \tag{3.32}$$

It can then easily be checked that $x = \delta\overline{\delta}$ is a maximum for $q(x)$. Therefore

$$q(x) \leq q(\delta\overline{\delta}) = \delta \cdot h(\overline{\delta}) + \overline{\delta} \cdot h(\delta) = h(\delta), \tag{3.33}$$

where the last equality follows from the fact that $h(\delta) = h(\overline{\delta})$. ■

Let us summarize the situation so far. We want to show that for any R and $\delta < 1 - h(R)$, the following expression (see (3.21))

$$n\delta \cdot \sum_{w=1}^{nR}\sum_{\ell=1}^{n} \overbrace{\binom{nR}{w}}^{s(n,w)} \cdot \overbrace{P_w^{\ell} \cdot \left(1 - P_w\right)^{n-\ell}}^{f(n,\ell,w)} \cdot \overbrace{\binom{n\delta}{\ell/2}\binom{n\overline{\delta}}{\ell/2}}^{g(n,\ell)},$$

tends to zero as n tends to infinity. Our approach is to show that each term inside the double sum is super-polynomially small in n, and so since there are only a polynomial number of terms, the whole sum will go to zero as n tends to infinity. We defined

$$m(n, \ell, w) = s(n, w) \cdot f(n, \ell, w) \cdot g(n, \ell), \tag{3.34}$$

to be the expression inside the double sum.

Proposition 3.19. *There are $N_6, \Gamma_6, \tau_6 > 0$ with*

$$n \geq N_6, \gamma \geq \Gamma_6 \implies m(n, \ell, w) \leq \exp(-\tau_6 \cdot n^y). \tag{3.35}$$

Proof: Using the inequality $\binom{a}{b} \leq 2^{a \cdot h(b/a)}$ (see Proposition 3.12), we obtain:

$$s(n, w) = \binom{nR}{w} \leq 2^{nR \cdot h(\frac{w}{nR})}, \tag{3.36}$$

and by Lemma 3.18

$$g(n, \ell) = \binom{n\delta}{\ell/2}\binom{n\overline{\delta}}{\ell/2} \leq 2^{n\left(\delta \cdot h(\frac{\ell}{2n\delta}) + \overline{\delta} \cdot h(\frac{\ell}{2n\overline{\delta}})\right)} \leq 2^{n \cdot h(\delta)}. \tag{3.37}$$

We also know from Lemma 3.17 that for any $\epsilon_5 > 0$, there are N_5, Γ_5 with

$$n \geq N_5, \gamma \geq \Gamma_5 \implies f(n, \ell, w) \leq 2^{-n(1-\epsilon_5)}. \tag{3.38}$$

Since $m(n, \ell, w)$ is the product of $s(n, w)$, $f(n, \ell, w)$ and $g(n, \ell)$ (see (3.34)), combining (3.36), (3.37) and (3.38), we see that for any $\epsilon_5 > 0$, there are N_5, Γ_5 with

$$n \geq N_5, \gamma \geq \Gamma_5 \implies m(n, \ell, w) \leq 2^{nR \cdot h(\frac{w}{nR}) - n(1-\epsilon_5) + n \cdot h(\delta)} \leq 2^{-n\left[1 - h(\delta) - R - \epsilon_5\right]}. \tag{3.39}$$

Setting $\epsilon_5 = \frac{1}{2} \cdot \big(1 - h(\delta) - R\big)$, we see that

$$m(n, \ell, w) \leq \exp\big(-\tau_6 \cdot n\big),$$

where $\tau_6 = \frac{1}{2} \cdot \big(1 - h(\delta) - R\big) \cdot \ln(2)$. Furthermore since $R < 1 - h(\delta)$, we have $\tau_6 > 0$.

It is clear that $-\tau_6 \cdot n \leq -\tau_6 \cdot n^y$ (since $\tau_6 > 0$ and $0 < y < 1$). So we have actually shown a stronger statement than the required result. We set $N_6 = N_5$ and $\Gamma_6 = \Gamma_5$ to complete the proof. ■

The value of Γ_6 is what we will use for our constant $\hat{\Gamma_u}$:

Definition 3.20. We set $\hat{\Gamma_u}$ to be some arbitrary value of Γ_6 that satisfies (3.35) (such a value exists by Proposition 3.19).

It is important to note that $\hat{\Gamma_u}$ is a *constant*, in the sense that it depends only on R, δ and y (which we have fixed throughout). In particular it does *not* depend on n. Throughout, a value x written as $\hat{x}$ indicates that it depends only on R, δ and y. All variables written this way depend only on R, δ and y, but the converse will not be true.

This result of Case 1 is summarized in the following theorem:

Theorem 3.21. *Suppose we are given $0 < R < 1$ and $0 < \delta < \frac{1}{2}$ with $R < 1 - h(\delta)$. Let $\hat{\Gamma_u}$ be defined as in Definition 3.20. Then there are $N_6, \tau_6 > 0$ (depending only on R, δ and y) for which*

$$n \geq N_6 \implies \forall \ell = 1, \ldots, \lfloor 2n\delta \rfloor\,,\ \forall w = \left\lceil \hat{\Gamma_u} \cdot n^{1-y} \right\rceil, \ldots, \lfloor nR \rfloor :\ m(n, \ell, w) \leq \exp\big(-\tau_6 \cdot n^y\big).$$

3.4.5 Case 2: Small α, Small γ.

We recall once more the definitions of α and γ:

$$\alpha = \frac{\ell}{n}, \qquad \gamma = \frac{w}{n^{1-y}}. \tag{3.40}$$

Outline: For this case we will show that there exist constants $A, \Gamma, \tau > 0$ (depending only on R, δ and y) for which for all n large enough the statement

$$m(n, \ell, w) \leq \exp\big(-\tau \cdot n^y\big)$$

holds for all $\ell = 1, \ldots, \lfloor A \cdot n \rfloor$ and $w = 1, \ldots, \left\lfloor \Gamma \cdot n^{1-y} \right\rfloor$ (i.e. for all ℓ, w with $\alpha \leq A$ and $\gamma \leq \Gamma$).

We will make use of the following theorem:

Theorem 3.22. *For any b, x with $b \geq 1$ and $0 \leq x \leq 1$ we have*

$$1 - x \leq \Big(1 - \frac{x}{b}\Big)^b. \tag{3.41}$$

Proof: See Appendix C. ■

We start with the following lemma:

Lemma 3.23. *For all* $0 < R < 1, n \in \mathbb{N}^*, w = 0, \ldots, nR$*, if* $nR \geq 1$ *then*

$$P_w \leq \frac{w}{(nR)^{1-y}}. \tag{3.42}$$

Proof: We start by recalling the definition of P_w (see (3.18)):

$$P_w = \frac{1}{2} - \frac{1}{2} \cdot \Big(1 - \frac{2w}{nR}\Big)^{(nR)^y} = \frac{1}{2} \cdot \bigg(1 - \Big(1 - \frac{2w/(nR)^{1-y}}{(nR)^y}\Big)^{(nR)^y}\bigg).$$

Now setting

$$x = \frac{2w}{(nR)^{1-y}}, \qquad\qquad b = (nR)^y, \tag{3.43}$$

from Theorem 3.22 we obtain

$$P_w = \frac{1}{2} \cdot \bigg(1 - \Big(1 - \frac{x}{b}\Big)^b\bigg) \leq \frac{1}{2} \cdot \Big(1 - (1 - x)\Big) = \frac{w}{(nR)^{1-y}}, \tag{3.44}$$

as required. ■

Lemma 3.24. *For any* $a > 1$*, there is* $X > 0$ *with*

$$0 < x \leq X \implies e^{-x} \leq 1 - \frac{x}{a}.$$

Proof: The curve of e^{-x} is convex and goes through the point $(0, 1)$, where its derivative is -1. The line $1 - \frac{x}{a}$ also goes through $(0, 1)$, and its derivative is $-\frac{1}{a} > -1$. So it will intersect the curve of e^{-x} at some other point (x_0, y_0), with $x_0 > 0$. We let $X = x_0$. ■

Lemma 3.25. *For any* $a > 1$*, there is* Γ_a *with*

$$\gamma \leq \Gamma_a \implies n \ln(1 - P_w) \leq -\frac{1}{a} \cdot \frac{wn^y}{R^{1-y}}.$$

Proof: We first recall that γ is defined as

$$\gamma = \frac{w}{n^{1-y}}, \tag{3.45}$$

and

$$P_w = \frac{1}{2} - \frac{1}{2} \cdot \Big(1 - \frac{2w}{nR}\Big)^{(nR)^y} \geq \frac{1}{2} - \frac{1}{2} \cdot \exp\Big(\frac{-2w}{(nR)^{1-y}}\Big), \tag{3.46}$$

since for any $x \in \mathbb{R}$ we have $1 + x \leq \exp(x)$. Suppose we have a fixed $a > 1$. Using Lemma 3.24 with $x = \frac{2\gamma}{R^{1-y}}$ (and recalling that R and y are fixed), we deduce that there is $\Gamma > 0$ for which

$$\gamma \leq \Gamma \implies \exp\bigg(-\frac{2\gamma}{R^{1-y}}\bigg) \leq 1 - \frac{1}{a} \cdot \frac{2\gamma}{R^{1-y}} \tag{3.47}$$

We also know that $\forall x \leq 1$ we have $\ln(1-x) \leq -x$, and so

$$\begin{aligned}
\gamma \leq \Gamma \implies n\ln\big(1-P_w\big) &\leq -nP_w \\
&\leq \tfrac{n}{2}\Big(-1+\exp\Big(\tfrac{-2\gamma}{R^{1-y}}\Big)\Big) && \text{(using (3.46))} \\
&\leq \tfrac{n}{2}\Big(-1+1-\tfrac{1}{a}\cdot\tfrac{2\gamma}{R^{1-y}}\Big) && \text{(using (3.47), since } \gamma \leq \Gamma\text{)} \\
&= -\tfrac{1}{a}\cdot\tfrac{n^y w}{R^{1-y}} && \text{(using (3.45)).}
\end{aligned}$$

Setting $\Gamma_a = \Gamma$ then gives us the required result. ■

Definition 3.26. We define

$$\hat{a} = \Big[2\sqrt{\delta\overline{\delta}} + \frac{3}{4}\big(1-2\sqrt{\delta\overline{\delta}}\big)\Big]^{-1}. \tag{3.48}$$

This choice for $\hat{a}$ will become clear later in the section. Notice that $\hat{a}$ depends only on δ. A straightforward inspection shows that $\hat{a} > 1$. Therefore applying Lemma 3.25 we obtain:

Corollary 3.27. *Let $\hat{a}$ be defined as in (3.48). Then there is $\hat{\Gamma_a} > 0$ with*

$$\gamma \leq \hat{\Gamma_a} \implies n\ln(1-P_w) \leq -\frac{1}{\hat{a}}\cdot\frac{wn^y}{R^{1-y}}. \tag{3.49}$$

Definition 3.28. Let $\hat{\Gamma_a}$ be a fixed value that satisfies (3.49).

Recall that we are trying to show that the expression

$$m(n,\ell,w) = s(n,w)\cdot f(n,\ell,w)\cdot g(n,\ell)$$

from (3.21) is superpolynomially small in n. Setting $m_1(n,\ell,w) = \ln\big(m(n,\ell,w)\big)$, we will show that there are constants N and $\tau > 0$ for which $n \geq N$ implies that for all appropriate values of ℓ and w we have:

$$m_1(n,\ell,w) \leq -\tau\cdot n^y. \tag{3.50}$$

Definition 3.29. We define

$$\begin{aligned}
s_1(n,w) &= \ln\big(s(n,w)\big). \\
f_1(n,\ell,w) &= \ln\big(f(n,\ell,w)\big). \\
g_1(n,\ell) &= \ln\big(g(n,\ell)\big).
\end{aligned}$$

Note that

$$m_1(n,\ell,w) = s_1(n,w) + f_1(n,\ell,w) + g_1(n,\ell).$$

Propositions 3.30, 3.31 and 3.32, will provide upper bounds on $s_1(n,w)$, $f_1(n,\ell,w)$ and $g_1(n,\ell)$. We will then use these to prove (3.50).

Proposition 3.30. *We can upper bound $s_1(n,w)$ in the following ways:*

$$s_1(n,w) \;\leq\; -w\ln\left(\frac{w}{nR}\right) \;-\; nR\ln\left(1-\frac{w}{nR}\right). \tag{3.51}$$

$$s_1(n,w) \;\leq\; -w\cdot\frac{\ln\left(\frac{w}{nR}\right)-1}{\ln(2)}. \tag{3.52}$$

Proof:
• Using the inequality $\binom{a}{b} \leq 2^{a\cdot h(b/a)}$ we obtain:

$$s(n,w) = \binom{nR}{w} \leq 2^{nR\cdot h\left(\frac{w}{nR}\right)} = e^{-w\ln\left(\frac{w}{nR}\right)+(w-nR)\ln\left(1-\frac{w}{nR}\right)}.$$

Note that $1-\frac{w}{nR}<1$, so $w\ln\left(1-\frac{w}{nR}\right)<0$. Since $s_1(n,w)=\ln\big(s(n,w)\big)$, (3.51) then follows.

• We have the following general bound on the binomial coefficients:

$$\binom{a}{b} \leq \Big(\frac{a\cdot e}{b}\Big)^b,$$

which directly leads to (3.52). ■

Proposition 3.31. *Let $\hat{a}$ and $\hat{\Gamma}_a$ be taken from Definitions 3.26 and 3.28. If $\gamma \leq \hat{\Gamma}_a$ and $nR>1$ then*

$$f_1(n,\ell,w) \;\leq\; \ell\ln\left(\frac{w}{(nR)^{1-y}}\right) \;-\; \frac{wn^y}{R^{1-y}}\cdot\frac{1}{\hat{a}} \;-\; \ell\ln\left(1-\frac{w}{(nR)^{1-y}}\right).$$

Proof: Recall from (3.14) that $f(n,\ell,w)$ is defined as

$$f(n,\ell,w) = \big(P_w\big)^\ell\cdot\big(1-P_w\big)^{n-\ell}.$$

This means that

$$f_1(n,\ell,w) = \ell\ln(P_w) + n\ln(1-P_w) - \ell\ln(1-P_w). \tag{3.53}$$

Now Lemma 3.23 (which applies, since we are assuming that $nR>1$) tells us that

$$P_w \leq \frac{w}{(nR)^{1-y}},$$

and Corollary 3.27 (which applies, since we are assuming that $\gamma\leq\hat{\Gamma}_a$) tells us that

$$n\ln(1-P_w) \leq -\frac{1}{\hat{a}}\cdot\frac{wn^y}{R^{1-y}}.$$

Combining these with (3.53) we deduce:

$$f_1(n,\ell,w) \leq \ell\ln\left(\frac{w}{(nR)^{1-y}}\right) - \frac{wn^y}{R^{1-y}}\cdot\frac{1}{\hat{a}} - \ell\ln\left(1-\frac{w}{(nR)^{1-y}}\right), \tag{3.54}$$

as required. ■

Proposition 3.32. *We can upper bound $g_1(n,\ell)$ in the following ways:*

$$\begin{aligned} g_1(n,\ell) \;\leq\; & -\tfrac{\ell}{2}\ln\left(\tfrac{\ell}{2n\delta}\right) \;-\; \tfrac{\ell}{2}\ln\left(\tfrac{\ell}{2n\overline{\delta}}\right) \\ & -\; n\delta\cdot\ln\left(1-\tfrac{\ell}{2n\delta}\right) \;-\; n\overline{\delta}\cdot\ln\left(1-\tfrac{\ell}{2n\overline{\delta}}\right). \end{aligned} \tag{3.55}$$

$$\begin{aligned} g_1(n,\ell) \;\leq\; & \tfrac{\ell}{2}\ln\left(\tfrac{2n\delta}{\ell}-1\right) \;+\; \tfrac{\ell}{2}\ln\left(\tfrac{2n\overline{\delta}}{\ell}-1\right) \\ & -\; n\delta\cdot\ln\left(1-\tfrac{\ell}{2n\delta}\right) \;-\; n\overline{\delta}\cdot\ln\left(1-\tfrac{\ell}{2n\overline{\delta}}\right). \end{aligned} \tag{3.56}$$

Proof: Recall the definition of $g(n,\ell)$ (see (3.14)):

$$g(n,\ell) = \binom{n\delta}{\ell/2}\cdot\binom{n\overline{\delta}}{\ell/2}.$$

Again using the inequality $\binom{a}{b} \leq 2^{a\cdot h(b/a)}$ we obtain:

$$\begin{aligned} \tbinom{n\delta}{\ell/2} \;&\leq\; 2^{n\delta\cdot h\left(\frac{\ell}{2n\delta}\right)} \\ &\leq\; 2^{-\frac{\ell}{2}\log_2\left(\frac{\ell}{2n\delta}\right)-(\frac{\ell}{2}-n\delta)\log_2\left(1-\frac{\ell}{2n\delta}\right)}, \end{aligned}$$

and a similar bound for $\binom{n\overline{\delta}}{\ell/2}$. Since $g_1(n,\ell) = \ln\big(g(n,\ell)\big)$, we obtain

$$\begin{aligned} g_1(n,\ell) \;\leq\; & -\tfrac{\ell}{2}\cdot\ln\left(\tfrac{\ell}{2n\delta}\right) - \left(n\delta-\tfrac{\ell}{2}\right)\cdot\ln\left(1-\tfrac{\ell}{2n\delta}\right) \\ & -\tfrac{\ell}{2}\cdot\ln\left(\tfrac{\ell}{2n\overline{\delta}}\right) - \left(n\delta-\tfrac{\ell}{2}\right)\cdot\ln\left(1-\tfrac{\ell}{2n\overline{\delta}}\right). \end{aligned} \tag{3.57}$$

• **1)** Because $\ln\left(1-\frac{\ell}{2n\delta}\right) < 0$ and $\ln\left(1-\frac{\ell}{2n\overline{\delta}}\right) < 0$, we can remove terms to obtain (3.55):

$$g_1(n,\ell) \leq -\frac{\ell}{2}\ln\left(\frac{\ell}{2n\delta}\right) - n\delta\ln\left(1-\frac{\ell}{2n\delta}\right) - \frac{\ell}{2}\ln\left(\frac{\ell}{2n\overline{\delta}}\right) - n\overline{\delta}\ln\left(1-\frac{\ell}{2n\overline{\delta}}\right).$$

• **2)** In general, for any $x > 0$ we have

$$-\frac{\ell}{2}\cdot\ln(x) + \frac{\ell}{2}\cdot\ln(1-x) = \frac{\ell}{2}\cdot\ln\left(\frac{1-x}{x}\right) = \frac{\ell}{2}\cdot\ln\left(\frac{1}{x}-1\right),$$

so applying this with $x = \frac{\ell}{2n\delta}$ and then $x = \frac{\ell}{2n\overline{\delta}}$, (3.57) can be rewritten as (3.56). ■

We summarize the bounds obtained by the last three theorems in (3.58) below: Let $\hat{a}$ and $\hat{\Gamma_a}$ be taken from

Definitions 3.26 and 3.28. If $\gamma \leq \hat{\Gamma}_a$ and $n > \frac{1}{R}$ then

$$\begin{aligned}
s_1(n,w) &\leq \overbrace{-w \cdot \frac{\ln\left(\frac{w}{nR}\right)-1}{\ln(2)}}^{t_1}. \\
f_1(n,\ell,w) &\leq \overbrace{\ell \ln\left(\frac{w}{(nR)^{1-y}}\right)}^{t_2} \quad \overbrace{-\frac{wn^y}{R^{1-y}} \cdot \frac{1}{\hat{a}}}^{t_3} \quad \overbrace{-\ell \ln\left(1-\frac{w}{(nR)^{1-y}}\right)}^{t_4}. \\
g_1(n,\ell) &\leq \overbrace{-\frac{\ell}{2}\ln\left(\frac{\ell}{2n\delta}\right)}^{t_5} \quad \overbrace{-\frac{\ell}{2}\ln\left(\frac{\ell}{2n\overline{\delta}}\right)}^{t_6} \\
&\quad \overbrace{-n\delta \cdot \ln\left(1-\frac{\ell}{2n\delta}\right)}^{t_7} \quad \overbrace{-n\overline{\delta} \cdot \ln\left(1-\frac{\ell}{2n\overline{\delta}}\right)}^{t_8}.
\end{aligned} \tag{3.58}$$

Recall that

$$m_1(n,\ell,w) = s_1(n,w) + f_1(n,\ell,w) + g_1(n,\ell).$$

So using these we obtain a bound on $m_1(n,\ell,w)$ consisting in a sum of eight terms $t_1,\ldots,t_8$. Let $m_2(n,\ell,w)$ be obtained by removing terms t_1 and t_4 from $m_1(n,\ell,w)$:

$$\begin{aligned}
m_2(n,\ell,w) &= \ell \cdot \ln\left(\tfrac{w}{(nR)^{1-y}}\right) - \tfrac{wn^y}{R^{1-y}} \cdot \tfrac{1}{\hat{a}} - \tfrac{\ell}{2} \cdot \ln\left(\tfrac{\ell}{2n\delta}\right) - \tfrac{\ell}{2} \cdot \ln\left(\tfrac{\ell}{2n\overline{\delta}}\right) \\
&\quad - n\delta \cdot \ln\left(1-\tfrac{\ell}{2n\delta}\right) - n\overline{\delta} \cdot \ln\left(1-\tfrac{\ell}{2n\overline{\delta}}\right).
\end{aligned} \tag{3.59}$$

Outline for the rest of Case 2: We will first show (Lemma 3.34) that for α, γ small enough, we have $m_2(n,\ell,w) = \theta\big(-wn^y\big)$. We will then show (Lemma 3.35) that the remaining terms t_1 and t_4 are $o(n^y)$. We then can deduce that $m_1(n,\ell,w) = \theta\big(-wn^y\big)$.

The following lemma will be useful:

Lemma 3.33. *For all $x, b \in \mathbb{R}_{>0}$ we have*

$$-x\ln(bx) \leq \frac{1}{be}.$$

Proof: See Appendix C. ■

Lemma 3.34. *Let $\hat{a}$ and $\hat{\Gamma}_a$ be taken from Definitions 3.26 and 3.28. Then there are $A_7, \tau_7 > 0$ with*

$$\alpha \leq A_7 \implies m_2(n,\ell,w) \leq -\tau_7 \cdot w \cdot n^y.$$

Proof: $m_2(m,\ell,w)$ was defined in (3.59) as

$$\begin{aligned} m_2(n,\ell,w) \quad &= \quad \ell \ln\left(\tfrac{w}{(nR)^{1-y}}\right) \quad - \quad \tfrac{wn^y}{R^{1-y}} \cdot \tfrac{1}{\hat{a}} \quad - \quad \tfrac{\ell}{2}\ln\left(\tfrac{\ell}{2n\delta}\right) \quad - \quad \tfrac{\ell}{2}\ln\left(\tfrac{\ell}{2n\overline{\delta}}\right) \\ &\quad - \quad n\delta\ln\left(1-\tfrac{\ell}{2n\delta}\right) \quad - \quad n\overline{\delta}\ln\left(1-\tfrac{\ell}{2n\overline{\delta}}\right). \end{aligned} \tag{3.60}$$

First note that the third and fourth terms can be expressed as

$$-\frac{\ell}{2}\cdot\ln\left(\frac{\ell}{2n\delta}\right) - \frac{\ell}{2}\cdot\ln\left(\frac{\ell}{2n\overline{\delta}}\right) = -\frac{\ell}{2}\cdot\ln\left(\frac{\ell^2}{4n^2\delta\overline{\delta}}\right) = -\ell\cdot\ln\left(\frac{\ell}{c_1\cdot n}\right),$$

where we let $c_1 = 2\sqrt{\delta\overline{\delta}}$. Since $0 < \delta < \frac{1}{2}$ and $\overline{\delta} = 1-\delta$ (by definition), we have $0 < \delta\overline{\delta} < \frac{1}{4}$, and therefore

$$0 < c_1 < 1. \tag{3.61}$$

Next, recalling that $\alpha = \frac{\ell}{n}$, we can express the last two terms of (3.60) as

$$\begin{aligned} -n\delta\ln\left(1-\tfrac{\ell}{2n\delta}\right) - n\overline{\delta}\ln\left(1-\tfrac{\ell}{2n\overline{\delta}}\right) \quad &= \quad -\tfrac{\ell\delta}{\alpha}\ln\left(1-\tfrac{\alpha}{2\delta}\right) - \tfrac{\ell\overline{\delta}}{\alpha}\ln\left(1-\tfrac{\alpha}{2\overline{\delta}}\right) \\ &= \quad -\ell\ln\left(\overbrace{\left(1-\frac{\alpha}{2\delta}\right)^{\delta/\alpha}\cdot\left(1-\frac{\alpha}{2\overline{\delta}}\right)^{\overline{\delta}/\alpha}}^{\beta}\right). \end{aligned} \tag{3.62}$$

Now, if we let β be defined as in (3.62) then

$$\begin{aligned} m_2(m,\ell,w) \quad &= \quad \ell\ln\left(\tfrac{w}{(nR)^{1-y}}\right) \quad - \quad \tfrac{wn^y}{R^{1-y}}\cdot\tfrac{1}{\hat{a}} \quad -\ell\cdot\ln\left(\tfrac{\ell}{c_1\cdot n}\right) \quad - \quad \ell\ln\left(\beta\right) \\ &= \quad -\ell\ln\left(\tfrac{(nR)^{1-y}}{w}\cdot\tfrac{\ell}{c_1\cdot n}\cdot\beta\right) \quad - \quad \tfrac{wn^y}{R^{1-y}}\cdot\tfrac{1}{\hat{a}} \\ &= \quad n^y\cdot\left[-\tfrac{\ell}{n^y}\ln\left(\tfrac{\ell}{n^y}\cdot\overbrace{\frac{R^{1-y}}{wc_1}}^{b}\cdot\beta\right) \quad - \quad \tfrac{w}{R^{1-y}}\cdot\tfrac{1}{\hat{a}}\right]. \end{aligned}$$

From Lemma 3.33 we know that for any $x, b \in \mathbb{R}_{>0}$ we have

$$-x\ln(xb) \leq \frac{1}{be}.$$

Applying this with $x = \frac{\ell}{n^y}$ and $b = \frac{R^{1-y}}{wc_1} \cdot \beta$ we obtain

$$\begin{aligned} m_2(m,\ell,w) &= n^y \cdot \Big[-x\ln(xb) \quad - \quad \tfrac{w}{R^{1-y}} \cdot \tfrac{1}{\hat{a}} \Big] \\ &\leq n^y \cdot \Big[\tfrac{wc_1}{e \cdot R^{1-y}} \cdot \tfrac{1}{\beta} \quad - \quad \tfrac{w}{R^{1-y}} \cdot \tfrac{1}{\hat{a}} \Big] \\ &= n^y \cdot \Big[\tfrac{w}{R^{1-y}} \cdot \big(\tfrac{c_1}{e} \cdot \tfrac{1}{\beta} - \tfrac{1}{\hat{a}} \big) \Big]. \end{aligned} \tag{3.63}$$

Recall that β was defined as

$$\beta = \left(1 - \frac{\alpha}{2\delta}\right)^{\delta/\alpha} \cdot \left(1 - \frac{\alpha}{2\overline{\delta}}\right)^{\overline{\delta}/\alpha} = \left[\left(1 - \frac{\alpha}{2\delta}\right)^{2\delta/\alpha} \cdot \left(1 - \frac{\alpha}{2\overline{\delta}}\right)^{2\overline{\delta}/\alpha} \right]^{1/2}.$$

In general we have

$$\lim_{x \to 0} \big(1 - x\big)^{1/x} = \frac{1}{e},$$

and so applying this to our case with $x = \frac{\alpha}{2\delta}$ and then $x = \frac{\alpha}{2\overline{\delta}}$, we can deduce

$$\lim_{\alpha \to 0} \left[\left(1 - \frac{\alpha}{2\delta}\right)^{2\delta/\alpha} \cdot \left(1 - \frac{\alpha}{2\overline{\delta}}\right)^{2\overline{\delta}/\alpha} \right]^{1/2} = \left[\frac{1}{e} \cdot \frac{1}{e}\right]^{1/2} = \frac{1}{e},$$

and so

$$\lim_{\alpha \to 0} \ \frac{1}{\beta} \ = \ e.$$

We can write this formally by saying that for any $\epsilon_8 > 0$ there is $A_8 > 0$ for which

$$\alpha \leq A_8 \implies \left| \frac{1}{\beta} - e \right| \leq \epsilon_8 \implies \frac{1}{\beta} \leq e + \epsilon_8. \tag{3.64}$$

Going back to (3.63), we have

$$n^y \cdot \frac{w}{R^{1-y}} \cdot \frac{c_1}{e} > 0, \tag{3.65}$$

and therefore combining (3.65) and (3.64) we obtain

$$\alpha \leq A_8 \implies n^y \cdot \frac{w}{R^{1-y}} \cdot \frac{c_1}{e} \cdot \frac{1}{\beta} \leq n^y \cdot \frac{w}{R^{1-y}} \cdot \frac{c_1}{e} \cdot \big(e + \epsilon_8\big).$$

So using this with (3.63), we now have

$$\alpha \leq A_8 \implies m_2(n,\ell,w) \leq n^y \cdot \frac{w}{R^{1-y}} \cdot \left[\frac{c_1}{e} \cdot (e + \epsilon_8) - \frac{1}{\hat{a}} \right]. \tag{3.66}$$

We now show that if ϵ_8 is close enough to 0 and $\hat{a}$ close enough to 1, then the following term from (3.66)

$$\frac{c_1}{e} \cdot (e + \epsilon_8) - \frac{1}{\hat{a}} \tag{3.67}$$

is negative. Set

$$\epsilon_8 = \frac{1}{2} \cdot \frac{e}{c_1} \cdot (1 - c_1). \tag{3.68}$$

Since $c_1 < 1$ (see (3.61)), we have $\epsilon_8 > 0$. Next, recall (see Definition 3.26) that we had set the value of $\hat{a}$ to

$$\hat{a} \;=\; \left[2\sqrt{\delta\bar{\delta}} + \frac{3}{4}\big(1 - 2\sqrt{\delta\bar{\delta}}\big)\right]^{-1} \;=\; \left[c_1 + \frac{3}{4}\big(1 - c_1\big)\right]^{-1} \tag{3.69}$$

(and $c_1 = 2\sqrt{\delta\bar{\delta}}$ by definition). It now becomes clear why this value was chosen for $\hat{a}$. Plugging (3.68) and (3.69) into (3.67) we deduce that

$$\begin{aligned}
\tfrac{c_1}{e} \cdot (e + \epsilon_8) - \tfrac{1}{\hat{a}} &= \tfrac{c_1}{e} \cdot \left(e + \tfrac{1}{2} \cdot \tfrac{e}{c_1} \cdot (1 - c_1)\right) - \left(c_1 + \tfrac{3}{4}\big(1 - c_1\big)\right) \\
&= c_1 + \tfrac{1}{2} \cdot (1 - c_1) \;\; - \;\; c_1 - \tfrac{3}{4}\big(1 - c_1\big) \\
&= -\tfrac{1}{4} \cdot (1 - c_1) \\
&< 0.
\end{aligned}$$

Therefore setting

$$\tau_7 = -\frac{1}{R^{1-y}} \cdot \left[\frac{c_1}{e} \cdot (e + \epsilon) - \frac{1}{a}\right],$$

we have $\tau_7 > 0$, and (3.66) leads to

$$\alpha \le A_8 \implies m_2(n, \ell, w) \le n^y \cdot \frac{w}{R^{1-y}} \cdot \left[\frac{c_1}{e} \cdot (e + \epsilon_8) - \frac{1}{a}\right] = -\tau_7 \cdot w \cdot n^y. \tag{3.70}$$

Setting $A_7 = A_8$ gives us the required result. ■

Lemma 3.35. *There are $N_9, A_9, \Gamma_9, \tau_9 > 0$ with*

$$n \ge N_9, \alpha \le A_9, \gamma \le \Gamma_9 \implies m_1(n, \ell, w) \le -\tau_9 \cdot n^y. \tag{3.71}$$

Proof: Recall that we had a bound on $m_1(n, \ell, w)$ consisting of eight terms $t_1, \ldots, t_8$, see (3.58). We then chose six of these terms to make up $m_2(n, \ell, w)$, see (3.59). In Lemma 3.34 we showed that there was some $\tau_7 > 0$ for which

$$m_2(n, \ell, w) \le -\tau_7 \cdot w \cdot n^y \tag{3.72}$$

for α small enough. We fix τ_7 so be some value that satisfies (3.72).

In this proof we will show that the two remaining terms of $m_1(n, \ell, w)$ (namely t_1 and t_4) are dominated by (3.72) as n gets large. More formally we will show that for each of these terms t_i, given any ϵ there are N, A, Γ for which

$$n \ge N, \alpha \le A, \gamma \le \Gamma \implies \frac{t_i}{\tau_7 \cdot w \cdot n^y} \le \epsilon.$$

• **1)** t_1 is dominated by $-\tau_7 \cdot w \cdot n^y$.
First recall that

$$t_1 = -\frac{w \ln\left(\frac{w}{nR}\right)}{\ln(2)} - \frac{w}{\ln(2)}.$$

So using the definition $\gamma = \frac{w}{n^{1-y}}$ we have

$$\frac{t_1}{\tau_7 \cdot w \cdot n^y} = \frac{1}{\tau_7 \cdot \ln(2)} \cdot \left[-\frac{\ln\left(\frac{w}{nR}\right)}{n^y} - \frac{1}{n^y} \right] = \frac{1}{\tau_7 \cdot \ln(2)} \cdot \left[-\frac{\ln\left(\frac{\gamma}{Rn^y}\right)}{n^y} - \frac{1}{n^y} \right],$$

which we can write as

$$\frac{t_1}{\tau_7 \cdot w \cdot n^y} = \frac{1}{\tau_7 \cdot \ln(2)} \cdot \left[-\frac{\ln\left(\frac{\gamma}{R}\right)}{n^y} + \frac{\ln(n^y)}{n^y} - \frac{1}{n^y} \right]. \tag{3.73}$$

Recall that R and y are fixed, and that

$$\lim_{x \to \infty} \frac{\ln(x)}{x} = 0.$$

So setting $x = n^y$ we see that if γ is upper bounded then (3.73) will tend to zero as n gets large. Formally, for any $\epsilon_{10} > 0$ there are N_{10}, Γ_{10} with

$$n \geq N_{10}, \gamma \leq \Gamma_{10} \implies \frac{t_1}{\tau_7 \cdot w \cdot n^y} < \epsilon_{10}. \tag{3.74}$$

• **2)** t_4 is dominated by $-\tau_7 \cdot w \cdot n^y$.
We start by recalling that

$$t_4 = -\ell \ln\left(1 - \frac{w}{(nR)^{1-y}}\right).$$

Using the definitions $\alpha = \frac{\ell}{n}$ and $\gamma = \frac{w}{n^{1-y}}$ we obtain

$$\frac{t_4}{\tau_7 \cdot w \cdot n^y} = -\frac{\ell \ln\left(1 - \frac{w}{(nR)^{1-y}}\right)}{\tau_7 \cdot w \cdot n^y} = -\frac{\alpha \cdot n^{1-y} \cdot \ln\left(1 - \frac{\gamma}{R^{1-y}}\right)}{\tau_7 \cdot w} = -\frac{\alpha}{\tau_7} \cdot \frac{\ln\left(1 - \frac{\gamma}{R^{1-y}}\right)}{\gamma}. \tag{3.75}$$

Since $\frac{\ln(1-x)}{x} \to -1$ when x tends to zero, by making both α and γ small enough we can bring (3.75) as close to zero as we need. Formally, for any $\epsilon_{12} > 0$, there are A_{12}, Γ_{12} with

$$\alpha \leq A_{12}, \gamma \leq \Gamma_{12} \implies \frac{t_4}{\tau_7 \cdot w \cdot n^y} \leq \epsilon_{12}. \tag{3.76}$$

• **3)** Combining it all.
We have

$$m_1(n, \ell, w) \leq m_2(n, \ell, w) + t_1 + t_4.$$

From Lemma 3.34 we know that there is A_7 for which $\alpha \leq A_7$ implies

$$\begin{aligned} m_1(n, \ell, w) &\leq -\tau_7 \cdot w \cdot n^y + t_1 + t_4 \\ &= -\tau_7 \cdot w \cdot n^y \cdot \left(1 + \frac{t_1}{-\tau_7 \cdot w \cdot n^y} + \frac{t_4}{-\tau_7 \cdot w \cdot n^y}\right). \end{aligned} \tag{3.77}$$

Now let $\epsilon_{10} = \epsilon_{12} = \frac{1}{3}$ (in fact anything $< \frac{1}{2}$ would do). Pick N_{10}, Γ_{10} from (3.74), A_{12}, Γ_{12} from (3.76). Now set

$$\begin{array}{rcll} N_9 & = & N_{10}, & \\ \Gamma_9 & = & \min(\Gamma_{10}, \Gamma_{12}), & \\ A_9 & = & A_{12}, & \\ \epsilon_9 & = & 1/3 & (= \epsilon_{10} = \epsilon_{12}). \end{array}$$

Combining (3.74), (3.76) and (3.77), we can deduce that if $n \geq N_9, \alpha \leq A_9, \gamma \leq \Gamma_9$ then

$$m_1(n, \ell, w) \leq -\tau_7 \cdot w \cdot n^y \cdot (1 - \epsilon_9 - \epsilon_9) = -\frac{\tau_7 \cdot w}{3} \cdot n^y \leq -\frac{\tau_7}{3} \cdot n^y,$$

where the last inequality holds because $w \geq 1$. So setting $\tau_9 = \frac{\tau_7}{3}$ gives us the required result. ■

Definition 3.36. Let $\hat{A}$ be a value for A_9 that satisfies (3.71) (we know that such a value exists by Lemma 3.35).

The result of Case 2 is summarized in the following theorem:

Theorem 3.37. *Suppose we are given $0 < R < 1$ and $0 < \delta < \frac{1}{2}$. Let $\hat{A}$ be taken from Definition 3.36. Then there are $N_9, \Gamma_9, \tau_9 > 0$ (depending only on R, δ and y) for which*

$$n \geq N_9 \implies \forall \ell = 1, \ldots, \left\lfloor \hat{A} n \right\rfloor, \ \forall w = 1, \ldots, \left\lfloor \Gamma_9 \cdot n^{1-y} \right\rfloor : \ m(n, \ell, w) \leq \exp\big(- \tau_9 \cdot n^y \big).$$

3.4.6 Case 3: Small γ, $\alpha \geq \hat{A}$.

We will make use of some of the work done in Case 2. Let $\hat{a}$ and $\hat{\Gamma_a}$ be taken from Definitions 3.26 and 3.28. We will assume throughout this section (Case 3) that $\gamma \leq \hat{\Gamma_a}$. Recall from (3.25) that α and γ are defined as

$$\alpha = \frac{\ell}{n}, \qquad \gamma = \frac{w}{n^{1-y}}. \tag{3.78}$$

Propositions 3.30, 3.31 and 3.32 still hold. We rewrite them below after some algebraic manipulations:

$$\begin{array}{rcl} s_1(n, w) & \leq & n \cdot \Big[\overbrace{-\frac{\gamma}{n^y} \cdot \ln\left(\frac{\gamma}{R n^y}\right)}^{u_1} \quad \overbrace{-R \ln\left(1 - \frac{\gamma}{R n^y}\right)}^{u_2} \Big]. \\ f_1(n, \ell, w) & \leq & n \cdot \Big[\overbrace{\alpha \ln\left(\frac{\gamma}{R^{1-y}}\right)}^{u_3} \quad \overbrace{-\alpha \ln\left(1 - \frac{\gamma}{R^{1-y}}\right)}^{u_4} \quad \overbrace{-\frac{\gamma}{R^{1-y}} \cdot \frac{1}{a}}^{u_5} \Big]. \\ g_1(n, \ell) & \leq & n \cdot \Big[\overbrace{\frac{\alpha}{2} \ln\left(\frac{2\delta}{\alpha} - 1\right) - \delta \ln\left(1 - \frac{\alpha}{2\delta}\right) + \frac{\alpha}{2} \ln\left(\frac{2\overline{\delta}}{\alpha} - 1\right) - \overline{\delta} \ln\left(1 - \frac{\alpha}{2\overline{\delta}}\right)}^{u_6} \Big]. \end{array} \tag{3.79}$$

Outline of Case 3: Intuitively, when $\hat{A} \leq \alpha \leq 1$ and $\gamma \to 0$, the terms in (3.79) behave as follows: u_1, u_2, u_4 and u_5 tend to zero, u_6 is upper bounded by some positive value that depends on $\hat{A}$, and $u_3 \to -\infty$. Therefore the sum of all u_i's will tend to $-\infty$, and so we can certainly upper bound it by $-\tau$ for some $\tau > 0$ (any value will do). This means that we can upper bound $m_1(n, \ell, w)$ by

$$-\tau \cdot n,$$

with $\tau > 0$. This is actually a stronger statement than is required (we need only $-\tau \cdot n^y$).

Proposition 3.38. *If $\alpha \geq \hat{A}$ then there is a value $\hat{c_A} \geq 0$ depending only on δ and $\hat{A}$ for which*

$$g_1(n, \ell) \leq \hat{c_A} \cdot n.$$

Proof: We know that

$$g_1(n, \ell) \leq n \cdot u_6(\alpha),$$

where

$$u_6(\alpha) = \frac{\alpha}{2} \ln\left(\frac{2\delta}{\alpha} - 1\right) - \delta \ln\left(1 - \frac{\alpha}{2\delta}\right) + \frac{\alpha}{2} \ln\left(\frac{2\overline{\delta}}{\alpha} - 1\right) - \overline{\delta} \ln\left(1 - \frac{\alpha}{2\overline{\delta}}\right).$$

Now because $0 < A \leq \alpha \leq 2\delta < 1$, we study the function $u_6(\alpha)$ over the range $I_\alpha = [\hat{A}, 2\delta]$. We note that $u_6(\alpha)$ is differentiable and therefore continuous over I_α. So $u_6(\alpha)$ is a continuous real function over a closed bounded interval, it is therefore bounded. In particular there exists an upper bound c_2 (depending only on δ and $\hat{A}$). We set $\hat{c_A} = \max(c_2, 0)$ (to ensure that $\hat{c_A} \geq 0$) and obtain

$$g_1(n, \ell) \leq u_6(\alpha) \cdot n \leq \hat{c_A} \cdot n.$$

■

Lemma 3.39. *Let $u_3 = \alpha \ln(\frac{\gamma}{R^{1-y}})$ be taken from (3.79). If $\alpha \geq \hat{A}$ then for any $\tau_{14} > 0$ there is Γ_{14} with*

$$\gamma \leq \Gamma_{14} \implies u_3 \leq -\tau_{14}.$$

Proof: First notice that if $\gamma < R^{1-y}$ then $\ln(\frac{\gamma}{R^{1-y}}) < 0$. Therefore

$$\alpha \geq \hat{A}, \gamma < R^{1-y} \implies \alpha \cdot \ln\left(\frac{\gamma}{R^{1-y}}\right) \leq \hat{A} \cdot \ln\left(\frac{\gamma}{R^{1-y}}\right), \tag{3.80}$$

and

$$\lim_{\gamma \to 0} \hat{A} \cdot \ln\left(\frac{\gamma}{R^{1-y}}\right) = -\infty.$$

So formally for any $\tau_{15} > 0$ there is $\Gamma_{15} > 0$ with

$$\gamma \leq \Gamma_{15} \implies \hat{A} \cdot \ln\left(\frac{\gamma}{R^{1-y}}\right) \leq -\tau_{15}. \tag{3.81}$$

So we set $\tau_{15} = \tau_{14}$, take some value Γ_{15} that satisfies (3.81). Letting $\Gamma_{14} = \min\left(\Gamma_{15}, R^{1-y}\right)$, and combining this with (3.80) we obtain:

$$\alpha \geq \hat{A}, \gamma \leq \Gamma_{14} \implies \alpha \cdot \ln\left(\frac{\gamma}{R^{1-y}}\right) \leq \hat{A} \cdot \ln\left(\frac{\gamma}{R^{1-y}}\right) \leq -\tau_{14},$$

as required. ■

We will show that the terms u_1, u_2, u_4 and u_5 in (3.79) are dominated by $u_3 = \alpha \cdot \ln\left(\frac{\gamma}{R^{1-y}}\right)$.

Proposition 3.40. *If $\alpha \geq \hat{A}$ then for any $\epsilon_{16} > 0$ there are $N_{16}, \Gamma_{16} > 0$ with*

$$n \geq N_{16}, \gamma \leq \Gamma_{16} \implies s_1(n,\ell) + f_1(n,\ell,w) \leq n \cdot \big[u_3 + \epsilon_{16}\big].$$

Proof: We know from (3.79) that

$$s_1(n,\ell) + f_1(n,\ell,w) \leq n \cdot \big[u_1 + u_2 + u_3 + u_4 + u_5\big]. \tag{3.82}$$

Recalling that

$$\lim_{x \to 0} x \cdot \ln(x) = 0,$$

we can deduce (by setting $x = \frac{\gamma}{n^y}$) that as γ gets small and n gets large,

$$u_1 = -\frac{\gamma}{n^y} \cdot \ln\left(\frac{\gamma}{Rn^y}\right)$$

will tend to zero. Similarly, using the fact that

$$\lim_{x \to 0} \ln(1-x) = 0,$$

we can show that as γ gets small and n gets large, u_2, u_4 and u_5 all tend to zero. So formally this means that given any $\epsilon_{16} > 0$, there are $N_{16}, \Gamma_{16} > 0$ with

$$n \geq N_{16}, \gamma \leq \Gamma_{16} \implies u_1 + u_2 + u_4 + u_5 \leq \epsilon_{16}.$$

The required statement then follows immediately. ■

We can now combine all this to obtain the following:

Proposition 3.41. *If $\alpha \geq \hat{A}$ there are $N_{21}, \Gamma_{21}, \tau_{21} > 0$ with*

$$n \geq N_{21}, \gamma \leq \Gamma_{21} \implies m_1(n,\ell,w) \leq -\tau_{21} \cdot n^y. \tag{3.83}$$

Proof: We are supposing throughout this proof that $\alpha \geq \hat{A}$. Recall that

$$m_1(n,\ell,w) = s_1(n,w) + f_1(n,\ell,w) + g_1(n,\ell).$$

We first set $\epsilon_{16} = \frac{1}{2}$ in Proposition 3.40, and get values N_{16} and Γ_{16} with

$$n \geq N_{16}, \gamma \leq \Gamma_{16} \implies s_1(n,\ell) + f_1(n,\ell,w) \leq n \cdot \big[u_3 + \frac{1}{2}\big]. \tag{3.84}$$

Next, we know from Proposition 3.38 that there is some value $\hat{c_A} \geq 0$ (depending only on R and $\hat{A}$) with

$$g_1(n,\ell) \leq \hat{c_A} \cdot n. \tag{3.85}$$

We now set $\tau_{14} = \hat{c_A} + 1$ in Lemma 3.39. This gives us some value Γ_{14} with

$$\gamma \le \Gamma_{14} \implies u_3 \le -(\hat{c_A} + 1). \tag{3.86}$$

So setting $N_{21} = N_{16}$ and $\Gamma_{21} = \min(\Gamma_{14}, \Gamma_{14})$ we combine (3.84), (3.85) and (3.86) to deduce that if $\alpha \le \hat{A}$ then

$$\begin{aligned} n \ge N_{21}, \gamma \le \Gamma_{21} \implies m_1(n,\ell,w) &\le \underbrace{s_1(n,w) + f_1(n,\ell,w)}_{} + \underbrace{g_1(n,\ell)}_{} \\ &\le \quad n \cdot \big[u_3 + \tfrac{1}{2}\big] \quad + \quad n \cdot \hat{c_A} \\ &= \quad n \cdot \big[u_3 + \tfrac{1}{2} + \hat{c_A}\big] \\ &\le \quad n \cdot \big[-(\hat{c_A} + 1) + \tfrac{1}{2} + \hat{c_A}\big] \\ &= \quad n \cdot \big[-\tfrac{1}{2}\big], \end{aligned}$$

so setting $\tau_{21} = \frac{1}{2}$ we have shown a stronger statement than the required result. Indeed because $\tau_{21} > 0$ and $0 < y < 1$, we have $-\tau_{21} \cdot n \le -\tau_{21} \cdot n^y$. ■

Definition 3.42. Let Γ_9 and τ_9 be values that satisfy (3.71) (such values exists by Lemma 3.35), in Case 2. Let Γ_{21} and τ_{21} be values that satisfy (3.83) (such values exist by Proposition 3.41). We define $\hat{\Gamma_\ell}$ as

$$\hat{\Gamma_\ell} = \min\big(\Gamma_9, \Gamma_{21}\big).$$

Our three constants $\hat{A}, \hat{\Gamma_\ell}$ and $\hat{\Gamma_u}$ have now all been defined. Once more, these values depend only on R, δ and y. Letting $\tau_{22} = \min(\tau_9, \tau_{21})$ and $N_{22} = \max(N_9, N_{21})$, we summarize the result for Cases 2 and 3 below:

Theorem 3.43. *Suppose we are given $0 < R < 1$ and $0 < \delta < \frac{1}{2}$. Let $\hat{\Gamma_\ell}$ be taken from Definition 3.42. Then there are $N_{22}, \tau_{22} > 0$ (depending only on R, δ and y) for which*

$$n \ge N_{22} \implies \forall \ell = 1, \dots, \lfloor 2\delta n \rfloor\,,\ \forall w = 1, \dots, \Big\lfloor \hat{\Gamma_\ell} \cdot n^{1-y} \Big\rfloor\ :\ m(n,\ell,w) \le \exp\big(-\tau_{22} \cdot n^y\big).$$

3.4.7 Case 4: Any α, $\hat{\Gamma_\ell} \le \gamma \le \hat{\Gamma_u}$.

Outline: We will first show that for n large enough we have $f_1(n,\ell,w) + g_1(n,\ell) \le n \cdot [v + \epsilon]$, where v is some function of α and γ, and ϵ can be made as small as necessary. Then, we will show that there is some $\tau > 0$ for which $v \le -\tau$ (for all values α, γ we are considering in Case 4). Finally we will show that $\frac{s_1(n,w)}{n}$ tends to zero when n gets large, and therefore is dominated by $f_1(n,\ell,w) + g_1(n,\ell)$.

We start by giving a reminder of the definitions of α and γ:

$$\alpha = \frac{\ell}{n} \quad , \qquad\qquad \gamma = \frac{w}{n^{1-y}}.$$

Recall that P_w was defined in (3.18) as follows:

$$P_w = \frac{1}{2} - \frac{1}{2} \cdot \left(1 - \frac{2w}{nR}\right)^{(nR)^y}.$$

Definition 3.44. We define β as

$$\beta = \exp\Big(-\frac{2\gamma}{R^{1-y}}\Big).$$

Notice that β depends on γ, and therefore on n. We are assuming in this section that $\hat{\Gamma_\ell} \leq \gamma \leq \hat{\Gamma_u}$. ($\hat{\Gamma_u}$ and $\hat{\Gamma_\ell}$ are constants depending only on R, δ and y taken from Definitions 3.20 and 3.42). So because $\exp(-x)$ is a decreasing function we have

$$\overbrace{\exp\big(-\frac{2\hat{\Gamma_u}}{R^{1-y}}\big)}^{\hat{B_1}} \leq \beta \leq \overbrace{\exp\big(-\frac{2\hat{\Gamma_\ell}}{R^{1-y}}\big)}^{\hat{B_2}}. \tag{3.87}$$

Furthermore notice that since $\frac{2\hat{\Gamma_\ell}}{R^{1-y}}, \frac{2\hat{\Gamma_u}}{R^{1-y}} > 0$, we have

$$0 < \hat{B_1}, \hat{B_2} < 1. \tag{3.88}$$

We know that for any constant $c \in \mathbb{R}$,

$$\lim_{x\to\infty} \left(1 - \frac{c}{x}\right)^x = \exp(-c).$$

The following lemma essentially states that if c depends on x but is *bounded* then we have an equivalent result:

Lemma 3.45. *Let $f : \mathbb{R} \to \mathbb{R}$ be a bounded function. Then for any $\epsilon > 0$ there is X with*

$$x \geq X \implies \exp\big(-f(x)\big) - \epsilon \leq \left(1 - \frac{f(x)}{x}\right)^x \leq \exp\big(-f(x)\big) + \epsilon.$$

Proof: See Appendix C. ■

Intuitively the fact that $\gamma \in \big[\hat{\Gamma_\ell},\ \hat{\Gamma_u}\big]$ (in Case 4) enables us to treat γ like a constant. More precisely we use the fact that it can get neither arbitrarily large nor arbitrarily close to zero as n gets large. From Lemma 3.45 we deduce the following:

Corollary 3.46. *Let β be taken from Definition 3.44. If $\gamma \leq \hat{\Gamma_u}$ (which we assume throughout case 4), then for any $\epsilon > 0$ there is N with*

$$n \geq N \implies \frac{1}{2} \cdot (1 - \beta) - \epsilon \leq P_w \leq \frac{1}{2} \cdot (1 - \beta) + \epsilon. \tag{3.89}$$

Proof: Recall that

$$P_w \;=\; \frac{1}{2} - \frac{1}{2}\cdot\left(1-\frac{2w}{nR}\right)^{(nR)^y} \;=\; \frac{1}{2} - \frac{1}{2}\cdot\left(1-\frac{2\gamma/R^{1-y}}{(nR)^y}\right)^{(nR)^y}. \tag{3.90}$$

So applying Lemma 3.45 with $x = (nR)^y$ and $f(x) = \frac{2\gamma}{R^{1-y}}$ (γ depends on n) leads to the required result, since $\beta = \exp\big(-f(x)\big)$. ■

Now recall that $f(n,\ell,w)$ was defined in (3.21) as

$$f(n,l,w) = (P_w)^\ell \cdot (1-P_w)^{n-\ell}, \tag{3.91}$$

and since $f_1(n,\ell,w) = \ln\big(f(n,\ell,w)\big)$ we obtain

$$f_1(n,\ell,w) = \ell\ln\big(P_w\big) + \big(n-\ell\big)\cdot\ln\big(1-P_w\big). \tag{3.92}$$

We now give an upper bound on $f_1(n,\ell,w)$:

Proposition 3.47. *For any $\epsilon_{23} > 0$, there is $N_{23} > 0$ with*

$$n \geq N_{23} \implies f_1(n,\ell,w) \leq n\cdot\left[-\ln(2) + \alpha\ln\left(\frac{1-\beta}{1+\beta}\right) + \ln\big(1+\beta\big) + \epsilon_{23}\right]. \tag{3.93}$$

Proof: From Corollary 3.46 we have that for any $\epsilon_{23} > 0$, there is N_{23} with

$$n \geq N_{23} \implies \ln(P_w) \leq \ln\left(\frac{1}{2}\cdot(1-\beta)\right) + \epsilon_{23}, \tag{3.94}$$

and

$$n \geq N_{23} \implies \ln(1-P_w) \leq \ln\left(\frac{1}{2}\cdot(1+\beta)\right) + \epsilon_{23}. \tag{3.95}$$

Therefore because $\ell = \alpha\cdot n$, $n \geq N_{23}$ implies that

$$\begin{aligned}
f_1(n,l,w) \;&=\; \ell\ln\big(P_w\big) + \big(n-\ell\big)\cdot\ln\big(1-P_w\big) \\
&\leq\; \ell\cdot\left(\ln\big(\tfrac{1}{2}(1-\beta)\big) + \epsilon_{23}\right) \;+\; \big(n-\ell\big)\cdot\ln\left(\tfrac{1}{2}(1+\beta) + \epsilon_{23}\right) \qquad \text{(using (3.94) and (3.95))} \\
&\leq\; n\cdot\Big[-\alpha\ln(2) + \alpha\ln(1-\beta) + \alpha\cdot\epsilon_{23} \\
&\qquad\qquad -\ln(2) + \ln(1+\beta) + \alpha\ln(2) - \alpha\ln(1+\beta) + (1-\alpha)\cdot\epsilon_{23}\Big] \\
&=\; n\cdot\left[-\ln(2) + \alpha\ln\left(\tfrac{1-\beta}{1+\beta}\right) + \ln\big(1+\beta\big) + \epsilon_{23}\right],
\end{aligned}$$

as required. ■

Now, we define

$$c = \frac{1+\beta}{1-\beta}, \tag{3.96}$$

so that the bound in (3.93) can be written as

$$n \geq N_{23} \implies f_1(n,\ell,w) \leq n \cdot \left[-\ln(2) + \alpha \ln\left(\frac{1}{c}\right) \ln\left(\frac{2c}{c+1}\right) + \epsilon_{23} \right]. \tag{3.97}$$

Notice that because $\frac{1+x}{1-x}$ is a strictly increasing function, if $0 < \hat{B}_1 \leq \beta \leq \hat{B}_2 < 1$ then

$$1 < \overbrace{\left(\frac{1+\hat{B}_1}{1-\hat{B}_1}\right)}^{\hat{C}_1} \leq c \leq \overbrace{\left(\frac{1+\hat{B}_2}{1-\hat{B}_2}\right)}^{\hat{C}_2}. \tag{3.98}$$

Next, the bound on $g_1(n,w)$ from Proposition 3.32 still holds, we rewrite it below (after some algebraic manipulations, see (3.79)):

$$g_1(n,\ell) \leq n \cdot \left[\frac{\alpha}{2} \ln\left(\frac{2\delta}{\alpha} - 1\right) - \delta \ln\left(1 - \frac{\alpha}{2\delta}\right) + \frac{\alpha}{2} \ln\left(\frac{2\overline{\delta}}{\alpha} - 1\right) - \overline{\delta} \ln\left(1 - \frac{\alpha}{2\overline{\delta}}\right) \right]. \tag{3.99}$$

We combine (3.97) and (3.99) to obtain the following definition:

Definition 3.48. Let $v(\alpha, c)$ be the following function:

$$\begin{aligned} v(\alpha,c) \quad &= \quad -\ln(2) + \alpha \ln(\tfrac{1}{c}) + \ln(\tfrac{2c}{c+1}) \\ &\quad + \tfrac{\alpha}{2} \ln(\tfrac{2\delta}{\alpha} - 1) - \delta \ln(1 - \tfrac{\alpha}{2\delta}) + \tfrac{\alpha}{2} \ln(\tfrac{2\overline{\delta}}{\alpha} - 1) - \overline{\delta} \ln(1 - \tfrac{\alpha}{2\overline{\delta}}). \end{aligned} \tag{3.100}$$

(δ is a parameter with $0 < \delta < \frac{1}{2}$, and $\overline{\delta} = 1 - \delta$).

So using this with Proposition 3.47, we see that for any $\epsilon_{23} > 0$, there is N_{23} with

$$n \geq N_{23} \implies f_1(n,\ell,w) + g_1(n,\ell) \leq n \cdot \big[v(\alpha,c) + \epsilon_{23} \big]. \tag{3.101}$$

Proposition 3.49. *There is $\tau_{24} > 0$ (depending only on R, δ and y) for which for any $0 < \alpha < 1$ and $\hat{C}_1 \leq c \leq \hat{C}_2$ we have*

$$v(\alpha,c) \leq -\tau_{24}.$$

Proof: We will proceed by carefully analyzing the function $v(\alpha, c)$. We divide the proof into steps:

• **1)** For fixed c we find which α maximizes $v(\alpha, c)$.
We start by differentiating $v(\alpha, c)$ with respect to α. We define

$$v'(\alpha,c) = \frac{\partial}{\partial \alpha} v(\alpha,c).$$

This gives us

$$\begin{aligned} v'(\alpha,c) &= \ln\left(\tfrac{1}{c}\right)+\tfrac{1}{2}\ln\left(\tfrac{2\delta}{\alpha}-1\right)+\tfrac{\alpha}{2}\left[\tfrac{2\delta}{\alpha}-1\right]^{-1}\left(-\tfrac{2\delta}{\alpha^2}\right)-\delta\left[1-\tfrac{\alpha}{2\delta}\right]^{-1}\left(-\tfrac{1}{2\delta}\right) \\ &\quad +\tfrac{1}{2}\ln\left(\tfrac{2\overline{\delta}}{\alpha}-1\right)+\tfrac{\alpha}{2}\left[\tfrac{2\overline{\delta}}{\alpha}-1\right]^{-1}\left(-\tfrac{2\overline{\delta}}{\alpha^2}\right)-\overline{\delta}\left[1-\tfrac{\alpha}{2\overline{\delta}}\right]^{-1}\left(-\tfrac{1}{2\overline{\delta}}\right) \\ &= \ln\left(\tfrac{1}{c}\right)+\tfrac{1}{2}\ln\left(\left(\tfrac{2\delta}{\alpha}-1\right)\cdot\left(\tfrac{2\overline{\delta}}{\alpha}-1\right)\right)-\tfrac{\delta}{2\delta-\alpha}+\tfrac{\delta}{2\delta-\alpha}-\tfrac{\overline{\delta}}{2\overline{\delta}-\alpha}+\tfrac{\overline{\delta}}{2\overline{\delta}-\alpha} \\ &= \ln\left(\tfrac{1}{c}\right)+\tfrac{1}{2}\ln\left(\tfrac{4\delta\overline{\delta}}{\alpha^2}-\tfrac{2}{\alpha}+1\right). \end{aligned}$$

Now,

$$\begin{aligned} v'(\alpha,c)=0 &\iff \ln\left(\tfrac{1}{c}\right)+\tfrac{1}{2}\ln\left(\tfrac{4\delta\overline{\delta}}{\alpha^2}-\tfrac{2}{\alpha}+1\right)=0 \\ &\iff \ln\left(\tfrac{4\delta\overline{\delta}}{\alpha^2}-\tfrac{2}{\alpha}+1\right)=\ln\left(c^2\right) \\ &\iff \tfrac{4\delta\overline{\delta}}{\alpha^2}-\tfrac{2}{\alpha}+1=c^2 \\ &\iff (1-c^2)\cdot\alpha^2-2\cdot\alpha+4\delta\overline{\delta}=0. \end{aligned}$$

We solve this quadratic equation in α to obtain

$$v'(\alpha,c)=0 \iff \alpha=\frac{1\pm\sqrt{1-4\delta\overline{\delta}\cdot(1-c^2)}}{1-c^2}=\frac{1\pm\sqrt{1+4\delta\overline{\delta}\cdot(c^2-1)}}{1-c^2}.$$

Clearly we have $1+\sqrt{1+4\delta\overline{\delta}\cdot(c^2-1)}>0$ and $1-c^2<0$ (since $c>1$). This means that the first solution

$$\alpha_1=\frac{1+\sqrt{1+4\delta\overline{\delta}\cdot(c^2-1)}}{1-c^2} \tag{3.102}$$

is negative. So since we are considering the range $0<\alpha<2\delta$, the only extremal point we need to look at is the other solution

$$\alpha_2=\frac{1-\sqrt{1+4\delta\overline{\delta}\cdot(c^2-1)}}{1-c^2}=\frac{\sqrt{1+4\delta\overline{\delta}\cdot(c^2-1)}-1}{c^2-1}.$$

We write this as a function of c, so we define

$$u(c)=\frac{\sqrt{1+4\delta\overline{\delta}\cdot(c^2-1)}-1}{c^2-1}. \tag{3.103}$$

Now because $v'(\alpha,c)$ is continuous for $\alpha\in]0,2\delta[$, and

$$\lim_{\alpha\to 0}v'(\alpha,c)=\infty, \qquad \lim_{\alpha\to 2\delta}v'(\alpha,c)=-\infty,$$

we can deduce that $v'(\alpha, c) > 0$ when $\alpha < u(c)$ and $v'(\alpha, c) < 0$ (otherwise $\alpha = u(c)$ is the only zero of $v'(\alpha, c)$). Therefore $\alpha = u(c)$ is a maximal point of $v(\alpha, c)$. So if we let

$$t(c) = v\big(u(c), c\big),$$

then for any α, c with $0 < \alpha < 2\delta$ and $\hat{C}_1 \le c \le \hat{C}_2$, we have:

$$v(\alpha, c) \le t(c). \tag{3.104}$$

So we can achieve our goal by upper bounding $t(c)$. We are considering values of c in the range $1 < \hat{C}_1 \le c \le \hat{C}_2$ (see (3.98)). Our strategy is to show that t is strictly increasing, and that it tends to zero as c gets large, and therefore that $t(c) \le t(\hat{C}_2) < 0$, so $-\tau_{24} = t(\hat{C}_2)$ will be a suitable value (see Figure 3.1).

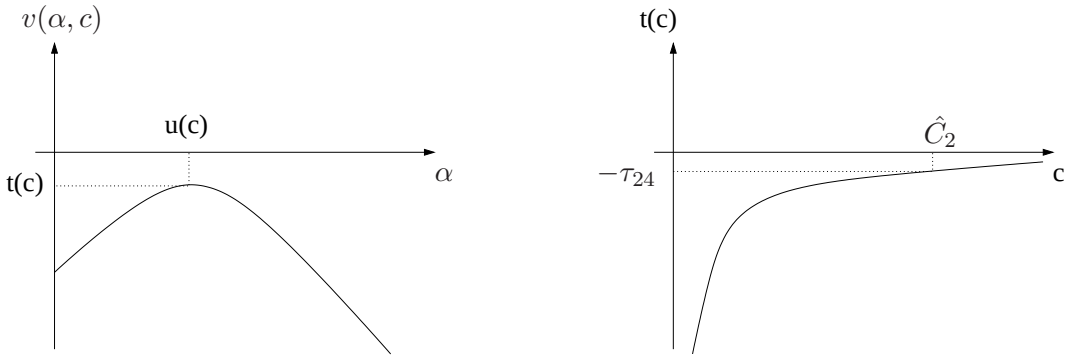


Figure 3.1: $v(\alpha, c)$ and $t(c)$.

• **2)** We show that $t(c)$ is strictly increasing for $c > 1$.
First of all, using the definition of $v(\alpha, c)$, we have:

$$\begin{aligned}
t(c) &= v\big(u(c), c\big) \\
&= -\ln(2) + u(c)\ln\big(\tfrac{1}{c}\big) + \ln\big(\tfrac{2c}{c+1}\big) \\
&\quad + \tfrac{u(c)}{2}\ln\big(\tfrac{2\delta}{u(c)} - 1\big) - \delta\ln\big(1 - \tfrac{u(c)}{2\delta}\big) + \tfrac{u(c)}{2}\ln\big(\tfrac{2\overline{\delta}}{u(c)} - 1\big) - \overline{\delta}\ln\big(1 - \tfrac{u(c)}{2\overline{\delta}}\big) \\
&= \ln\big(\tfrac{c}{c+1}\big) - \delta\ln\big(1 - \tfrac{u(c)}{2\delta}\big) - \overline{\delta}\ln\big(1 - \tfrac{u(c)}{2\overline{\delta}}\big) + u(c)\cdot\overbrace{\Big[\ln\Big(\frac{1}{c}\Big) + \frac{1}{2}\ln\Big(\big(\frac{2\delta}{u(c)} - 1\big)\big(\frac{2\overline{\delta}}{u(c)} - 1\big)\Big)\Big]}^{X}.
\end{aligned}$$

We will now show that $X = 0$. Recall from (3.103) that $u(c)$ was defined as

$$u(c) = \frac{\sqrt{V} - 1}{c^2 - 1}, \qquad \text{where } V = 1 + 4\delta\overline{\delta}\cdot(c^2 - 1). \tag{3.105}$$

We start with the conclusion we are trying to reach and use equivalences all along:

$$\begin{aligned}
X=0 \quad &\Longleftrightarrow \quad \ln\left(\tfrac{1}{c}\right)+\tfrac{1}{2}\ln\left(\tfrac{4\delta\overline{\delta}}{u^2(c)}-\tfrac{2\delta+2\overline{\delta}}{u(c)}+1\right)=0\\
&\Longleftrightarrow \quad \tfrac{4\delta\overline{\delta}}{u^2(c)}-\tfrac{2}{u(c)}+1=c^2\\
&\Longleftrightarrow \quad \tfrac{4\delta\overline{\delta}(c^2-1)^2}{(\sqrt{V}-1)^2}-\tfrac{2(c^2-1)}{\sqrt{V}-1}=c^2-1 \qquad \text{(from (3.105))}\\
&\Longleftrightarrow \quad \tfrac{4\delta\overline{\delta}(c^2-1)}{(\sqrt{V}-1)^2}-\tfrac{2}{\sqrt{V}-1}=1 \qquad \text{(we can divide by } c^2-1 \text{ because } c>1)\\
&\Longleftrightarrow \quad 4\delta\overline{\delta}(c^2-1)-2\sqrt{V}+2=(\sqrt{V}-1)^2\\
&\Longleftrightarrow \quad 4\delta\overline{\delta}(c^2-1)-2\sqrt{V}+2=V-2\sqrt{V}+1\\
&\Longleftrightarrow \quad 4\delta\overline{\delta}(c^2-1)+1=V.
\end{aligned}$$

The last line is true from (3.105), and so since we used equivalences all along we deduce that $X=0$. This means that

$$t(c)=\ln\left(\frac{c}{c+1}\right)-\delta\ln\left(1-\frac{u(c)}{2\delta}\right)-\overline{\delta}\ln\left(1-\frac{u(c)}{2\overline{\delta}}\right). \tag{3.106}$$

We let

$$a=4\delta\overline{\delta}\cdot(c^2-1), \tag{3.107}$$

and so

$$a'=\frac{\partial}{\partial c}a=8\delta\overline{\delta}c. \tag{3.108}$$

Recalling that $u(c)$ was defined in (3.103) as

$$u(c)=\frac{\sqrt{1+4\delta\overline{\delta}\cdot(c^2-1)}-1}{c^2-1}=\frac{\sqrt{1+a}-1}{c^2-1}, \tag{3.109}$$

we obtain:

$$\begin{aligned}
u'(c) &= \tfrac{\partial}{\partial c}u(c)\\
&= \tfrac{\frac{1}{2}\frac{a'}{\sqrt{1+a}}(c^2-1)-\left(\sqrt{1+a}-1\right)2c}{(c^2-1)^2}\\
&= \tfrac{c}{(c^2-1)^2}\cdot\left[\tfrac{4\delta\overline{\delta}(c^2-1)}{\sqrt{1+a}}-2\sqrt{1+a}+2\right]\\
&= \tfrac{c}{(c^2-1)^2}\cdot\left[\tfrac{a}{\sqrt{1+a}}-2\sqrt{1+a}+2\right]\\
&= \tfrac{c}{(c^2-1)^2}\cdot\left[\tfrac{1+a-1}{\sqrt{1+a}}-2\sqrt{1+a}+2\right]\\
&= \tfrac{c}{(c^2-1)^2}\cdot\left(2-\sqrt{1+a}-\tfrac{1}{\sqrt{1+a}}\right).
\end{aligned} \tag{3.110}$$

Now, using the expression for $t(c)$ in (3.106), we have:

$$\begin{aligned} t'(c) &= \tfrac{\partial}{\partial c} t(c) \\ &= \tfrac{c+1}{c}\tfrac{c+1-c}{(c+1)^2} + \delta\big[1+\tfrac{u(c)}{2\delta}\big]^{-1}\tfrac{u'(c)}{2\delta} + \overline{\delta}\big[1+\tfrac{u(c)}{2\overline{\delta}}\big]^{-1}\tfrac{u'(c)}{2\overline{\delta}} \\ &= \tfrac{1}{c(c+1)} + u'(c)\cdot\left[\tfrac{\delta}{2\delta+u(c)} + \tfrac{\overline{\delta}}{2\overline{\delta}+u(c)}\right] \\ &= \tfrac{1}{c(c+1)} + u'(c)\cdot\left[\tfrac{4\delta\overline{\delta}-u(c)}{4\delta\overline{\delta}-2u(c)+u(c)^2}\right]. \end{aligned}$$

Plugging (3.109) and (3.110) into this we obtain

$$t'(c) = \frac{1}{c(c+1)} + \frac{1}{c(c^2-1)}\cdot\frac{3a+4-(4+a)\sqrt{1+a}}{a+2-2\sqrt{1+a}}. \tag{3.111}$$

Now,

$$\begin{aligned} t'(c)>0 &\iff \tfrac{1}{c(c+1)} + \tfrac{1}{c(c^2-1)}\cdot\tfrac{3a+4-(4+a)\sqrt{1+a}}{a+2-2\sqrt{1+a}} > 0 \\ &\iff \tfrac{1}{c(c^2-1)}\cdot\tfrac{3a+4-(4+a)\sqrt{1+a}}{a+2-2\sqrt{1+a}} > -\tfrac{1}{c(c+1)} \\ &\iff -\tfrac{c(c+1)}{c(c^2-1)}\cdot\tfrac{3a+4-(4+a)\sqrt{1+a}}{a+2-2\sqrt{1+a}} < 1 \\ &\iff \tfrac{1}{c-1}\cdot\tfrac{3a+4-(4+a)\sqrt{1+a}}{2\sqrt{1+a}-a-2} < 1 \\ &\iff 3a+4-(4+a)\sqrt{1+a} > \big(c-1\big)\cdot\big(2\sqrt{1+a}-a-2\big) \\ &\iff 3a+4-\big(c-1\big)\cdot\big(-a-2\big) > (4+a)\sqrt{1+a} + \big(c-1\big)^2\cdot 2\sqrt{1+a} \\ &\iff 2a+2+ca+2c > \big(a+2c+2\big)\cdot\sqrt{1+a} \\ &\iff \big(2a+2+ca+2c\big)^2 > \big(a+2c+2\big)^2\cdot\big(1+a\big) \\ &\iff 4a^2+8a+4ca^2+12ca+4+8c+c^2a^2+4c^2a+4c^2 > \\ &\quad \big(a^2+4ca+4a+4c^2+8c+4\big)\cdot\big(1+a\big) \\ &\iff -a^3+a^2\big(c^2-1\big) > 0 \\ &\iff -a+c^2-1>0 \\ &\iff c^2-1 > 4\delta\overline{\delta}\big(c^2-1\big) \qquad \text{(by the definition of } a \text{ in (3.107))} \\ &\iff 1 > 4\delta\overline{\delta}. \end{aligned}$$

Since $\delta < \frac{1}{2}$ it follows that $4\delta\overline{\delta} < 1$, and therefore the last line is always true. Because we have used equivalences all the way we deduce that the original statement holds, namely $t'(c) > 0$ for all $c > 1$. So $t(c)$ is a strictly increasing function over the range we are concerned with.

• **3)** We show that $t(c)$ tends to zero when $c \to \infty$.
First recall from (3.103) that

$$u(c) = \frac{\sqrt{1 + 4\delta\overline{\delta} \cdot (c^2 - 1)} - 1}{c^2 - 1}.$$

So we have

$$\lim_{c\to\infty} u(c) = 0. \tag{3.112}$$

Now

$$\lim_{c\to\infty} t(c) = \lim_{c\to\infty} \left[\ln\left(\frac{c}{c+1}\right) - \delta \ln\left(1 - \frac{u(c)}{2\delta}\right) - \overline{\delta} \ln\left(1 - \frac{u(c)}{2\overline{\delta}}\right)\right].$$

Combining this with (3.112) we obtain

$$\lim_{c\to\infty} t(c) = \ln(1) + \lim_{u\to 0} \left[-\delta \ln\left(1 - \frac{u}{2\delta}\right) - \overline{\delta} \ln\left(1 - \frac{u}{2\overline{\delta}}\right)\right] = 0. \tag{3.113}$$

• **4)** Combining **2)** and **3)**, we deduce that $t(c) < 0$ for any $c > 1$, in particular $t(\hat{C}_2) < 0$. Therefore by setting

$$\tau_{24} = -t(\hat{C}_2),$$

we can deduce that

$$c \le \hat{C}_2 \implies t(c) \le -\tau_{24}. \tag{3.114}$$

Notice that $\hat{C}_2$ depends only on $\hat{B}_2$ (see (3.98)), which depends only on $\hat{\Gamma}_\ell$ (see (3.87)), which in turn depends only on R, δ and y. So as required, τ_{24} will depend only on R, δ and y.

Combining (3.114) with (3.104), for any α, c with $0 < \alpha < 2\delta$ and $\hat{C}_1 < c \le \hat{C}_2$, we have:

$$v(\alpha, c) \le t(c) \le -\tau_{24},$$

as required.

■

Finally we show that $s_1(n, w)$ is dominated by n, and will therefore be negligible.

Proposition 3.50. *If $\hat{\Gamma}_\ell \le \gamma \le \hat{\Gamma}_u$, then there are $\tau_{25} > 0$ and N_{25} with*

$$n \ge N_{25} \implies m_1(n, \ell, w) \le -\tau_{25} \cdot n.$$

Proof: First recall that

$$m_1(n, \ell, w) = s_1(n, w) + f_1(n, \ell, w) + g_1(n, \ell).$$

• **1)** We show that for n large enough $f_1(n,\ell,w)+g_1(n,\ell)\leq -\tau_{26}\cdot n$ for some τ_{26}.
Recall from (3.101) that for any $\epsilon_{23}>0$, there is N_{23} with

$$n\geq N_{23}\implies f_1(n,\ell,w)+g_1(n,\ell)\leq n\cdot\big[v(\alpha,\beta)+\epsilon_{23}\big], \tag{3.115}$$

Furthermore, Proposition 3.49 tells us that there is $\tau_{24}>0$ with

$$v(\alpha,\beta)\leq -\tau_{24},$$

and so by setting $\epsilon_{23}=\frac{\tau_{24}}{2}$ and $\tau_{26}=\frac{\tau_{24}}{2}$ we can ensure that

$$n\geq N_{23}\implies f_1(n,\ell,w)+g_1(n,\ell)\leq -\frac{\tau_{24}}{2}\cdot n=-\tau_{26}\cdot n.$$

• **2)** We show that $s_1(n,w)$ is dominated by $-\tau_{26}\cdot n$.
We know from Lemma 3.30 that

$$s_1(n,\ell)\leq -w\ln\left(\frac{w}{nR}\right)-\quad nR\ln\left(1-\frac{w}{nR}\right). \tag{3.116}$$

Since $\gamma=\frac{w}{n^{1-y}}$, this means that

$$\frac{s_1(n,\ell)}{n}\leq -\frac{\gamma}{n^y}\ln\left(\frac{1}{R}\cdot\frac{\gamma}{n^y}\right)\ -\quad R\ln\left(1-\frac{1}{R}\cdot\frac{\gamma}{n^y}\right). \tag{3.117}$$

Now because $\gamma\leq\hat{\Gamma_u}$, $\frac{\gamma}{n^y}$ tends to zero as n gets large. So because

$$\lim_{x\to 0}x\ln(x)=\lim_{x\to 0}\ln(1-x)=0,$$

we can deduce that $\frac{s_1(n,\ell)}{n}$ tends to zero as n gets large. So formally for any $\epsilon_{29}>0$ there is N_{29} with

$$n\geq N_{29}\implies\left|\frac{s_1(n,\ell,w)}{-\tau_{26}\cdot n}\right|\leq\epsilon_{29}. \tag{3.118}$$

3) We put all this together.
We need to be a little careful about using (3.118) to make sure the inequalities are in the right direction. (3.118) tells us that

$$n\geq N_{29}\implies -\epsilon_{29}\leq\frac{s_1(n,\ell,w)}{-\tau_{26}\cdot n},$$

and since $-\tau_{26}\cdot n<0$, this leads to

$$n\geq N_{29}\implies\big(-\tau_{26}\cdot n\big)\cdot\frac{s_1(n,\ell,w)}{-\tau_{26}\cdot n}\leq\big(-\tau_{26}\cdot n\big)\cdot\big(-\epsilon_{29}\big). \tag{3.119}$$

Setting $\epsilon_{29} = \frac{1}{2}$ we obtain

$$\begin{aligned} n \geq N_{29} \implies s_1(n,w) + f_1(n,\ell,w) + g_1(n,\ell) &\leq s_1(n,w) - \tau_{26} \cdot n \\ &= -\tau_{26} \cdot n \cdot \left(1 + \tfrac{s_1(m,w)}{-\tau_{26} \cdot n}\right) \\ &\leq -\tau_{26} \cdot n \cdot \big(1 - \epsilon_{29}\big) \qquad \text{(using (3.119))} \\ &= -\tfrac{\tau_{26}}{2} \cdot n \qquad \text{(since } \epsilon_{29} = \tfrac{1}{2}\text{),} \end{aligned}$$

and therefore setting $N_{25} = N_{29}$, and $\tau_{25} = \frac{\tau_{26}}{2}$ gives us the required result. ■

Once more this is a stronger result than was required, since we just needed to show that $m_1(n,\ell,w) \leq \tau_{25} \cdot n^y$. We summarize the result for Case 4 below:

Theorem 3.51. *Suppose we are given $0 < R < 1$ and $0 < \delta < \frac{1}{2}$. Let $\hat{\Gamma}_\ell$ and $\hat{\Gamma}_u$ be taken from Definitions 3.42 and 3.20. Then there are $N_{25}, \tau_{25} > 0$ (depending only on R, δ and y) for which*

$$n \geq N_{25} \implies \forall \ell = 1, \ldots, n,\ \forall w = \left\lceil \hat{\Gamma}_\ell \cdot n^{1-y} \right\rceil, \ldots, \left\lfloor \hat{\Gamma}_\ell \cdot n^{1-y} \right\rfloor :\ m(n,\ell,w) \leq \exp\big(-\tau_{25} \cdot n^y\big).$$

3.4.8 Conclusion

Now that we have covered all four cases presented in subsection 3.4.3, we can deduce the result we had set out to prove, namely Theorem 3.15, which we restate below:

Theorem 3.15. Suppose we are given $0 < R$, $0 < y < 1$ and $0 < \delta < \frac{1}{2}$ with $R < 1 - h(\delta)$. Then there are $N, \tau > 0$ (depending only on R, δ and y) for which

$$n \geq N \implies \forall \ell = 1, \ldots, \lfloor 2n\delta \rfloor, \forall w = 1, \ldots, \lfloor nR \rfloor \ :\ m(n,\ell,w) \leq \exp\big(-\tau \cdot n^y\big). \tag{3.120}$$

We can now complete the proof that our family of codes approaches the Gilbert-Varshamov bound with high probability. From Theorem 3.14 we know that

$$\Pr\Big[d_{\min}(\mathcal{C}) \leq n\delta\Big] \leq n\delta \cdot \sum_{w=1}^{nR} \sum_{\ell=1}^{2n\delta} m(n,\ell,w).$$

Theorem 3.15 then tells us that if $R < 1 - h(\delta)$ then there are $N, \tau > 0$ for which $N \geq n$ implies that

$$\Pr\Big[d_{\min}(\mathcal{C}) \leq n\delta\Big] \leq n\delta \cdot \sum_{w=1}^{nR} \sum_{\ell=1}^{2n\delta} \exp\big(-\tau \cdot n^y\big) \leq \big(2\delta^2 R\big) \cdot n^3 \cdot \exp\big(-\tau \cdot n^y\big).$$

So clearly we have

$$\lim_{n\to\infty} \Pr\Big[d_{\min}(\mathcal{C}) \leq n\delta\Big] = 0.$$

It would be interesting to determine the smallest column weight of M for which the resulting family still approaches the GV-bound. Our construction above had a column weight of $O(n^y)$, and we see that this value appears in the bound (3.120):

$$n \geq N \implies \forall \ell = 1, \ldots, \lfloor 2n\delta \rfloor, \forall w = 1, \ldots, \lfloor nR \rfloor \ : \ m(n, \ell, w) \leq \exp\big(- \tau \cdot n^y \big).$$

This leads to the question of whether a similar analysis on a construction using some other weight W would yield the modified bound

$$n \geq N \implies \forall \ell = 1, \ldots, \lfloor 2n\delta \rfloor, \forall w = 1, \ldots, \lfloor nR \rfloor \ : \ m(n, \ell, w) \leq \exp\big(- \tau \cdot W \big).$$

If this were the case when $W = \log(n) \cdot f(n)$, where $f(n)$ is any function for which $f(n) \to \infty$ when $n \to \infty$, then the corresponding family would approach the GV-bound, and be encodable in time $O\big(n \log(n) f(n)\big)$.

Chapter 4

Short Algebraic-Geometric Codes for Transmission over the Erasure Channel

4.1 Introduction

Algebraic-Geometric (AG) codes are arguably the most powerful class of algebraic codes in existence. They contain the Reed-Solomon (RS) codes as a subclass, but unlike RS-codes, they allow for the construction of arbitrarily long codes over a fixed alphabet, with asymptotically good performance. In fact it was shown [87] that for a square $q \geq 49$ it is possible to construct infinite families of AG-codes over $\mathbb{F}_q$ that beat the asymptotic Gilbert-Varshamov bound.

Despite their excellent properties, and despite the algorithmic advances regarding their encoding and decoding, there are very few practical uses of AG-codes, whereas RS-codes have been and are being used in many applications. One possible reason is that RS-codes are better understood, and have somewhat better hardware implementations.

Nevertheless, AG-codes are better than RS-codes since they allow the construction of much longer codes over the same alphabet, while enabling a similarly structured encoding and decoding process. This advantage can be interpreted in different ways. The straightforward interpretation is that larger pieces of data can be protected using the same field operations as RS-codes. A different interpretation is that if a piece of data is to be protected using a code of some given length n, then an AG-code allows this to be done with a smaller finite field, which in turn means that the encoding and decoding algorithms will run faster.

The latter interpretation could be a major insight into a practical exploitation of AG-codes. The reason is that in many applications the size of the data to be encoded is constrained by outer applications, such as those that do not allow an unreasonably long delay. Moreover, because practical implementations of encoding and decoding algorithms for AG-codes scale quadratically with the block-length, having an AG-code of large block-length may be unfeasible in many situations. However, for applications requiring very short blocks (such as video streaming), AG-codes can be made to run very fast.

This chapter is concerned with illustrating and quantifying the performance of very short AG-codes over the Erasure Channel, and more specifically of comparing them to RS-codes. A number of codes have been suggested to protect the data in this transmission model, the most prominent of which are Tornado codes [46], RA-codes [22], LT-codes [45], and Raptor codes [76]. While these have been shown to have excellent perfor-

mance on the Erasure Channel, the lengths of the codes need to be reasonably large. When very short blocks are required, AG and RS-codes become competitve solutions.

We compare the performances of AG and RS-codes for block lengths up to 64. The smaller field size enables faster encoding and decoding, but the drawback is an increased error probability due to the larger minimum distance. We measure this by developing an efficient algorithm to compute these exact error probabilities.

Finally, this work has been motivated by practical needs, which leads us to focus on a specific transmission problem. We obtain some practical data to illustrate the speed-ups predicted in theory. The work was done in collaboration with the company Digital Fountain and the codes presented are being used in some of their commercial products. This is, as far as we know, the first practical use of AG-codes.

4.2 The Erasure Channel

We will be concerned in this chapter with transmissions over the Q-ary erasure channel. Informally, an alphabet element sent over this channel is either received intact (with some probability $1-p$) or lost completely (with probability p). In the latter case it it said to have been *erased*.

Definition 4.1. The Q-ary erasure channel over an alphabet Σ of size Q has input set Σ, output set $\Sigma \cup \{?\}$ (where ? means *erasure*) and transition matrix $M = (M_{ij})_{i\in\Sigma, j\in\Sigma\cup\{E\}}$, where

$$M_{ij} = \begin{cases} p & \text{if } j = ? \\ 1-p & \text{if } j = i \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

Decoding a linear code over this channel is particularly simple. Given a generator matrix, decoding can be reduced to solving a system of linear equations. If G denotes the $k \times n$ generator matrix, u a message vector and c the corresponding codeword then we know that

$$uG = c. \tag{4.2}$$

Given only G and c, recovering u amounts to solving a system of n linear equations in k variables. Each erasure removes one component of c. In other words it removes one equation (corresponding to one column of G). If $I \subseteq [n]$ denotes the set of indices of the positions that are *not* erased (we call these *intact*), then decoding reduces to solving the system of equations

$$uG' = c' \tag{4.3}$$

where G' is the $k \times |I|$ submatrix of G consisting of those columns whose indices are in I, and c' is the subvector of c containing the indices in I. We say that the decoding *succeeds* if this submatrix G' has rank k (i.e. we can solve the system), and *fails* otherwise. It is clear that if $|I| < k$ then the decoding will always fail.

Proposition 4.2. *If a codeword of an $[n, k, d]_Q$-code is transmitted over the Q-ary erasure channel, and $\geq n - d + 1$ positions are intact (equivalently $\leq d - 1$ position are erased), then the decoding will succeed.*

Proof: Clearly the system (4.3) has at least one solution (namely the actual message vector u). So we need to show that this solution is unique. If there was another solution $v \in \mathbb{F}_Q^k$ then the codeword $vG \in \mathbb{F}_Q^n$ would have the same entries as c at the positions in I (and $|I| = n - d + 1$). Therefore

$$d(vG, c) \leq n - (n - d + 1) = d - 1, \tag{4.4}$$

leading to a contradiction (since vG and c are both codewords). ■

Although very simple the Q-ary erasure channel has been very relevant, in large part due to the Internet. Data is divided into packets to which are appended checksums. At the receiver side packets are either assumed to be intact or simply discarded. The latter can happen for various reasons, for example if the checksum verification fails, or the packet might simply not arrive if a router runs out of buffer memory somewhere along the way.

4.3 Algebraic-Geometric Codes

In this section we describe the construction and properties of AG-codes. We start by looking at RS-codes, which are in fact special cases of AG-codes.

4.3.1 Reed-Solomon Codes

Throughout this chapter $\mathbb{F}_q$ will denote the finite field of size q. We first note that there is a bijection between $\mathbb{F}_q^k$ and the set of polynomials in $\mathbb{F}_q[x]$ of degree $< k$:

Definition 4.3. For $u = (u_1, \dots u_k) \in \mathbb{F}_q^k$ we define the corresponding polynomial

$$f_u(x) = \sum_{i=0}^{k-1} u_{i+1} \cdot x^i. \tag{4.5}$$

We will define Reed-Solomon codes through their encoding map.

Definition 4.4. Let $\mathbb{F}_q$ be a finite field, let $k \leq n \leq q$, and let $\alpha_1, \dots, \alpha_n$ be distinct elements of $\mathbb{F}_q$. The $[n, k]_q$ *Reed-Solomon* (RS) code corresponding to these field elements has encoding map $\varphi : \mathbb{F}_q^k \to \mathbb{F}_q^n$ with

$$\varphi\big(u\big) = \big(f_u(\alpha_1), \dots, f_u(\alpha_n)\big). \tag{4.6}$$

So RS-codes are obtained by evaluating polynomials of bounded degrees at field elements. The Singleton bound states that for any $[n, k, d]$-code

$$d \leq n - k + 1, \tag{4.7}$$

and codes for which we have equality in (4.7) are said to be *maximum distance separable* (MDS). This is the case for RS-codes:

Theorem 4.5. *Reed-Solomon codes are MDS.*

Proof: See for example [82]. ■

This means that over the erasure channel k intact (non-erased) elements are sufficient to guarantee successful decoding (the code can recover from $d - 1 = n - k$ erasures). Since k intact elements are also necessary (or else there is not enough information), MDS codes are sometimes also called *optimal*.

4.3.2 Algebraic-Geometric Codes

AG-codes are a natural generalization of RS-codes. For a full introduction to AG-codes, see [60] or [81]. We will assume knowledge of elementary algebraic geometry, which can be found in [70][72][81]. RS-codes are constructed by evaluating polynomials of bounded degree at certain field elements. As explained in the introduction, their big drawback is the fact that the length is bounded by the field size (since the polynomials must be evaluated at *distinct* elements), so long codes require large fields. The most obvious way around this would be to evaluate multivariate polynomials at points of $\mathbb{F}_q^m$, this is the principle of *Reed-Muller codes* (so RS-codes would be the special case $m = 1$). However, while this does indeed increase the length, it also incurs a large cost in the decrease of the minimum distance.

There is a more efficient way of improving the length. Instead of evaluating all multivariate polynomials up to a certain degree at randomly chosen elements of $\mathbb{F}_q^m$, we evaluate certain functions at well chosen points of this space. These well chosen points are the elements of an *algebraic curve*, and the functions will be taken from some *linear space* of this curve.

Definition 4.6. Let $\mathcal{X}$ be a smooth nonsingular curve of genus g over $\mathbb{F}_q$, let $P_1, \ldots, P_n, Q$ be $n+1$ distinct $\mathbb{F}_q$-rational points of $\mathcal{X}$, let $\alpha < n$ be a positive integer, let $\mathcal{L}(\alpha Q)$ be the linear space of the divisor αQ. A (one-point) Algebraic-Geometric (AG) code $\mathcal{C}$ is obtained as the image of the evaluation map $\varphi : \mathcal{L}(\alpha Q) \to \mathbb{F}_q^n$ with

$$\varphi(f) = \big(f(P_1), \ldots, f(P_n)\big). \tag{4.8}$$

We will denote such a code by $\mathcal{C}[\mathcal{X}, (P_1, \ldots, P_n), \alpha Q]$. The *genus of the code* refers to the genus of the underlying curve.

Explicitly constructing these codes (for example by finding a generator matrix) is somewhat more difficult than RS-codes. We essentially need to know the points on the curve $\mathcal{X}$, and a basis of the $\mathbb{F}_q$-space $\mathcal{L}(\alpha Q)$. Fortunately, this can be computed using the algorithm of Heß from [31].

The resulting dimension and minimum distance are described in the following proposition [72]:

Proposition 4.7. *Let $\mathcal{C}$ be an AG-code defined as above. Then $\mathcal{C}$ is an $[n, k, d]$-code with*

$$k \geq \alpha + 1 - g \tag{4.9}$$
$$d \geq n - \alpha. \tag{4.10}$$

Furthermore if $2g - 2 < \alpha$ then we have equality in (4.9).

Proof: We will start by showing that the evaluation map φ defined in (4.8) is injective. Suppose there is some $f \in \mathcal{L}(\alpha Q)$ with $\varphi(f) = 0$. This means that for all $i \in [n]$

$$f(P_i) = 0, \tag{4.11}$$

and therefore that f has at least n zeros. But since $f \in \mathcal{L}(\alpha Q)$ it has only one pole of degree α, and so because $\alpha < n$, we must have $f = 0$ (f must have as many zeros as poles). So φ is injective, and therefore $k = \dim(\alpha Q)$. Now the Theorem of Riemann [72] tells us that

$$\dim(\alpha Q) \geq \deg(\alpha Q) + 1 - g = \alpha + 1 - g, \tag{4.12}$$

with equality if $2g - 2 < \alpha$. So (4.9) follows immediately.

Next, suppose there is a non-zero codeword of weight $< n - \alpha$. This means that there is $f \in \mathcal{L}(\alpha Q)$ which has at least $\alpha + 1$ zeros. However once again f must have as many zeros as poles, so cannot have more than α zeros, so we must have $f = 0$. We deduce that all non-zero codewords have weight at least $n - \alpha$, from which (4.10) follows. ■

We will assume for the rest of this chapter that $2g - 2 < \alpha < n$. We can deduce from Proposition 4.7 that

$$n - k + 1 - g \ \leq \ d \ \leq \ n - k + 1, \tag{4.13}$$

where the second inequality follows from the Singleton bound. So the genus g represents the "gap" to the Singleton bound, and it is therefore desirable to choose a curve whose genus is as small as possible.

In Definition 4.6, constructing a code of length n required $n + 1$ *distinct* points on the curve $\mathcal{X}$, which means that $\mathcal{X}$ needs to have at least $n + 1$ points. It turns out that over a given field, a curve must have large genus to have many points. We therefore have a trade-off between the length of the code (we would like $\mathcal{X}$ to have many points) and its distance (we would like $\mathcal{X}$ to have a small genus). For a given field size q, the maximum number of points on a curve over $\mathbb{F}_q$ of genus g is denoted $N_q(g)$, and a curve over $\mathbb{F}_q$ of genus g having $N_q(g)$ points is called a *maximal curve*.

4.4 The Specific Codes

We will be concerned only with very short codes (with lengths up to 64). As explained in the introduction, the work in this chapter was motivated by certain practical needs, and for this reason we use only finite fields in the form $\mathbb{F}_{2^\ell}$, where ℓ is a power of 2. Indeed it has been established that working with such fields yields very large practical advantages (essentially due to the representation of field elements by bytes rather than uneven fractions thereof). Were we to use RS-codes, we could work over $\mathbb{F}_{16}$ for length of up to 16, then over $\mathbb{F}_{256}$ for lengths between 17 and 64. However with AG-codes we can use $\mathbb{F}_{16}$ for any length. The cost of doing this is that we need to use a curve with more points, and therefore with higher genus, which decreases the minimum distance.

Since a larger genus means a smaller minimum distance (see (4.13)), for a given length n we would like to use a curve with genus as small as possible. This means taking the smallest g for which $N_{16}(g) \geq n + 1$ (see Section 4.3.2), and then using a maximal curve of genus g. Determining N_q and finding maximal curves is a well studied problem, partly motivated by the construction of good AG-codes. The best known upper and lower bounds on $N_q(g)$ for many values of q and g are regularly updated in [88]. The following table (see [88] [67] [68] [69] [55] [66]) gives the value of $N_{16}(g)$ (or the best known bounds), and a corresponding maximal curve:

g	$N_{16}(g)$	maximal curve
1	25	$x^2 + x = y^3 + y$
2	33	$x^5 = y^2 + y$
3	38	$x^2y^4 + x^2y = \omega x^3 + 1$
4	45	$y^2 + xy + x^2 + y^3 + xy^3 + x^2y^2 + x^3y + x^4 + x^3y^2 + x^4y + x^5 = 0$
5	$\in [49, 53]$	$-$
6	65	$x^5 = y^4 + y$

Table 4.1: Maximal curves over $\mathbb{F}_{16}$ for genera 1 to 6.

(for the genus 3 curve, ω denotes a third root of unity in $\mathbb{F}_{16}$). We will not use a genus 5 curve, partly because is it not known whether the best known curves are maximal, and mainly because we make only a small gain in length compared to the genus 4 curve.

For a given n, we choose the curve of smallest genus with which we can construct a code of length n:

n	which curve
$n \leq 16$	Reed-Solomon ($g = 0$)
$17 \leq n \leq 24$	$g = 1$
$25 \leq n \leq 32$	$g = 2$
$33 \leq n \leq 37$	$g = 3$
$38 \leq n \leq 44$	$g = 4$
$45 \leq n \leq 64$	$g = 6$

Table 4.2: Curves used in our application.

We will see in section 4.6 that the encoding technique we use requires on average only half as many basic operations for codes over $\mathbb{F}_{16}$ as for codes over $\mathbb{F}_{256}$ (which would be the standard method for these sorts of lengths). On top of these theoretical advantages, a smaller field also means that in practical implementations more machine dependent optimizations are possible.

As seen in the previous section, the price we pay for this speed-up is a decrease in the minimum distance of the code (by an additive factor of g), which in turn means that for a fixed erasure channel the probability of unsuccessful decoding (referred to as the *error probability*) will increase. To quantify this we derive in the next section an efficient algorithm for computing the exact error probabilities.

4.5 Computing the Error Probabilities

Recall that we are considering transmission over the Q-ary erasure channel, in which each alphabet element is either received intact (with probability $1 - p$) or lost completely (with probability p). So for a given code $\mathcal{C}$, we transmit n elements, some of which might get erased. Let $I \subseteq [n]$ denote the indices of the elements that are *not* erased (we call these *intact*).

We say that I is *good* if we can recover our codeword from the elements in I, and *bad* otherwise. So if G is the generator matrix of our code then I is good if and only if the $k \times |I|$ submatrix of G constructed by

taking only the columns with indices in I has full rank. So clearly all sets I with $|I| < k$ are bad. Likewise, as seen in section 4.3, an AG-code of genus g has minimum distance at least $n - k + 1 - g$. So all sets I with $|I| \geq k + g$ are good (in particular with RS-codes, of genus 0, all sets of size $\geq k$ are good).

Definition 4.8. For a given $[n, k]$-code $\mathcal{C}$, we define B_r to be the number of bad subsets of size r.

Notice that B_r depends only on the code $\mathcal{C}$. Since there are in total $\binom{n}{r}$ subsets $I \subseteq [n]$ of size r (i.e., the number of erasure patterns), the fraction of subsets of size r that are bad is $B_r / \binom{n}{r}$. Furthermore for a fixed number of erasures $n - r$, all erasure patterns are equally likely, so we obtain the following proposition:

Proposition 4.9. *The error probability with an $[n, k]$ AG-code over the Q-ary erasure channel with erasure probability p is given by*

$$\sum_{r=0}^{n} \frac{B_r}{\binom{n}{r}} \cdot P\big[\ |I| = r\ \big] = \sum_{r=0}^{n} B_r \cdot (1-p)^r \cdot p^{n-r}. \tag{4.14}$$

With an RS-code (genus 0), I is bad if and only if $|I| < k$, so we have

$$B_r = \begin{cases} \binom{n}{r} & \text{if } r < k \\ 0 & \text{otherwise .} \end{cases}$$

We can therefore deduce:

Corollary 4.10. *The error probability with an $[n, k]$ RS-code over the Q-ary erasure channel with erasure probability p is given by*

$$\sum_{r=0}^{k-1} \binom{n}{r} \cdot (1-p)^r \cdot p^{n-r}.$$

With AG-codes of genus greater than 0, the situation is more complicated. As above, decoding will fail whenever $|I| < k$, but on top of that it will also sometimes fail when $k \leq |I| \leq k + g - 1$ (see Figure 4.1 below). So we need to determine how often it fails in these cases, i.e., to find the values of B_r.

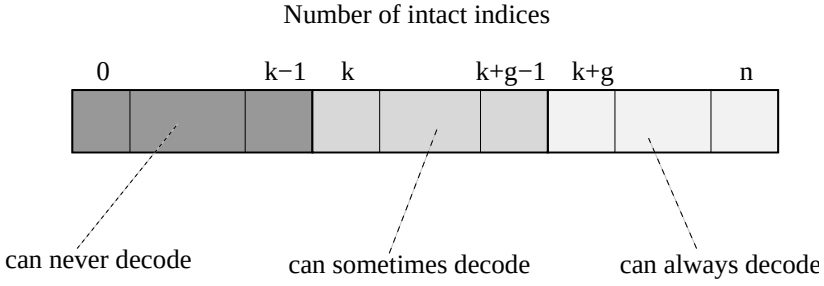


Figure 4.1: The overhead of an AG-code of genus g.

4.5.1 Reduction to an Abelian Group Problem

We will show that for an AG-code, the problem of determining the number of bad subsets of a given size reduces to an abelian group problem.

Definition 4.11. Let G be a finite abelian group. Suppose we have two subsets $S, T \subseteq G$, and an integer r. We denote by $\theta(G, S, T, r)$ the number of r-subsets $W \subseteq S$ for which

$$\sum_{w \in W} w \quad \in T.$$

Now we suppose throughout that we have fixed AG-code $\mathcal{C}\big[\mathcal{X}, (P_1, \ldots, P_n), \alpha Q\big]$.

We denote by $\mathbb{D}(\mathcal{X})$ the divisor group of $\mathcal{X}$, and by $\mathbb{D}^0(\mathcal{X})$ its subgroup consisting in the divisors of degree 0. Recalling that the principal divisors $\text{Prin}(\mathcal{X})$ form a subgroup of $\mathbb{D}^0(\mathcal{X})$, $\text{Pic}(\mathcal{X})$ (the *Picard group*) and $\text{Pic}^0(\mathcal{X})$ are defined as

$$\text{Pic}(\mathcal{X}) = \mathbb{D}(\mathcal{X})/\text{Prin}(\mathcal{X}), \qquad \text{and} \qquad \text{Pic}^0(\mathcal{X}) = \mathbb{D}^0(\mathcal{X})/\text{Prin}(\mathcal{X}). \tag{4.15}$$

Definition 4.12. For a divisor $D \in \mathbb{D}(\mathcal{X})$, we will denote by $\overline{D}$ the image of the divisor $\big(D - \deg(D) \cdot Q\big)$ in $\text{Pic}^0(\mathcal{X})$.

Note that since $D - \deg(D) \cdot Q$ has degree 0, its image modulo $\text{Prin}(\mathcal{X})$ is indeed in the group $\text{Pic}^0(\mathcal{X})$. The following Theorem establishes the link between our problem of determining error probabilities, and the function θ from Definition 4.11:

Theorem 4.13. *Suppose we have an AG-code $\mathcal{C}\big[\mathcal{X}, (P_1, \ldots, P_n), \alpha Q\big]$. The number of bad subsets $I \subseteq [n]$ with $|I| = r$ is given by*

$$\theta\big(\text{Pic}^0(\mathcal{X}), S, T, r\big),$$

where $S = \{\overline{P_1}, \ldots, \overline{P_n}\}$, and $T = \big\{\overline{-D} \;\big|\; D$ is a positive divisor of degree $\alpha - r\big\}$.

Proof: Let I be a subset of $[n]$ of size r. Let K be the function field of $\mathcal{X}$. We have:

$$\begin{aligned}
I \text{ is bad} \quad &\iff \quad \text{there is a codeword that is zero at all entries } i \in I \\
&\iff \quad \exists f \in \mathcal{L}(\alpha Q) : f(P_i) = 0 \;\; \forall i \in I \\
&\iff \quad \exists f \in K : (f) \geq \big(\textstyle\sum_{i \in I} P_i\big) - \alpha Q \\
&\iff \quad \exists f \in K, D \in \mathbb{D}(\mathcal{X}) : D \geq 0, \; (f) = \big(\textstyle\sum_{i \in I} P_i\big) - \alpha Q + D. \\
&\iff \quad \exists f \in K, D \in \mathbb{D}(\mathcal{X}) : D \geq 0, \; (f) = \Big(\textstyle\sum_{i \in I} (P_i - Q)\Big) + D - (\alpha - r) \cdot Q.
\end{aligned}$$

Notice that since $\deg\big((f)\big) = 0$, any D that satisfies the last line will have degree $\alpha - r$. Now if we take the projection onto $\text{Pic}^0(\mathcal{X})$, then we get:

$$\begin{aligned}
I \text{ is bad} \quad &\iff \quad \exists f \in K, D \in \mathbb{D}(\mathcal{X}) : D \geq 0, \text{ and } \overline{(f)} = \big(\textstyle\sum_{i \in I} \overline{P_i}\big) + \overline{D} \\
&\iff \quad \exists D \in \mathbb{D}(\mathcal{X}) : D \geq 0, \text{ and } \overline{0} = \big(\textstyle\sum_{i \in I} \overline{P_i}\big) + \overline{D} \\
&\iff \quad \exists D \in \mathbb{D}(\mathcal{X}) : D \geq 0, \text{ and } \textstyle\sum_{i \in I} \overline{P_i} = \overline{-D} \\
&\iff \quad \textstyle\sum_{i \in I} \overline{P_i} \in T,
\end{aligned}$$

where $T = \{\overline{-D} \mid D \in \mathbb{D}(\mathcal{X}) \text{ and } D \geq 0\}$.

Once more, the only candidates for D that can verify this property must have degree $\alpha - r$. ■

4.5.2 The Group Algebra $\mathbb{C}[G]$

We now look at how to compute $\theta(G, S, T, r)$. The brute force approach would be to consider all $\binom{n}{r}$ r-subsets W of S and count how many of them have the property that $\sum_{w \in W} w \in T$. This would require $(r-1) \cdot \binom{n}{r}$ group operations in G, and $\binom{n}{r}$ tests of whether an element $g \in G$ belongs to T (namely $\sum_{w \in W} w$). So this is exponential in $n = |G|$ if r is a constant fraction of n, which makes the method highly impractical.

A better approach is to consider the group algebra $\mathbb{C}[G]$.

Definition 4.14. Let $G = (\{g_1, \ldots, g_m\}, +)$ be a finite abelian group and let $\mathbb{C}$ denote the field of complex numbers. The *group algebra* $\mathbb{C}[G]$ is a vector space over $\mathbb{C}$ of dimension m with basis elements $[g_1], \ldots, [g_m]$. There is a product on the basis elements $\mathbb{C}[G]$

$$[g_i] \cdot [g_j] = [g_i + g_j],$$

that extends naturally to the whole vector space. $\mathbb{C}[G]$ forms a ring under this product and the standard vector space addition.

Let $w \in \mathbb{C}[G]$. We can write w as an m-components vector (in basis $\{[g_1], \ldots, [g_m]\}$):

$$w = \begin{pmatrix} c_1 \\ \vdots \\ c_m \end{pmatrix} = \sum_{j=1}^{m} c_j \cdot [g_j].$$

We then call $c_j \in \mathbb{C}$ the j^{th} *component* of w. Notice that while addition in $\mathbb{C}[G]$ is done component by component, multiplication is more complicated:

$$\left(\sum_{i=1}^{m} a_i [g_i]\right)\left(\sum_{j=1}^{m} b_j [g_j]\right) = \sum_{\ell=1}^{m} \left(\sum_{i,j \mid g_i + g_j = g_\ell} a_i b_j\right)[g_\ell].$$

So multiplying two elements of $\mathbb{C}[G]$ requires m^2 multiplications and $m \cdot (m-1)$ additions in $\mathbb{C}$.

We will now look at elements of the polynomial ring $\mathbb{C}[G][x]$ (polynomials whose coefficients are in $\mathbb{C}[G]$).

Definition 4.15. Let $G = \{g_1, \ldots, g_m\}$ be a finite abelian group, and let $S \subseteq G$. We define the polynomial $p_S(x) \in \mathbb{C}[G][x]$ as follows:

$$p_S(x) = \prod_{g_i \in S} (x + [g_i]).$$

Note that $p_S(x)$ has degree $|S|$. This polynomial will be of great interest to us because it is closely linked to $\theta(G, S, T, r)$, as established by the following theorem:

Theorem 4.16. *Let $G = \{g_1, \ldots, g_m\}$ be a finite abelian group and let S be a subset of G, with $|S| = n$. Let $v_0, \ldots, v_n \in \mathbb{C}[G]$ be the coefficients of $p_S(x)$, so that*

$$p_S(x) = \sum_{i=0}^{n} v_i \cdot x^i.$$

For each $i = 0, \ldots, n$, let $a_{ij} \in \mathbb{C}$ be the j^{th} component of v_i:

$$v_i = \sum_{j=1}^{m} a_{ij} \cdot [g_j] \quad \in \mathbb{C}[G].$$

Then for any $T \subseteq G, r = 0, \ldots, n$ we have

$$\theta\big(G, S, T, r\big) = \sum_{j | g_j \in T} a_{(n-r)j}.$$

So Theorem 4.16 is saying that the i^{th} coefficient $v_i \in \mathbb{C}[G]$ of $p_S(x)$ holds all the information we need about $(n-i)$-subsets of S. Its j^{th} component a_{ij} is a positive integer and represents the number of $(n-i)$-subsets of S whose elements sum up to g_j in G.

Proof: The *elementary symmetric polynomials* [92] in n variables are defined as

$$\sigma_i(x_1, \ldots, x_n) = \sum_{W \subseteq [n], |W| = n-i} \prod_{\ell \in W} x_\ell, \qquad \text{for } \; i = 0, \ldots, n. \tag{4.16}$$

They have the property that for any $a_1, \ldots, a_n$:

$$\prod_{i=1}^{n} (x + a_i) = \sum_{i=0}^{n} \sigma_i(a_1, \ldots, a_n) \cdot x^i. \tag{4.17}$$

Since $p_S(x)$ is in the same form as the left hand side of (4.17), its coefficients v_i can be written as

$$v_i = \sigma_i\big([g_1], \ldots, [g_n]\big) = \sum_{W \subseteq [n], |W| = n-i} \prod_{\ell \in W} [g_\ell] = \sum_{W \subseteq [n], |W| = n-i} \Big[\sum_{\ell \in W} g_\ell \Big], \tag{4.18}$$

from which the result follows. ■

Example 4.17. It is perhaps more intuitive to see why this theorem holds with small examples. If $n = 3$ and $S = \{g_1, g_2, g_3\}$ then it can easily be checked that

$$p_S(x) = x^3 + \big([g_1] + [g_2] + [g_3]\big) \cdot x^2 + \big([g_1 + g_2] + [g_1 + g_3] + [g_2 + g_3]\big) \cdot x + \big([g_1 + g_2 + g_3]\big). \tag{4.19}$$

Now S has three 1-subsets ($\{g_1\}, \{g_1\}$ and $\{g_3\}$), which all appear in the coefficient of x^2. Likewise S has three 2-subsets ($\{g_1, g_2\}, \{g_1, g_3\}$ and $\{g_2, g_3\}$), which all appear in the coefficient of x (more precisely the sum of whose elements all appear). Finally, S has of course a single 3-subset $\{g_1, g_2, g_3\}$, the sum of whose elements appears in the constant coefficient.

Recall that our aim is to determine B_r, the number of bad subsets of size r for $r = k, \ldots, k + g - 1$. From Theorem 4.13 we know that this can be reduced to computing

$$\theta_r = \theta\big(G, S, T, r\big), \tag{4.20}$$

where $G = \text{Pic}^0(\mathcal{X})$, and $S, T \subseteq G$ (see Theorem 4.13). Now Theorem 4.16 tells us that we can determine θ_r for any r from $p_S(x) \in \mathbb{C}[G][x]$. Our next step is to efficiently compute this polynomial.

4.5.3 Efficiently Computing the Polynomial $p_S(x)$

Throughout we suppose that $|G| = m$ and $|S| = n$. Computing $p_S(x)$ requires $\frac{n\cdot(n-1)}{2}$ multiplications in $\mathbb{C}[G]$ (and the same number of additions in $\mathbb{C}[G]$). While adding two vectors $u, v \in \mathbb{C}[G]$ is done component-wise (and so requires m additions in $\mathbb{C}$), multiplying u and v is more complicated and requires m^2 multiplications and $m \cdot (m-1)$ additions in $\mathbb{C}$.

However, using *Fast Fourier Transforms (FFT)* [17], two vectors in $\mathbb{C}[G]$ can be multiplied much faster. Multiplication in the algebra $A = \mathbb{C}[G] = (\mathbb{C}^m, \cdot)$ is slow (namely $O(m^2)$ operations in $\mathbb{C}$), while in the algebra $B = (\mathbb{C}^m, *)$ (where $*$ denotes component-wise multiplication), we can multiply two elements using only m multiplications in $\mathbb{C}$. The algebras A and B can be linked through *Discrete Fourier Transforms (DFT)*.

Definition 4.18. Given a cyclic group C_ℓ, the corresponding DFT matrix $D_\ell \in \mathbb{C}^{\ell\times\ell}$ is defined as

$$(D_\ell)_{ij} = \omega^{(i-1)(j-1)}, \tag{4.21}$$

where $\omega = e^{\frac{2i\pi}{\ell}}$ is a primitive ℓ^{th} root unity in $\mathbb{C}$.

It can easily be checked that D_ℓ is invertible, with

$$(D_\ell^{-1})_{ij} = \frac{1}{\ell} \cdot \omega^{-(i-1)(j-1)}. \tag{4.22}$$

Proposition 4.19. *Let $G = C_{\ell_1} \times \ldots \times C_{\ell_k}$ be an abelian group with $|G| = m$. Let $A = \mathbb{C}[G] = \big(\mathbb{C}^m, \cdot\big)$ and $B = \big(\mathbb{C}^m, *\big)$, where $*$ denotes component-wise multiplication. Let D_G be the $m \times m$ matrix defined as follows:*

$$D_G = D_{\ell_1} \otimes \ldots \otimes D_{\ell_k}. \tag{4.23}$$

Then there are bases of A and B for which the mapping $\varphi : A \to B$ given by

$$\varphi(u) = D_G \cdot u \tag{4.24}$$

is a $\mathbb{C}$-algebra isomorphism.

Directly computing $\varphi(u)$ would require $O(m^2)$ operations in $\mathbb{C}$. However we can make use of (4.23), and successively multiply appropriate subvectors of u by each D_{ℓ_i}, which will require only $O\big(\sum_{i=1}^k \ell_i^2\big)$ operations.

Note: This can actually be further reduced to $O\big(\sum_{i=1}^k \ell_i \log(\ell_i)\big)$ operations using Fast Fourier Transforms (see for example [17], Chapter 13), but will not be necessary for the codes in which we are interested.

Corollary 4.20. *Let $G = C_{\ell_1} \times \ldots \times C_{\ell_k}$ be an abelian group of size m. Let A and B be the two algebras as above. Then the DFT $\varphi : A \to B$ can be computed using $\sum_{i=1}^k \ell_i^2$ multiplications and $\sum_{i=1}^k \ell_i(\ell_i - 1)$ additions in $\mathbb{C}$, i.e. a total of*

$$\sum_{i=1}^k 3\ell_i^2 - \ell_i = O\Big(\sum_{i=1}^k \ell_i^2\Big) \tag{4.25}$$

operations in $\mathbb{C}$.

So we can compute the polynomial

$$p_S(x) = \prod_{i=1}^{n} \big(x + [g_i]\big) \tag{4.26}$$

by first using the Fourier transforms $h_i = \varphi\big([g_i]\big)$ to obtain the polynomial $\hat{p_S}(x) \in B[x]$ defined as

$$\hat{p_S}(x) = \prod_{i=1}^{n} \big(x + h_i\big) = \sum_{i=0}^{n} w_i \cdot x^i, \tag{4.27}$$

and then taking the inverse transforms $v_i = \varphi^{-1}(w_i)$ to get the coefficients of $p_S(x) = \sum_{i=0}^{n} v_i \cdot x^i$. The algorithm is given below in pseudo-code:

Algorithm 4.1: COMPUTE $p_S(x)$
Input: An abelian group G, and a subset $S = \{g_1, \ldots, g_n\} \subseteq G$.
Output: The coefficients $(v_0, \ldots, v_n)$ of the polynomial $p_S(x) = \prod_{i=1}^{n} \big(x + [g_i]\big)$

```
1:  for d = 1 to n do
2:      compute h_d ← φ(g_d)
3:      for i = 1 to m do
4:          w_d[i] ← 1
5:          for j = 1 to d − 1 do
6:              w_{d−j}[i] ← w_{d−j}[i] · h_d[i] + w_{d−j−1}[i]
7:          end for
8:          w_0[i] ← h_d[i]
9:      end for
10: end for
11: for k = 0 to n − 1 do
12:     compute v_k ← φ^{−1}(w_k)
13: end for
14: return (v_0, ..., v_{n−1}, 1)
```

To evaluate the running time of this algorithm we first let $t = \sum_i \ell_i^2$ be the number of operations required for the DFTs (see Proposition 4.19). We decompose the operations as follows:

1. n DFTs (line 2). This requires $O(nt)$ operations in $\mathbb{C}$.
2. $O(mn^2)$ multiplications in $\mathbb{C}$ (line 6).
3. $O(mn^2)$ additions in $\mathbb{C}$ (line 6).
4. n inverse DFTs (line 12). This requires $O(nt)$ operations in $\mathbb{C}$.

This gives us a total of $O(mn^2 + nt)$ operations in $\mathbb{C}$.

4.5.4 The Final Algorithm

We can now combine all the work above to obtain the following algorithm for computing the error probabilities of our codes:

Algorithm 4.2: ERROR PROBABILITY OF AN AG-CODE
Input: An AG-code $\mathcal{C}\big[\mathcal{X}, (P_1, \ldots, P_n), \alpha Q\big]$, and a channel erasure probability p.
Output: The probability of a decoding failure

1: $g \leftarrow \text{genus}(\mathcal{X})$
2: $k \leftarrow \alpha + 1 - g$
3: $G \leftarrow \text{Pic}^0(\mathcal{X})$
4: $S \leftarrow \{\overline{P_1}, \ldots, \overline{P_n}\}$
5: compute the coefficients $v_0, \ldots, v_n \in \mathbb{C}[G]$ of $p_S(x)$ (where $v_i = \sum_{j=1}^{m} a_{ij} \cdot [g_j]$)
6: **for** $r = k$ to $k + g$ **do**
7: $\quad T \leftarrow \big\{\overline{-D} \;\big|\; D$ is a positive divisor of degree $\alpha - r\big\}$.
8: $\quad B_r \leftarrow \sum_{j|g_j \in T} a_{(n-r)j}$
9: **end for**
10: **return** $\sum_{r=0}^{k-1} \binom{n}{r} \cdot p^{n-r} \cdot (1-p)^r \;+\; \sum_{r=k}^{k+g} B_r \cdot p^{n-r} \cdot (1-p)^r$

Notes: • In step 3 we use the software package Magma [19] to compute $\text{Pic}^0(\mathcal{X})$.

• Computing T (step 7) can be done for example by brute force search since there is only a finite number of positive divisors of degree $\alpha - r$ in $\mathbb{D}(\mathcal{X})$ (we can enumerate the prime divisors of degrees at most $\alpha - r$, and look at all appropriate combinations).

• $p_S(x)$ only needs to be computed once to obtain the error probabilities for codes of length n of all dimensions k (since $P_1, \ldots, P_n$ stay the same in the construction, only α changes). Furthermore, assuming that for each n the set of points P_i we use for our codes of length n is contained in the set we use for our codes of length $n+1$, then we can construct the $p_S(x)$ of degree $n+1$ from that of degree n (see the algorithm for generating $P_S(x)$).

4.5.5 The Error Probabilities for our Specific Codes

The $\text{Pic}^0(\mathcal{X})$ groups of the curves in which we are interested (see Table 4.1) are given in Table 4.3 below, along with the corresponding value of $t = \sum_i \ell_i^2$. These were computed with the Magma software package [19].

Genus	$G = \text{Pic}^0(\mathcal{X})$	$m = \lvert G \rvert$	t
1	$C_5 \times C_5$	25	50
2	$C_5 \times C_5 \times C_5 \times C_5$	525	100
3	$(C_3)^3 \times (C_8)^3$	13,824	219
4	$(C_3)^4 \times (C_8)^4$	331,776	292
6	$(C_5)^{12}$	244,140,625	300

Table 4.3: Pic^0 groups of our curves

We implemented in C++ the algorithm above on these groups to obtain the appropriate error probabilities for all codes that interest us. To assess their impact, we note that in practice for a given message of size k (i.e. a given code dimension), a target error probability P_T is set, and the length (equivalently the overhead) is chosen to be the smallest value for which the actual error probability stays below P_T. So the cost of the speed-up obtained by AG-codes over RS-codes can be measured by how much extra overhead is required to obtain a certain target error probability P_T (or equivalently how much smaller the rate of the code needs to be).

Below are some graphs giving the required overhead for different channel erasure probabilities p and target error probabilities P_T. We use RS-codes over $\mathbb{F}_{256}$, and AG-codes over $\mathbb{F}_{16}$, each time choosing the code with the smallest genus enabling us to achieve the required length.

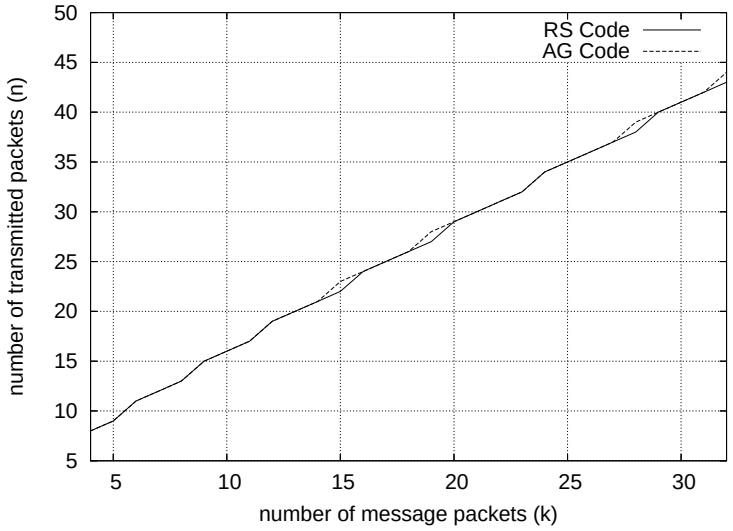


Figure 4.2: Required length to achieve a target error probability of $P_T = 10^{-3}$ on a channel with erasure probability $p = 0.1$.

The graph below presents the same data as Figure 4.2, but in terms of rate rather than length.

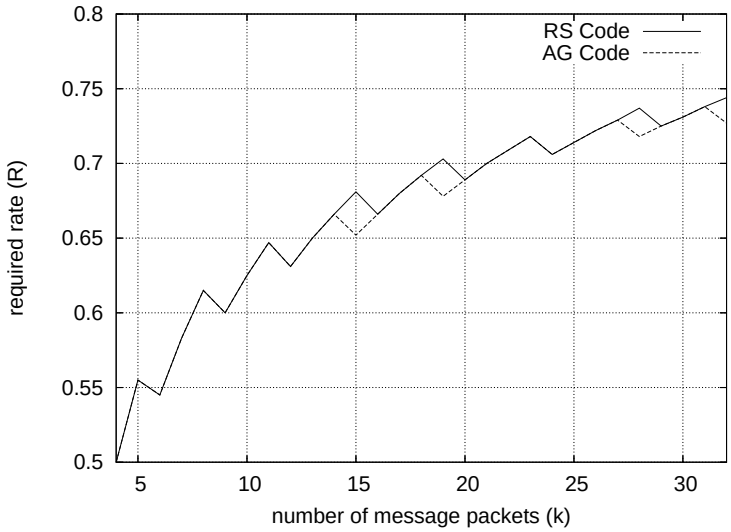


Figure 4.3: Required rate to achieve a target error probability of $P_T = 10^{-3}$ on a channel with erasure probability $p = 0.1$.

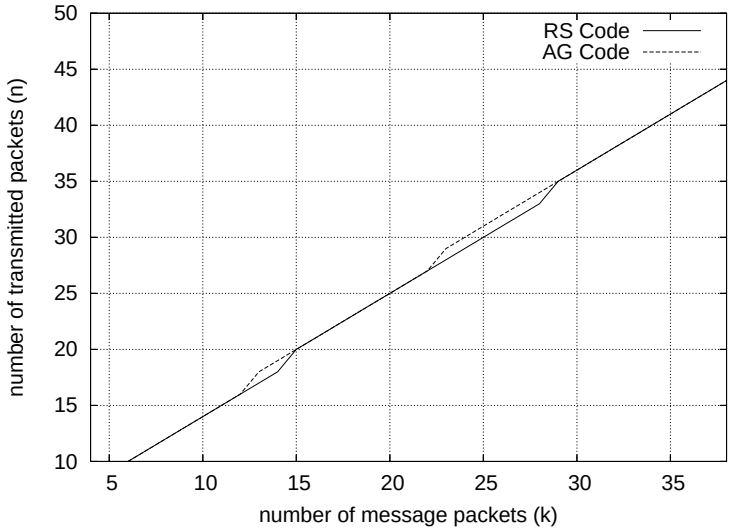


Figure 4.4: Required length when $P_T = 10^{-6}$ and $p = 0.01$.

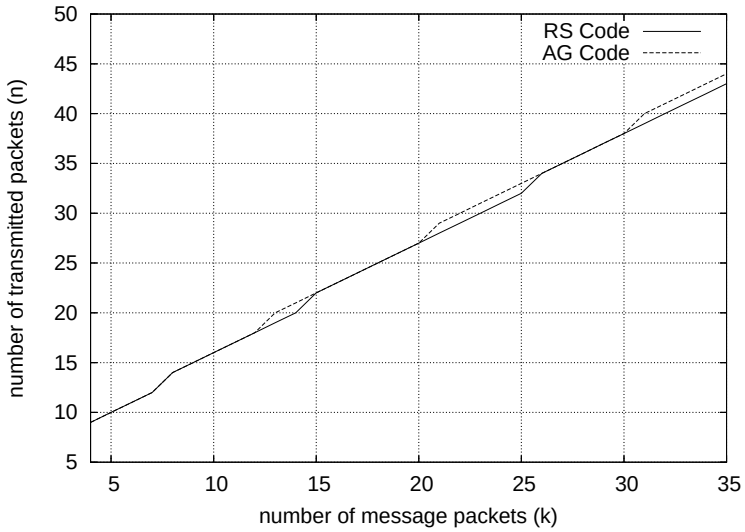


Figure 4.5: Required length when $P_T = 10^{-9}$ and $p = 0.01$.

We observe that the AG-code does not require us to transmit many more elements than the RS-code in order to achieve the target error probability, and in fact in most cases requires the same amount. So in this sense we can argue that the drawback of higher error probabilities is not a large one. As explained earlier, we were motivated by specific transmission problems and the parameters in the graphs above are chosen to reflect these.

4.6 Interleaved Vector-Matrix Multiplication

Since we are considering applications of AG-codes, we will also present some implementation properties quantifying the theoretical speed-ups that smaller fields enable. The aim of this section is to describe the interleaving technique of [10]. This makes the encoding and decoding processes for $\mathbb{F}_{2^\ell}$-codes faster, by making use of the fact that computers can perform many bit operations in a single cycle. This parallelism is utilized to encode many message vectors concurrently.

It is important to note that this does not improve the complexity (i.e. the asymptotic behavior), but does nonetheless make things faster for the lengths in which we are interested.

4.6.1 The Regular Representation

The aim in this subsection is to reduce additions and multiplications in $\mathbb{F}_{2^\ell}$ to additions and multiplications of binary vectors and matrices. Throughout we let $q = 2^\ell$. First recall that $\mathbb{F}_q$ is a vector space of dimension ℓ over $\mathbb{F}_2$. Throughout this section we fix an arbitrary basis $V = \{v_1, \ldots, v_\ell\}$ of $\mathbb{F}_q$ over $\mathbb{F}_2$. V establishes a canonical bijection between $\mathbb{F}_q$ and $\mathbb{F}_2^\ell$:

Definition 4.21. Given a basis V of $\mathbb{F}_q$ over $\mathbb{F}_2$, we let $\sigma : \mathbb{F}_q \to \mathbb{F}_2^\ell$ be the bijection defined as follows: If $\gamma = \sum_{i=1}^{\ell} a_i \cdot v_i$ then

$$\sigma(\gamma) = (a_1, \ldots, a_\ell). \tag{4.28}$$

Notice that σ is additive with respect to component-wise addition in $\mathbb{F}_2$. So addition in $\mathbb{F}_q$ is reduced to adding binary vectors. To deal with multiplication in $\mathbb{F}_q$ we can also express field elements as $\ell \times \ell$ matrices:

Definition 4.22. Given a basis V of $\mathbb{F}_q$ over $\mathbb{F}_2$, we let $\tau : \mathbb{F}_q \to \mathbb{F}_2^{\ell \times \ell}$ be the mapping defined as follows:

$$\tau(\gamma) = \begin{pmatrix} \sigma(v_1 \cdot \gamma) \\ \vdots \\ \sigma(v_\ell \cdot \gamma) \end{pmatrix}. \tag{4.29}$$

It can then easily be checked that this reduces multiplications in $\mathbb{F}_q$ to binary vector-matrix multiplications, i.e., the following proposition holds:

Proposition 4.23. *For any $\gamma, \mu \in \mathbb{F}_q$ we have*

$$\sigma(\gamma \cdot \mu) = \tau(\gamma) \cdot \sigma(\mu). \tag{4.30}$$

4.6.2 Interleaving

We are studying the problem of transmitting a file over a packet network modeled as an erasure channel. We suppose that the packet size L is fixed (so L bit packets correspond to the Q-ary erasure channel with $Q = 2^L$). The most obvious way to send a file over such a channel is to use an $[n, k]_Q$-code (so that each packet can be identified with a field element). Therefore a file consisting of k packets would be identified with a message vector and encoded to a codeword of n packets, which would then be transmitted. This is however highly impractical for large Q since performing the additions and multiplications in $\mathbb{F}_Q$ becomes extremely slow.

Instead we use a smaller field $\mathbb{F}_q$ and *interleave* many codewords within the packets. More precisely, suppose that $q = 2^\ell$ and $Q = 2^L$ and also suppose for simplicity that ℓ divides L, with $L = b\ell$. We then use an $[n, k]_q$-code $\mathcal{C}$.

One packet consists of $L = b\ell$ bits. We can arrange these in a $b \times \ell$ matrix, each row of which can be interpreted as an element of $\mathbb{F}_q$ (using the bijection σ defined above):

$$\text{1 packet} \longleftrightarrow \begin{pmatrix} g_{11} & \cdots & g_{1\ell} \\ \vdots & & \vdots \\ g_{b1} & \cdots & g_{b\ell} \end{pmatrix} \in \mathbb{F}_2^{b \times \ell} \overset{\sigma}{\longleftrightarrow} \begin{pmatrix} \gamma_1 \\ \vdots \\ \gamma_b \end{pmatrix} \in \mathbb{F}_q^b. \tag{4.31}$$

Now k packets can be concatenated, leading to the following interpretation:

$$k \text{ packets} \longleftrightarrow \overline{M} \in \mathbb{F}_2^{b \times k\ell} \overset{\sigma}{\longleftrightarrow} M \in \mathbb{F}_q^{b \times k} \tag{4.32}$$

(each packet corresponds to a column of M). $\overline{M}$ is the "binary version" of M, obtained by replacing each entry γ by $\sigma(\gamma)$. Throughout this section, for any matrix P over $\mathbb{F}_q$ we will write its binary version as $\overline{P}$ (the two can be linked either through σ or τ).

We compute the encoding by interpreting each row of M as a message vector for our code $\mathcal{C}$, which will get post-multiplied by G to obtain a codeword. So the encoding of the b message vectors in M consists of computing the $b \times n$ matrix C with

$$C = M \cdot G, \tag{4.33}$$

so each row of C is a codeword. This matrix multiplication could be done with standard finite field arithmetic, but the idea in [10] is to perform this as a multiplication of *binary* matrices.

The data to be encoded is given binary form (i.e. as $\overline{M}$). Rather than converting it to elements of $\mathbb{F}_q$ (to M) which we would then multiply by the generator matrix $G \in \mathbb{F}_q^{k \times n}$, instead we store G in its binary form $\overline{G}$ and then perform the binary multiplication.

We construct $\overline{G} \in \mathbb{F}_2^{k\ell \times n\ell}$ by replacing each entry γ of $G \in \mathbb{F}_q^{k \times n}$ with the $\ell \times \ell$ matrix $\tau(\gamma)$. We then compute the matrix $\overline{C} \in \mathbb{F}_2^{b \times n\ell}$ as follows:

$$\overline{C} = \overline{M} \cdot \overline{G}. \tag{4.34}$$

Just as $\overline{M}$ was identified with k packets, we interpret $\overline{C}$ as n packets which can then be transmitted. Likewise, just as b message vectors were interleaved into the k packets of $\overline{M}$, b codewords are interleaved into the n packets of $\overline{C}$. Notice that the link between the $\mathbb{F}_q$-matrix and its binary version is established through σ for M and C, and through τ for G.

Multiplying binary matrices can be done by XOR'ing entire columns. The key point for practical applications is that many bits can be XOR'ed in a single CPU cycle (how many depends on how big the registers of the specific machine are). So while this does not improve the asymptotic running time (the number of bits that can be XOR'ed in a single operation is of course constant), it can nonetheless make things much faster for a fixed set of parameters. We will refer to one such column (i.e. an element of $\mathbb{F}_2^b$) as a *symbol*.

The algorithm can be described as follows:

Algorithm 4.3: BINARY MULTIPLY BY XOR'ING COLUMNS
Input: A binary $b \times k\ell$ matrix $\overline{M}$, a binary $k\ell \times n\ell$ matrix $\overline{G}$.
Output: $\overline{C} = \overline{M} \cdot \overline{G}$.

```
Set C̄ to the b × nℓ zero matrix
for i = 1, ..., kℓ do
    for j = 1, ..., nℓ do
        if (Ḡ_ij = 1) then
            ( Column j of C̄ ) ← ( Column j of C̄ ) XOR ( Column i of M̄ )
        end if
    end for
end for
return C̄.
```

The number of XORs of symbols (i.e. columns) that needs to be performed is equal to the number of ones in $\overline{G}$.

So the interleaving technique for multiplying two matrices over $\mathbb{F}_q$ involves interpreting them in their binary forms, which can then be multiplied using Algorithm 4.3 by XOR'ing symbols.

To summarize the encoding process:

1. The generator matrix is stored in its binary form $\overline{G} \in \mathbb{F}_q^{k\ell \times n\ell}$.
2. We interpret our k packets (consisting of $kL = kb\ell$ bits) as a matrix $\overline{M} \in \mathbb{F}_2^{b \times k\ell}$
3. Compute $\overline{C} = \overline{M} \cdot \overline{G}$ using the algorithm above (so this is done exclusively through XORs of symbols).

4. $\overline{C} \in \mathbb{F}_2^{b \times n\ell}$ then contains the n encoded packets to be transmitted.

4.6.3 Encoding Time

We assess the running time of this algorithm in terms of the number of XORs of symbols per output symbol produced. We assume that our code is systematic (indeed any code can be brought into systematic form [51]), so its generator matrix G can be written as

$$G = \big(I_k \mid A\big), \tag{4.35}$$

where A is a $k \times (n-k)$ matrix in $\mathbb{F}_q$. We let $\overline{A}$ be the binary version of A (replacing each field element γ by the $\ell \times \ell$ binary matrix $\tau(\gamma)$).

The systematic packets (the first k) of course do not need to be computed, though we still count them as output packets. We need only compute $\overline{M} \cdot \overline{A}$, and so the number of XORs of symbols that needs to be performed is equal to the number of ones in the $k\ell \times (n-k)\ell$ matrix $\overline{A}$ (see the algorithm above). We expect about half of its entries to be 1 which leads to an expectation of $\frac{k(n-k)\ell^2}{2}$ XORs of symbols. Since the total number of symbols produced is $n\ell$, the number of XORs per output symbol is

$$\frac{k\ell(n-k)}{2n}. \tag{4.36}$$

Notice that this is proportional to ℓ, so a smaller field size yields an improvement in the theoretical running time (the field size is $q = 2^\ell$).

For comparison, encoding without this XOR'ing technique would involve working with operations over $\mathbb{F}_q$. More precisely we would need to multiply the message $M \in \mathbb{F}_q^{b \times k}$ by $A \in \mathbb{F}_q^{k \times (n-k)}$ from (4.35). So this would require $bk(n-k)$ multiplications and $b(k-1)(n-k)$ additions over $\mathbb{F}_q$.

4.6.4 Decoding Time

The decoding time depends not only on the parameters of the code, but also on how many systematic packets were erased. The decoding process can be described as follows. Suppose that the n encoded packets were transmitted, and that e of the systematic positions were erased. We consider the submatrix D of A whose rows correspond to the positions of the erased systematic packets, and whose columns correspond to the positions of the intact (non-erased) redundant packets. The decoding is successful if and only if the rank of D is e. If so, then e columns of D are calculated such that the submatrix E of D formed by these columns is invertible, and the corresponding intact redundant packets are marked (these e marked packets form a $b \times e$ matrix S over $\mathbb{F}_q$, or equivalently, through σ, a $b \times e\ell$ binary matrix $\overline{S}$).

We then let T denote the $b \times (k-e)$ $\mathbb{F}_q$-matrix formed by the intact systematic packets. We let J be the $(k-e) \times e$ submatrix of A whose rows correspond to the intact systematic packets, and whose columns correspond to the marked redundant packets. We then use the interleaving technique (Algorithm 4.3) a first time to compute TJ, and a second time to compute $(S - TJ)E^{-1}$.

The decoding therefore consists of the following steps:

1. Compute E^{-1}. We call this the *equation solving step*.

2. Compute TJ. Finding T and J does not require any work since they are just submatrices of known matrices. We multiply the matrices in their binary forms using Algorithm 4.3. We consider the $b \times (k-e)\ell$ binary version $\overline{T}$ of T (through σ), and the $(k-e)\ell \times e\ell$ binary version $\overline{J}$ of J (through τ), and compute $\overline{T} \cdot \overline{J}$.

3. Compute the $b \times e\ell$ binary matrix $\overline{R} = \overline{S} - \overline{T} \cdot \overline{J}$.

4. Perform the multiplication $\overline{R} \cdot \overline{E^{-1}}$ to obtain the e systematic packets that were erased.

Step 1 is done through Gaussian elimination, which requires $O(e^3)$ field operations. As a rule of thumb, if the symbols are large, then the running time of this step is amortized over the computation of the XORs. However, if e is large, or if the symbols are small, then this step may add significantly to the decoding time.

As for the encoding, we assess the running time of the remaining steps as the number of XORs of symbols to produce an output symbol. The number of XORs for step 2 is equal to the number of ones in $\overline{J}$. Again since we expect half of the entries of this $(k-e)\ell \times e\ell$ matrix to be ones, we obtain a total of $\frac{(k-e)e\ell^2}{2}$ XORs for step 2. Step 3 involves adding two $b \times e\ell$ matrices, which requires us to simply XOR the columns one by one, leading to $e\ell$ XORs. Finally step 4 involves another matrix multiplication. We expect half of the entries of the $e\ell \times e\ell$ matrix $\overline{E^{-1}}$ to be ones, which leads us to a total of $\frac{e^2\ell^2}{2}$ XORs.

The (successful) decoder produces k packets, i.e. $k\ell$ symbols, so combining everything, we need

$$\left(\frac{(k-e)e\ell^2}{2} + e\ell + \frac{e^2\ell^2}{2}\right) \cdot \frac{1}{k\ell} = \frac{ek\ell + 2e}{2k} \tag{4.37}$$

XORs per output symbol, to which we must add the time taken by the equation solving step.

4.7 Implementations

As explained in the introduction, the work in this chapter was motivated by practical needs, so we include some implementations to illustrate the speed-ups predicted in theory.

We will focus in this section on the following transmission problem: a given file of size up to 64 kB is to be transmitted over an impaired packet network, where each packet has a payload of 1 kB. We compare the performances of RS and AG-codes, which in both cases are implemented using the interleaving technique of Section 4.6.

Our RS-codes are constructed over the field $\mathbb{F}_{256}$, and our AG-codes over $\mathbb{F}_{16}$. We implemented the encoding and decoding algorithms in C (compiled with `gcc`, and `gcc -O3`) and ran them on an AMD Athlon MP 2400+ 2 Ghz processor with 1GB of RAM and 256 kB of cache.

4.7.1 Encoding Bit Rates

We saw in the previous section that there is a theoretical speed-up factor of 2 for the encoding. However we found that in practice the speed-ups were in fact larger. This is true with no optimization, and the effect is amplified even more when optimization options (`gcc -O3`) are set on the compiler (see the graphs below).

This could be due to many reasons, such as more efficient caching, as larger symbols are XOR'ed together, but less often. These bit rates and ratios could of course change depending on implementation and on which machine they are run. The results were nonetheless useful for the context in which we were interested.

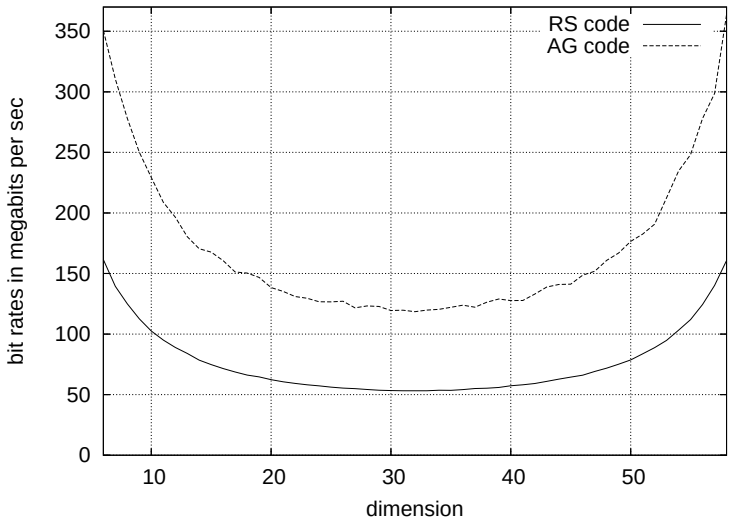


Figure 4.6: Encoding bit rates of RS and AG codes of length 64, with no optimization (`gcc`).

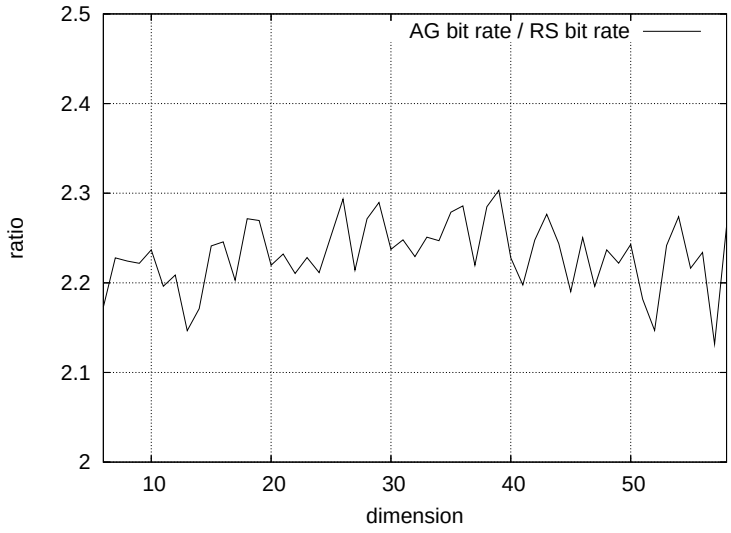


Figure 4.7: The ratio between the encoding bit rates from Figure 4.6.

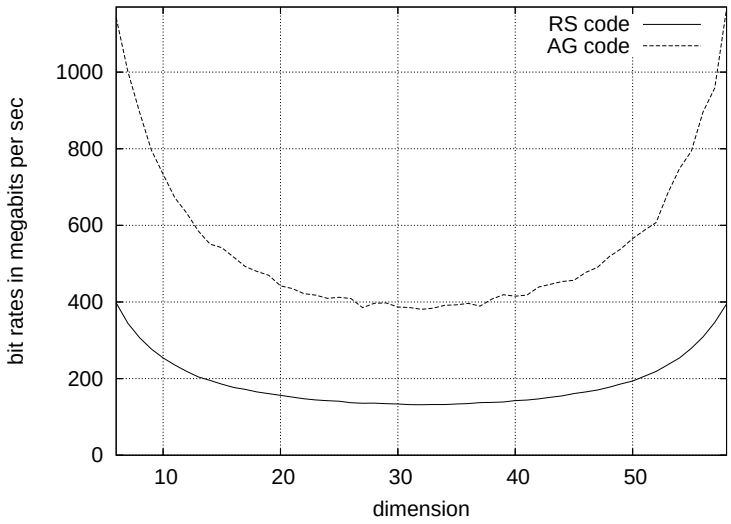


Figure 4.8: Encoding bit rates of RS and AG codes of length 64, with `gcc -O3` optimization.

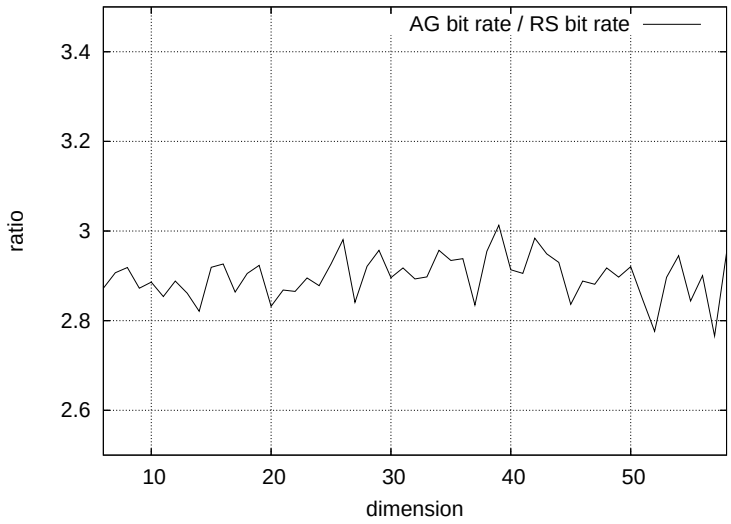


Figure 4.9: The ratio between the encoding bit rates from Figure 4.8.

4.7.2 Decoding Bit Rates

The theoretical decoding bit rates are a little more complicated. We showed in section 4.6.4 that we expected the decoder to need

$$\frac{ek\ell + 2e}{2k} \tag{4.38}$$

XORs of symbols per output symbol produced (where e is the number of erased systematic packets and the field size is 2^ℓ), plus time taken by the equation solving step (see section 4.6.4). The latter is essentially

$O(e^3)$, and since the smaller field size does not lead to the same improvements for this step as for the matrix multiplications, we expect the gap between AG and RS codes to be smaller when e is large.

In our experiments we supposed a "worst-case scenario", namely that there are $n - k$ erasures, of which as many as possible occur in the systematic packets. Formally this means that

$$e = \min(n - k, k). \tag{4.39}$$

We then make the erasures occur uniformly at random among the appropriate sets of packets.

The graphs below show the decoding bit rates under these conditions for AG and RS-codes.

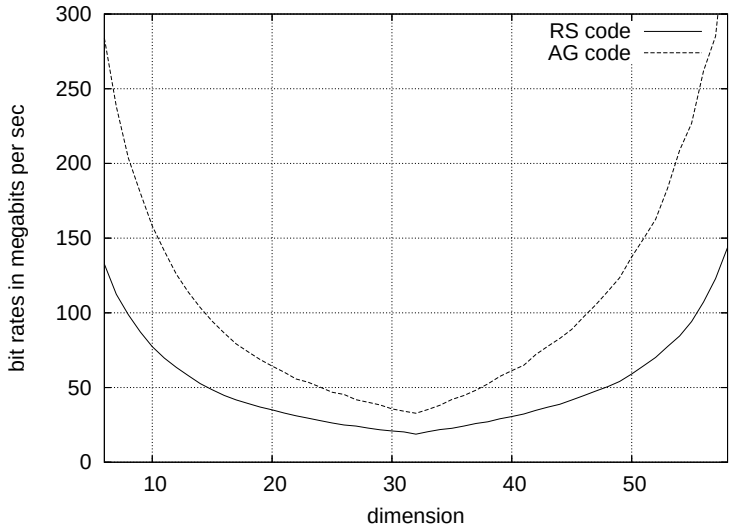


Figure 4.10: Decoding bit rates of RS and AG codes, with no optimization.

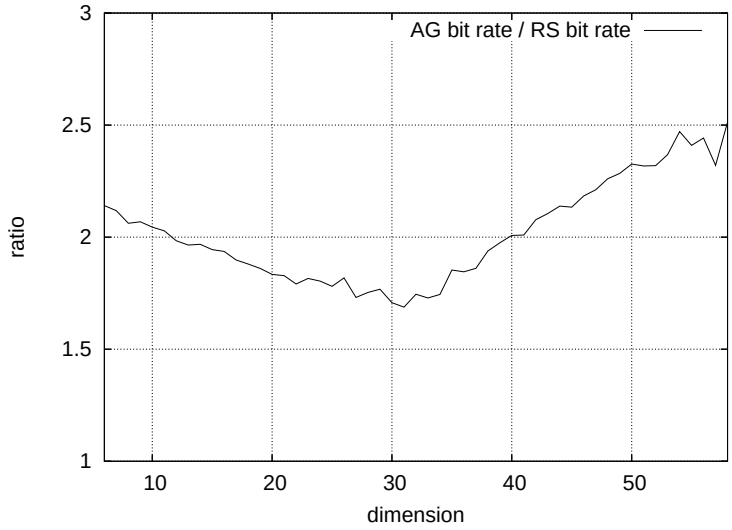


Figure 4.11: The ratio between the decoding bit rates from Figure 4.10.

We see that the ratio in Figure 4.10 is smaller when the rate gets close to $1/2$. This is probably due to the fact that these are the rates for which e is largest, and so as explained above the equation solving step takes a bigger share of the running time which reduces the difference between the two codes.

As for the encoding bit rate, the `-O3` optimization amplifies the gains that AG make over RS codes, as shown in the graphs below:

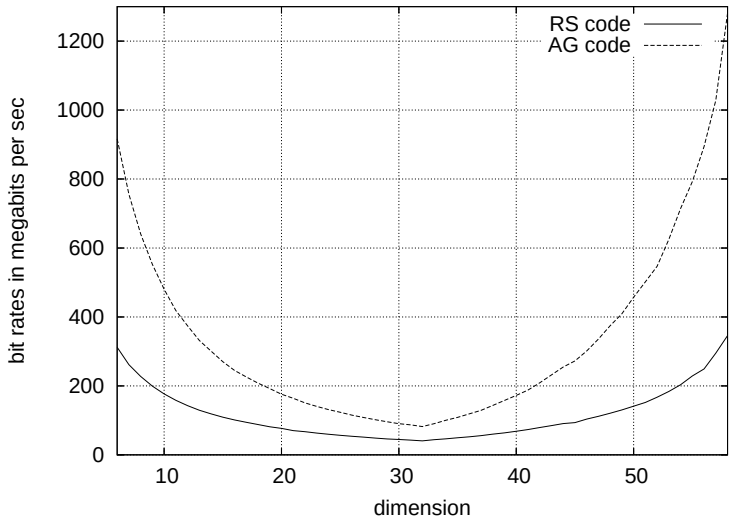


Figure 4.12: Decoding bit rates of RS and AG codes of length 64, with `gcc -O3` optimization.

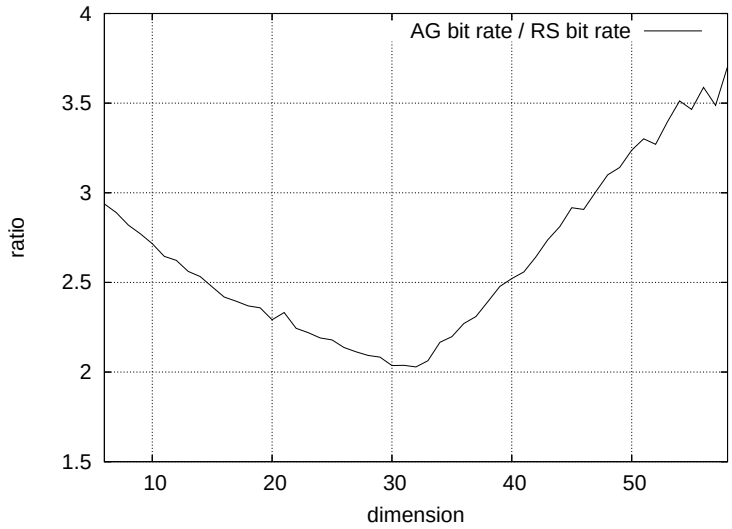


Figure 4.13: The ratio between the decoding bit rates from Figure 4.12.

The generator matrix of an RS-code can actually be expressed as a Cauchy matrix, which means that the equation solving step can be done faster ($O(e^2)$). This is the principle of *Cauchy Codes*, see [10]. With this improved decoding for RS-codes we obtain the following bit rates:

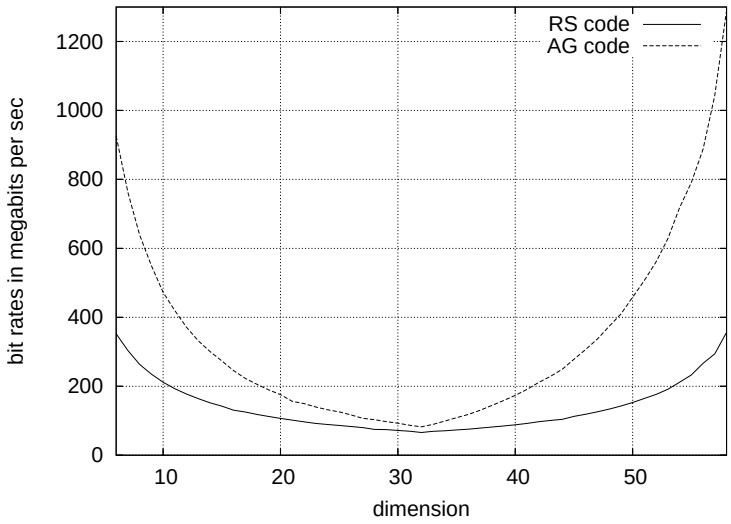


Figure 4.14: Decoding bit rates of RS (faster equation solving) and AG codes, with `-O3` optimization.

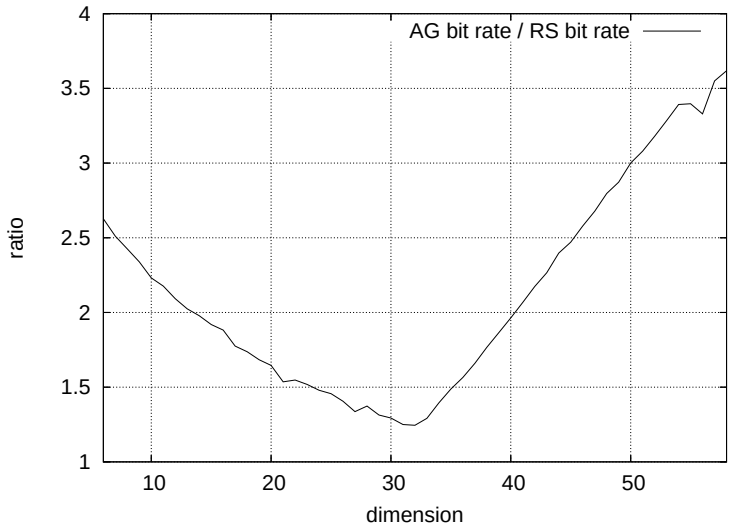


Figure 4.15: The ratio between the decoding bit rates from Figure 4.14.

As expected the gains are not quite as good, but we still get improvements for all dimensions, which get quite large when e moves aways from its maximum value.

4.8 Conclusion

For applications requiring very short blocks, AG and RS-codes become competitive solutions to protect data against packet loss. There is a strong argument to be made that AG-codes are in many cases the preferable option. Their key advantage is the use of smaller fields for a given length, which translates to faster encoding

and decoding times. Furthermore, the speed-ups predicted in theory seem to actually be amplified in practice. Although AG-codes do have higher error probabilities, we developed an algorithm to compute these and found that in many situations the consequences are in fact minor.

We conclude by saying that although AG-codes are most famous for their asymptotic properties, it seems that it is for very short lengths that they offer the greatest prospects for practical exploitations. The short AG-codes presented in this chapter are being used commercially for video delivery.

Chapter 5

Expander graphs

5.1 Introduction

Expander graphs and their constructions have been investigated since the 1970's. Their remarkable properties have led to applications in very diverse areas of computer science and discrete mathematics (coding theory, network design, cryptography, complexity and others).

There are different ways to define graph expansion, all of which can be shown to be related. Intuitively, a graph is a good expander if it is highly connected, meaning that all not-too-large sets of vertices have many neighbors. This is clearly easier to achieve with graphs of larger degree, and the challenge is to construct good expanders *of a given constant degree*. Perhaps surprisingly, it was shown [59] that a randomly chosen graph will have these properties with high probability. Explicit constructions are however more difficult to achieve.

In this work we will mainly be concerned with the algebraic characterization called *spectral expansion*, which measures the expansion of a graph by looking at its spectrum (more specifically the second largest eigenvalue). This will enable us to use standard tools from linear algebra to study expansion properties. It also directly governs the *mixing rate* of a graph, namely the speed at which a random walk on the graph will converge to its stationary distribution.

In 1986 Alon [2] gave an upper bound on the spectral expansion that can be achieved by an infinite family of graphs. Graphs reaching this bound are referred to as *Ramanujan graphs*, and were first explicitly constructed by Margulis [54] and independently by Lubotzky, Phillips and Sarnak [44].

More recently, Reingold, Vadhan and Wigderson [61] introduced the *zig-zag product*, which enables an elegant recursive construction. Although the resulting graphs are not Ramanujan, the construction is remarkable in that its analysis effectively relies only on linear algebra which makes it not only easier to follow but also somewhat more intuitive than any of the previous constructions.

The aim of this chapter is to introduce the necessary background for Chapter 6. Sections 5.2 to 5.5 will present some preliminaries, definitions and standard results on expander graphs. In Section 5.6 we describe some graph products and operations which will be used in the next chapter. Section 5.7 gives some results on the spectral expansion of biregular bipartite graphs, which, although straightforward adaptations of their non-bipartite counterparts, do not appear to feature prominently in the literature.

5.2 Background

Definition 5.1. We will use the following graph theory conventions:

• An *undirected graph* G is a pair (V, E) where V is a finite set (the set of vertices) and $E \subseteq V \times V$ is a symmetric relation on V (the set of edges). Note that self loops are allowed.

• An *undirected multigraph* G is a pair (V, E) where V is a finite set (the set of vertices) and $E \subseteq V \times V$ is a multiset (the set of edges) such that

$$(x, y) \in E \implies (y, x) \in E, \text{ with the same multiplicity.}$$

Note that multiples edges and multiple self loops are allowed.

• The *size* of a graph $G = (V, E)$ is defined as the number of vertices $|V|$.

• For any subset $S \subseteq V$ of vertices, the *set of neighbors* of S, denoted $N(S)$, is defined as

$$N(S) = \big\{v \in V \mid \exists s \in S : (s, v) \in E\big\}.$$

• The *degree* of a vertex is the number of incident edges (each self loop is counted as a single edge).

• A graph is said to be d*-regular* if all its vertices have degree d.

• A graph $G = (V, E)$ is said to be *bipartite* if there are two disjoint subsets $S, T \subseteq V$ with $V = S \cup T$ and for any $s_1, s_2 \in S$, $t_1, t_2 \in T$ we have

$$(s_1, s_2) \notin E, \text{ and } (t_1, t_2) \notin E.$$

We will refer to the elements S and T as the *left* and *right* vertices.

• Let $G = (V, E)$ be a bipartite graph with left and right vertex sets S and T. G is said to be *biregular* if there are ℓ, r for which all left vertices have degree ℓ and all right vertices have degree r.

ℓ and r are called the *left degree* and *right degree* respectively. Notice that

$$\ell \cdot |S| \;=\; r \cdot |T|. \tag{5.1}$$

• A *path of length* n is a sequence $v_0, \ldots, v_n$ of vertices, with $(v_{i-1}, v_i) \in E$ for each $i = 1, \ldots, n$.

• A *cycle of length* n is a path of length n in which $v_0 = v_n$.

• The *distance* between two vertices u, v is the length of the shortest path from u to v.

In this work we will be dealing mostly with regular undirected multigraphs. Unless otherwise stated, a *graph* will refer to an undirected multigraph. We will refer to a d-regular graph of size n as an $[n, d]$-graph. We will also be dealing with *biregular bipartite graphs*, whose properties are described in Section 5.7.

Definition 5.2. We will also use the following linear algebra notation:

• For any $n \in \mathbb{N}$, we define the set $[n]$ as

$$[n] = \{1, \ldots, n\}.$$

• A vector $x \in \mathbb{R}^n$ is a *probability distribution* (or *probability vector*) if $\forall i \in [n] : x_i \geq 0$, and

$$\sum_{i=1}^{n} x_i = 1.$$

• The *inner product*

$$\langle \cdot, \cdot \rangle : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$$

is defined as

$$\langle x, y \rangle = \sum_{i=1}^{n} x_i \cdot y_i.$$

• The *norm* of a vector $x \in \mathbb{R}^n$ is defined as

$$\|x\| = \sqrt{\langle x, x \rangle}.$$

• Two vectors $x, y \in \mathbb{R}^n$ are said to be *orthogonal* (or *perpendicular*), if

$$\langle x, y \rangle = 0.$$

We write this as $x \perp y$.

• Two vectors $x, y \in \mathbb{R}^n$ are said to be *parallel* if there is $0 \neq \beta \in \mathbb{R}$ such that

$$y = \beta x.$$

We write this as $x \parallel y$.

• A set of vectors $\{v_1, \ldots, v_n\}$ is said to be *orthonormal* if they are pairwise orthogonal, and $\|v_i\| = 1$ for each $i = 1, \ldots, n$.

• $x \in \mathbb{R}^n$ is said to be an *eigenvector* of a matrix $M \in \mathbb{R}^{n \times n}$ if there is an element $\lambda \in \mathbb{R}$ for which

$$Mx = \lambda x.$$

λ is then called the *eigenvalue* of M corresponding to x.

• The set of eigenvalues of a matrix M is called its *spectrum*, and is denoted by $\text{Spec}(M)$.

• A matrix is said to be *stochastic* if all its columns are probability vectors. It is *doubly stochastic* if all its rows and all its columns are probability vectors.

- $I_n \in \mathbb{R}^{n \times n}$ denotes the $n \times n$ identity matrix.
- 1_n denotes the vector in $\mathbb{R}^n$ whose entries are all 1.
- $1_n^{\parallel}$ denotes the space of vectors in $\mathbb{R}^n$ generated by 1_n:

$$1_n^{\parallel} = \{\beta \cdot 1_n \mid \beta \in \mathbb{R}\}.$$

- $1_n^{\perp}$ denotes the space of vectors in $\mathbb{R}^n$ that are orthogonal to 1_n:

$$1_n^{\perp} = \{v \in \mathbb{R}^n \mid \langle v, 1_n \rangle = 0\}.$$

Notice that $1_n^{\parallel}$ and $1_n^{\perp}$ have respective dimensions 1 and $n-1$, and that they have only the zero vector in common. We will use the terminology from [94] and call elements of $1_n^{\parallel}$ *uniform* and elements of $1_n^{\perp}$ *anti-uniform*. We have

$$\mathbb{R}^n = 1_n^{\parallel} \oplus 1_n^{\perp},$$

which means that any vector $w \in \mathbb{R}^n$ can be uniquely decomposed as $w = w^{\parallel} + w^{\perp}$ where $w^{\parallel}$ is uniform and $w^{\perp}$ is anti-uniform.

Proposition 5.3. *We have the following standard results:*

- ***The Cauchy-Schwarz inequality:*** *For any $u, v \in \mathbb{R}^n$:*

$$\big|\langle u, v \rangle\big| \leq \|u\| \cdot \|v\|. \tag{5.2}$$

- **The triangle inequality:** For any $u, v \in \mathbb{R}^n$:

$$\|u + v\| \leq \|u\| + \|v\|. \tag{5.3}$$

The *adjacency matrix* is a very natural way to represent a graph, and provides the link between graph theory and linear algebra.

Definition 5.4. The *adjacency matrix* $\mathrm{Adj}(A)$ of a graph A with vertex set $[n]$ is the $n \times n$ matrix such that $\mathrm{Adj}(A)_{ij}$ is equal to the number of edges between vertices i and j.

Notice that when A is undirected, $\mathrm{Adj}(A)$ is symmetric. When a graph A is regular, the *normalized adjacency matrix* (defined below) will be a very important tool to represent A. In fact, we will often identify a regular graph with its normalized adjacency matrix.

Definition 5.5. The *normalized adjacency matrix* of a d-regular graph A with vertex set $[n]$ is the $n \times n$ matrix

$$M_A = \frac{1}{d} \cdot \mathrm{Adj}(A). \tag{5.4}$$

Notice that since $\mathrm{Adj}(A)$ is symmetric, so is M_A. Furthermore, since each vertex has degree d we have

$$\forall i \in [n] : \sum_{j=1}^{n} \mathrm{Adj}(A)_{ij} = d,$$

and therefore for each row i

$$\sum_{j=1}^{n}(M_A)_{ij} = 1.$$

Furthermore since M_A is symmetric, we can deduce that it is a doubly stochastic matrix. We now give some more properties of M_A.

Theorem 5.6. *Any $n \times n$ real symmetric matrix has n real eigenvalues and n orthonormal eigenvectors.*

Proof: This is a standard result, see for example [7]. ■

So in particular, since the normalized adjacency matrix of a graph is real and symmetric, it has n real eigenvalues which we write (in decreasing order) $\lambda_0 \geq \ldots \geq \lambda_{n-1}$, and to which correspond respectively n eigenvectors $v_0, \ldots v_{n-1}$ with

$$\langle v_i, v_j \rangle = \delta_{ij}.$$

For a graph A, when we refer to the *spectrum of A* we will mean the spectrum of M_A.

Proposition 5.7. *Let A be a regular graph and let M_A be its normalized adjacency matrix. If $\lambda_0 \geq \ldots \geq \lambda_{n-1}$ are its (ordered) eigenvalues with corresponding orthonormal eigenvectors $v_0, \ldots, v_{n-1}$, then*

$$\lambda_0 = 1 \qquad \textit{and} \qquad v_0 = \frac{1_n}{\sqrt{n}}.$$

Furthermore, for all $i = 1, \ldots, n-1$ we have

$$|\lambda_i| \leq 1.$$

Proof: This is a standard result, we take the proof from [94]. We will start by showing the second part, namely that $|\lambda_i| \leq 1 \ \forall i$. Let λ be any eigenvalue with corresponding eigenvector v. For any $j \in [n]$, we denote by $(v)_j$ the j^{th} component of v. Let $k \in [n]$ be an index for which $|(v)_k|$ is maximal:

$$|(v)_k| = \max_{j \in [n]} |(v)_j|.$$

Now since $M_A \cdot v = \lambda v$, we have in particular that $(M_A \cdot v)_k = (\lambda v)_k$, and therefore $|(M_A \cdot v)_k| = |(\lambda v)_k|$. Letting $a_{ij} = \text{Adj}(A)_{ij}$, this leads to

$$\Big|\sum_{j=1}^{n} a_{kj} \cdot (v)_j\Big| = |\lambda| \cdot |(v)_k|,$$

and therefore

$$\begin{aligned}
|\lambda| &= \frac{\left|\sum_{j=1}^{n} a_{kj} \cdot (v)_j\right|}{|(v)_k|} \\
&\leq \frac{\sum_{j=1}^{n} |a_{kj}| \cdot |(v)_j|}{|(v)_k|} && \text{(by the triangle inequality)} \\
&\leq \sum_{j=1}^{n} |a_{kj}| && \text{(since } |(v)_j| \leq |(v)_k| \text{ for all } j \in [n]\text{)} \\
&= 1 && \text{(since } M_A \text{ is doubly stochastic).}
\end{aligned}$$

It now just remains to be shown that 1 is an eigenvalue. This follows immediately from the fact that M_A is doubly stochastic: Taking the uniform vector 1_n we see that for all i

$$(M_A \cdot 1_n)_i = \sum_{j=1}^{n} a_{ij} = 1 = (1_n)_i,$$

and therefore $M_A \cdot 1_n = 1_n$. Normalizing this eigenvector gives us

$$\lambda_0 = 1, \qquad v_0 = \frac{1_n}{\sqrt{n}}.$$

■

The spectrum of a graph can tell us about its expansion properties. The *second eigenvalue* will be of particular interest:

Definition 5.8. Let A be a non-bipartite graph and let $\lambda_0 \geq \ldots \geq \lambda_{n-1}$ be its eigenvalues. The *second eigenvalue* of A is defined as

$$\lambda_A = \max\big(|\lambda_1|, |\lambda_{n-1}|\big).$$

So λ_A is the second largest eigenvalue in absolute value. Notice that from proposition 5.7, for any graph A we have $0 \leq \lambda_A \leq 1$. Definition 5.8 applies only to *non-bipartite* graphs. We will see in Section 5.7 the corresponding definition for bipartite graphs.

Proposition 5.9. *Let A be a regular graph and let $\lambda_0 \geq \ldots \geq \lambda_{n-1}$ be its eigenvalues. Then*

- *A is connected if and only if $\lambda_1 < 1$.*
- *A is bipartite if and only if $\lambda_{n-1} = -1$.*

Proof: See for example [94]. ■

The following characterization of the second eigenvalue of a graph will be very useful in the next chapter:

Theorem 5.10. *For any non-bipartite graph A we have*

$$\lambda_A = \max_{0 \neq x \in 1_n^{\perp}} \frac{\big|\langle M_A \cdot x, x\rangle\big|}{\langle x, x\rangle}.$$

Proof: See for example [94]. ■

5.3 Expander Graphs

As explained in the introduction, there are different ways of measuring graph expansion. The most intuitive ways are combinatorial, and we start with *edge expansion*. For a graph $A = (V, E)$ and a subset $S \subseteq V$ of vertices, we let $\overline{S}$ denote the complement of S in V, and define the *edge boundary* of S as the set of outgoing edges from S:

$$\partial S = E \cap \big(S \times \overline{S}\big).$$

Definition 5.11. A graph $A = (V, E)$ is said to be an *h-edge expander* if

$$\forall S \subseteq V : |S| \leq \frac{|V|}{2} \implies |\partial S| \geq h \cdot |S|.$$

We also define the *edge expansion parameter* of A as

$$h(A) := \min\big\{h \mid \text{A is an } h\text{-edge expander}\big\}.$$

So edge expansion requires that sets have many outgoing edges. This is closely related to the concept of *vertex expansion*:

Definition 5.12. A graph $A = (V, E)$ is said to be an *(α, β)-vertex expander* if

$$\forall S \subseteq V : |S| \leq \alpha \cdot |V| \implies |N(S)| \geq \beta \cdot |S|.$$

This is saying that any set S of vertices that is not too large "expands" into its neighborhood (i.e. $N(S) \geq \beta \cdot |S|$). Often α is set to $\frac{1}{2}$ which leads to the following common definition:

Definition 5.13. A graph $A = (V, E)$ is said to be a *β-vertex expander* if

$$\forall S \subseteq V : |S| \leq \frac{|V|}{2} \implies |N(S)| \geq \beta \cdot |S|.$$

We also define

$$\beta(A) := \max\big\{\beta \mid \text{A is a } \beta\text{-vertex expander}\big\}.$$

Expander graphs are sometimes said to be "highly connected", referring to the fact that sets of vertices have many neighbors. Although this definition has a clear visual interpretation (and goes well with the word *expander*), it is sometimes difficult to prove results relating to the expansion of specific graphs using edge or vertex expansion. Instead we will be mostly concerned with the following algebraic characterization of graph expansion:

Definition 5.14. A regular graph $A = (V, E)$ is said to be a *λ-spectral expander* if its second eigenvalue λ_A has the property that

$$\lambda_A \leq \lambda.$$

Recall from Theorem 5.7, that $0 \leq \lambda_A \leq 1$. The value $1 - \lambda_A$ is referred to as the *spectral gap*. A larger gap means better expansion. We will refer to d-regular λ-spectral expander of size n as an $[n, d, \lambda]$-graph. When we say that a graph is a *λ-expander* we mean that it is a λ-spectral expander.

The definitions above are all essentially measuring the same thing. The relationships between them are important, in the sense that it is often easier to analyze and construct graphs based on their spectral expansion, while some applications make direct use of their combinatorial expansion properties.

The relationship between edge and spectral expansion is captured in the following theorem [33]:

Theorem 5.15. *For any $[n, d, \lambda_A]$-graph A we have*

$$\frac{d(1-\lambda_A)}{2} \leq h(A) \leq d\sqrt{2(1-\lambda_A)}. \tag{5.5}$$

This was proved by Dodziuk [23] and independently by Alon-Milman [6] (see [33]). We can also relate vertex and spectral expansion as follows:

Theorem 5.16. *For any $[n, d, \lambda_A]$-graph A we have*

$$1 - 2\beta(A) \leq \lambda_A \leq \sqrt{1 - \frac{(\beta(A)-1)^2}{d^2 \cdot (8 + 4(\beta(A)-1)^2)}}. \tag{5.6}$$

The second inequality of (5.6) was proved by Alon in [2]. The first inequality follows from the fact that $h(a) \leq d \cdot \beta(A)$ and Theorem 5.15.

5.4 Random Walks

The behavior of a random walk on a given graph is strongly related to its expansion properties. Although we have at our disposal a wide range of algebraic tools to study the spectral expansion of a graph, random walks have the advantage of having a very appealing intuition. When we look at graph products it is often convenient to conceptualize a product of two graphs A and B in terms of how one step of a random walk on this product is constructed from steps of walks on A and B. In our proofs we will often supplement the calculations with a a description of what we are doing in terms of random walks. Furthermore, many of the practical applications of expander graphs in computer science explicitly use the mixing properties of expander graphs.

Using the normalized adjacency matrix M_A, we can analyze random walks in algebraic terms. When we start with an initial distribution $x_0 \in \mathbb{R}^n$ on the vertices of A, after one step of a random walk on A the distribution will be

$$x_1 = M_A \cdot x_0.$$

Likewise after t steps it will be

$$x_t = (M_A)^t \cdot x_0.$$

We will also refer to M_A as the *transition matrix* of A.

In any connected non-bipartite d-regular graph A, taking a random walk on its vertices starting from any initial distribution will converge to the uniform distribution $\frac{1_n}{n}$. The spectral expansion λ_A determines the speed of this convergence. The better the expansion properties of A, the faster a walk will converge to the uniform distribution. This is written more formally in the following theorem (see for example [33]):

Theorem 5.17. *Let A be a non-bipartite λ_A-spectral expander. Starting with any initial distribution $x_0 \in \mathbb{R}^n$ on the vertices of A, the distribution x_t after t steps of a random walk will satisfy:*

$$\left\| x_t - \frac{1_n}{n} \right\| \leq \lambda_A^t.$$

So we can say that the distribution converges exponentially fast to the uniform distribution, with base λ_A.

Proof: Let $\lambda_0 \geq \ldots \geq \lambda_{n-1}$ be the eigenvalues of A, with corresponding normalized eigenvectors $v_0, \ldots, v_{n-1}$. We know from Theorem 5.6 that these eigenvectors form an orthonormal basis of $\mathbb{R}^n$. We can write x_0 in this basis as

$$x_0 = \sum_{i=0}^{n-1} \alpha_i \cdot v_i.$$

We have:

$$\begin{aligned} x_t &= A^t \cdot x_0 \\ &= A^t \cdot \textstyle\sum_{i=0}^{n-1} \alpha_i \cdot v_i \\ &= \textstyle\sum_{i=0}^{n-1} \alpha_i \cdot \lambda_i^t \cdot v_i \\ &= \alpha_0 \cdot \lambda_0^t \cdot v_0 + \textstyle\sum_{i=1}^{n-1} \alpha_i \cdot \lambda_i^t \cdot v_i \\ &= \alpha_0 \cdot v_0 + \textstyle\sum_{i=1}^{n-1} \alpha_i \cdot \lambda_i^t \cdot v_i \qquad \text{(since } \lambda_0 = 0\text{).} \end{aligned} \tag{5.7}$$

Recall from Proposition 5.7 that $v_0 = 1_n/\sqrt{n}$. This means that

$$\alpha_0 = \langle x, v_0 \rangle = \frac{\sum_{i=1}^n x_i}{\sqrt{n}} = \frac{1}{\sqrt{n}} \qquad \text{(since } x \text{ is a distribution)}, \tag{5.8}$$

so that

$$\alpha_0 \cdot v_0 = \frac{1_n}{n}. \tag{5.9}$$

Continuing with (5.7), we have

$$\begin{aligned} \left\| x_t - \tfrac{1_n}{n} \right\| &= \left\| x_t - \alpha_0 \cdot v_0 \right\| \\ &= \left\| \textstyle\sum_{i=1}^{n-1} \alpha_i \cdot \lambda_i^t \cdot v_i \right\| \\ &= \sqrt{\textstyle\sum_{i=1}^{n-1} \alpha_i^2 \cdot \lambda_i^{2t}} \qquad \text{(since the } v_i\text{'s are orthonormal)} \\ &\leq \lambda_A^t \cdot \sqrt{\textstyle\sum_{i=1}^{n-1} \alpha_i^2} \\ &\leq \lambda_A^t \cdot \|x_0\| \\ &\leq \lambda_A^t \qquad \text{(since } x_0 \text{ is a distribution).} \end{aligned}$$

■

5.5 Families of Expander Graphs

As we have previously seen with codes, there are many applications of expanders in which we do not know beforehand the size of the required graph. So just as we had worked families of codes in Chapter 3, we can define families of graphs. It is much more convenient (and elegant) to construct families of graphs that display the desired properties, rather than ad-hoc constructions of good graphs of different sizes. Furthermore, when expanders are employed to show asymptotic results it often becomes necessary to work with infinite families of graphs.

Definition 5.18. A (fixed degree) family of graphs of degree d is a sequence $\{A_i\}_{i\in\mathbb{N}}$, where A_i is an $[n_i, d]$-graph and

$$\lim_{i\to\infty} n_i = \infty.$$

A family of graphs is said to be a λ-*expander family* if each A_i is a λ-spectral expander. A family is said to be an *expander family* if it is a λ-expander family for some $\lambda < 1$. Recall that a smaller second eigenvalue means better expansion, so it is desirable to construct λ-expander families for λ as small as possible. We have the intuition that it is easier to construct expanders of larger degree (the "high connectivity" can be more readily achieved with many edges), so the challenge is to build the best possible expanders *of a given degree* d.

The following theorem (stated in [2]) gives a lower bound on the best λ that can be achieved, and its relationship to the degree.

Theorem 5.19. *(Alon-Boppana). Let $\{A_i\}_{i\in\mathbb{N}}$ be a family of graphs of degree d. Then*

$$\lim_{i\to\infty} \lambda(A_i) \geq \frac{2\sqrt{d-1}}{d}.$$

This is sometimes referred to as the *Alon-Boppana bound*. It provided a benchmark against which one can measure how good a given family of expander graphs is. Graphs achieving this bound are referred to as *Ramanujan graphs*.

Example 5.20. For any $i \in \mathbb{N}^*$ we define $\mathbb{Z}_i$ as the ring of integers modulo i:

$$\mathbb{Z}_i = \mathbb{Z}/i\mathbb{Z}. \tag{5.10}$$

Some examples of explicit expander family constructions:

1. Let $V_i = \mathbb{Z}_i \times \mathbb{Z}_i$. Each vertex $(x, y) \in V_i$ has the following 4 neighbors:

   $$(x + y, y), \qquad (x - y, y), \qquad (x, x + y), \qquad (x, x - y).$$

   Then $A_i = (V_i, E_i)$ is an $[i^2, 4]$-graph, and $\{A_i\}_{i\in\mathbb{N}^*}$ is an expander family [78].

2. Let $V_i = \mathbb{Z}_i \times \mathbb{Z}_i$. Each vertex $(x, y) \in V_i$ has the following eight neighbors:

   $$\begin{array}{llll} (x + y, y), & (x - y, y), & (x, y + x), & (x, y - x), \\ (x + y + 1, y), & (x - y + 1, y), & (x, y + x + 1), & (x, y - x + 1). \end{array} \tag{5.11}$$

   Then $A_i = (V_i, E_i)$ is an $[i^2, 8]$-graph, and $\{A_i\}_{i\in\mathbb{N}^*}$ is an expander family [33] [78].

   This was the first construction of an explicit expander family, and is due to Margulis [52] (1973). His proof was existential in the sense that it did not provide an explicit bound on the expansion of the family. This was obtained later by Gabber and Galil [27].

3. Let p_i denote the i^{th} prime. We let $V_i = \mathbb{Z}_{p_i}^{\times}$, and the edge set $E_i \subseteq V_i \times V_i$ is defined as

$$E_i \;=\; \big\{(x, x^{-1}), (x, x+1), (x, x-1) \;\big|\; x \in V_i \backslash \{0\}\big\} \;\cup\; \big\{(0,0), (0,1), (0,-1)\big\}. \tag{5.12}$$

Then $A_i = (V_i, E_i)$ is a $[p_i, 3]$-graph, and $\{A_i\}_{i \in \mathbb{N}^*}$ is an expander family [33].

4. Let p and q be distinct primes, with $p \equiv q \equiv 1 \bmod 4$, and let u be an integer for which $u^2 \equiv -1 \bmod q$ (such a u always exists). It can be shown [44] that there are exactly $(p+1)$ 4-tuples (a_0, a_1, a_2, a_3) with

$$a_0^2 + a_1^2 + a_2^2 + a_3^2 = p, \tag{5.13}$$

and for which $a_0 > 0$ is odd, and a_1, a_2, a_3 are even. To each such tuple we associate the matrix

$$\begin{pmatrix} a_0 + ua_1 & a_2 + ua_3 \\ -a_2 + ua_3 & a_0 - ua_1 \end{pmatrix} \in \mathrm{PGL}(2, \mathbb{F}_q), \tag{5.14}$$

and let S be the set of these $p+1$ matrices. The Cayley graph A_{pq} of $\mathrm{PGL}(2, \mathbb{F}_q)$ with respect to S is then an $[N, p+1]$-graph, where

$$N = |\mathrm{PGL}(2, \mathbb{F}_q)| = q(q^2 - 1). \tag{5.15}$$

It can be shown [44] that

$$\lambda_{A_{pq}} = \frac{2\sqrt{p}}{p+1}. \tag{5.16}$$

So if we fix $p \equiv 1 \bmod 4$, take an infinite sequence $q_1 < q_2 < \ldots$ of primes for which $q_i \equiv 1 \bmod 4$, and let $A_i = A_{pq_i}$, then $\{A_i\}_{i \in \mathbb{N}^*}$ is a family of Ramanujan graphs of degree $(p+1)$.

This construction is due to Lubotzky, Phillips and Sarnak [43] [44] (1988). The term *Ramanujan graph* comes from this family whose analysis uses the Ramanujan conjecture. This was later extended by Morgenstern [56] to obtain constructions of $(q+1)$-regular Ramanujan graphs for all prime powers q.

5.6 Graph Products and Operations

Because we are working with multigraphs, the edges $E \subseteq V \times V$ form a *multiset* (a set in which elements can appear multiple times). It is often inconvenient to refer to edges as elements of this multiset. Instead, having a *labeling* of the edges allows for more concise notation.

We start this section by introducing labelings and the notation that follows, which we will then use to present some graph products.

5.6.1 Edge Labelings

A *labeling* of an $[n, d]$-graph A consists of assigning distinct labels to the edges leaving each vertex of A. The labels will be elements of a set L of size d. We often have $L = [d]$. In this case for a vertex $u \in [n]$, if the edge labeled $i \in L$ connects u to v then we can say that v is the i^{th} neighbor of u.

Each edge has two labels (one corresponding to each one of its vertices), and these labels may be different. This is saying that if v is the i^{th} neighbor of u then u may not be the i^{th} neighbor of v. We write this formally as follows:

Definition 5.21. Let $A = (V, E)$ be an $[n, d]$-graph on vertex set $[n]$. A *labeling* in L (so $|L| = d$) for a vertex $u \in [n]$ is a bijection

$$\mu_u : L \to N(u). \tag{5.17}$$

A labeling μ for the whole graph A consists of a labeling for each one of the n vertices

$$\mu = \{\mu_u \mid u \in [n]\}. \tag{5.18}$$

All our graphs will either have some arbitrary (but fixed) labeling, or an implicit labeling from their construction. This enables us to employ the following notation:

Definition 5.22. Let A be an $[n, d]$-graph on vertex set $[n]$, and with a labeling in $[d]$. For $u, v \in [n]$ and $i \in [d]$ we use the notation

$$v = u[i] \tag{5.19}$$

to denote the fact that v is the i^{th} neighbor of u (i.e. that $\mu_u(i) = v$).

Definition 5.23. A labeling of an $[n, d]$-graph A is said to be a *d-edge-coloring* if for each edge its two labels are identical. More formally:

$$\forall u \in [n],\ i \in [d] :\ u[i][i] = u. \tag{5.20}$$

Not all d-regular graphs have d-colorings. Finding a d-edge-coloring is equivalent to partitioning the vertices of A into d perfect matchings. So for example if n is odd then A does not have a d-edge-coloring. It turns out that determining whether a given graph has d-edge-coloring is NP-complete [32]. In this work when we refer to a *coloring* we mean an edge-coloring.

Definition 5.24. A labeling is said to be a *half-coloring* if for each color $i \in [d]$ there is a corresponding color $\rho(i) \in [d]$ for which any edge colored i at one end will be colored $\rho(i)$ at the other end. More formally, there is a mapping $\rho : [d] \to [d]$ that satisfies

$$\forall u \in [n],\ i \in [d] :\ u[\rho(i)][i] = u[i][\rho(i)] = u. \tag{5.21}$$

We refer to ρ as the *partner mapping*. Notice that ρ is an involution. So a d-edge-coloring is a special case of a half-coloring, in which ρ is the identity map. When a labeling is a half-coloring, the labels will also be referred to as *colors*.

Half-colorings will be of interest to us because for our analysis of the expansion properties of the derandomized tensor product in Chapter 6 it will be necessary and sufficient for the graphs involved to have half-colorings. They are also interesting because some of the graph products we will see preserve half-colorings, but not colorings.

Non-regular graphs can also be labeled in a natural way. For each vertex u, the edges adjacent to u are labeled with elements from a set of size $\deg(u)$.

5.6.2 Graph Squaring

We use the notation from Definition 5.22 to describe our products.

Definition 5.25. Let A be an $[n, d]$-graph with a labeling. The square A^2 of A is the $[n, d^2]$-graph with vertex set $[n]$ and a labeling in $[d] \times [d]$. For any $a \in [n]$ and $(i, j) \in [d] \times [d]$, we have

$$a[i, j] = a[i][j]. \tag{5.22}$$

More intuitively, this can be interpreted as taking all paths of length 2 in A. So A^2 has the same vertex set as A, and we put an edge in A^2 between two vertices a, u for each path of length 2 between a and u in A. In terms of random walks, taking one step in A^2 can be decomposed into taking 2 substeps in A. For each of these substeps we have d choices which gives us a total of d^2 choices (as expected since the degree of A^2 is d^2).

Notice that the labeling of A^2 (with $[d] \times [d]$ as its set of labels) is implicit to its construction from A. It can also be checked that graph squaring preserves half-colorings but not colorings.

Since A^2 has the same size as A but has more edges, we expect it to be a better expander. More precisely, the second eigenvalue gets squared:

Proposition 5.26. *If A has second eigenvalue λ_A then A^2 has second eigenvalue*

$$\lambda_{(A^2)} = \lambda_A^2.$$

Proof: If we let $\lambda_0 \geq \ldots \geq \lambda_{n-1}$ be the spectrum of M_A, then the spectrum of M_A^2 will be $\lambda_0^2, \ldots, \lambda_{n-1}^2$. By definition we have

$$\lambda_A = \max\left\{|\lambda_1|, \ldots, |\lambda_{n-1}|\right\}, \qquad \text{and} \qquad \lambda_{(A^2)} = \max\left\{\lambda_1^2, \ldots, \lambda_{n-1}^2\right\}. \tag{5.23}$$

Squaring positive numbers preserves their ordering, so as required:

$$\lambda_{(A^2)} = \lambda_A^2. \tag{5.24}$$

■

5.6.3 Graph Tensoring

Definition 5.27. Let A be an $[n, d_1]$-graph, and let B be an $[m, d_2]$-graph. Their tensor product $A \otimes B$ is an $[nm, d_1 d_2]$-graph with vertex set $[n] \times [m]$ and a labeling in $[d_1] \times [d_2]$. For any $(a, b) \in [n] \times [m]$ and $(i, j) \in [d_1] \times [d_2]$, we have

$$(a, b)[i, j] = (a[i], b[j]). \tag{5.25}$$

Throughout this work, we will interpret the vertex set $[n] \times [m]$ as n copies of $[m]$. We will also follow the convention of [61] and refer to these copies as *clouds*. So for a vertex $(a, b) \in [n] \times [m]$, a describes which cloud it belongs to, and b describes its position within cloud a.

In the context of random walks, we can interpret a step (i, j) in $A \otimes B$ from the vertex (a, b) as follows:

1. Take one step *between clouds*. The different possibilities are given by the edges of A. The position b within the cloud does not change:

   $$(a, b) \to (a[i], b).$$

   Notice that there are exactly d_1 possible choices for this step.

2. Take one step *within* the new cloud $a[i]$. We view the cloud as a copy of B, and take one step along an edge of this copy (so we stay in the same cloud):

$$(a[i], b) \to (a[i], b[j]).$$

Notice that there are exactly d_2 possible choices for this step.

So the total number of choices for both steps is $d_1 d_2$, which as expected is equal to the degree of $A \otimes B$. The labeling of $A \otimes B$ (with $[d_1] \times [d_2]$ as its set of labels) is implicit to its construction from A and B. The two subsets into which step (i, j) were decomposed above are commutative, we could just as well have presented them the other way round.

It can also be checked that graph tensoring preserves both colorings and half-colorings. Graph tensoring can also be interpreted as an operation on the corresponding transition matrices. We conveniently have

$$M_{A \otimes B} = M_A \otimes M_B. \tag{5.26}$$

The expansion of $A \otimes B$ will be the worse of the two expansions:

Proposition 5.28. *If A and B have second eigenvalues λ_A and λ_B then $A \otimes B$ has second eigenvalue*

$$\lambda_{A \otimes B} = \max\left\{\lambda_A, \lambda_B\right\}.$$

Proof: If we let $\lambda_0 \geq \ldots \geq \lambda_{n-1}$ and $\mu_0 \geq \ldots \geq \mu_{m-1}$ and be the eigenvalues of M_A and M_B, then

$$\left\{\lambda_i \cdot \mu_j \;\middle|\; i = 0, \ldots, n-1, \;\; j = 0, \ldots, m-1\right\} \tag{5.27}$$

is the set of eigenvalues of $M_{A \otimes B}$. Since $\lambda_0 = \mu_0 = 1$, the result follows. ■

5.6.4 The Zig-Zag Product

The *Zig-zag product*, introduced in 2002 by Reingold, Vadhan and Wigderson [61] enables the recursive construction of expander families. In all previous explicit constructions of expander families, although the graphs were easy to describe, the proofs of why they lead to good expanders were highly algebraic and rather complex. It was therefore difficult to conceptualize the connection between the algebra and the actual graphs, or to get any intuition as to why the resulting families were in fact expanders.

The zig-zag construction however is remarkable in that its analysis effectively relies on linear algebra, which makes it not only easier to follow but also somewhat more intuitive. Once the expansion properties of the product are known, it is very simple to show that the recursion suggested in [61] leads to an expander family.

Definition 5.29. Let A be an $[n, d_1]$-graph, and let B be a $[d_1, d_2]$-graph. Their zig-zag product $A ⓩ B$ is an $[nd_1, d_2^2]$-graph with vertex set $[n] \times [d_1]$ and a labeling in $[d_2] \times [d_2]$. For any $(a, b) \in [n] \times [d_1]$ and $(i, j) \in [d_2] \times [d_2]$ we have

$$(a, b)[i, j] = (a[b[i]], b[i][j]). \tag{5.28}$$

The zig-zag product has an appealing intuition in terms of walks. We can view the construction as first replacing each vertex of A by a copy of the vertices of B (which we call a *cloud*), leading to the vertex set $[n] \times [d_1]$. We can then break up one step (labeled (i, j)) in $A ⓩ B$ from vertex (a, b) into three substeps:

1. We take one step within the current cloud (d_2 choices):

$$(a, b) \to (a, b[i]).$$

2. We take one step between clouds (deterministic, only one choice).

$$(a, b[i]) \to (a[b[i]], b[i]).$$

3. We take one step within the new cloud (d_2 choices)

$$(a[b[i]], b[i]) \to (a[b[i]], b[i][j]).$$

Step 2 is determined by which vertex we are on within the current cloud. Indeed, this vertex corresponds to an edge label in A, and therefore uniquely defines a neighbor of the current cloud. This leads to a total of d_2^2 choices, which as expected is equal to the degree of $A ⓩ B$.

The expansion of $A ⓩ B$ can be bounded as follows:

Theorem 5.30. *If A is an $[n, d_1, \lambda_A]$-graph, and B is an $[d_1, d_2, \lambda_B]$-graph, then $A ⓩ B$ is a $[n \cdot d_1, d_2, f(\lambda_A, \lambda_B)]$-graph, where*

$$f(\lambda_A, \lambda_B) = \frac{1}{2} \cdot (1 - \lambda_B^2) \cdot \lambda_A + \frac{1}{2} \cdot \sqrt{(1 - \lambda_B^2)^2 \lambda_A^2 + 4\lambda_B^2}. \tag{5.29}$$

Furthermore, if $\lambda_A, \lambda_B < 1$ then $f(\lambda_A, \lambda_B) < 1$.

Proof: See [61]. ■

Although the bound (5.29) is rather complicated, it can be shown that

$$f(\lambda_A, \lambda_B) \leq \lambda_A + \lambda_B + \lambda_B^2. \tag{5.30}$$

As explained earlier, this product leads to recursive construction of fixed degree expander families.

Let B be a fixed $[\ell^8, \ell, \lambda]$-graph for some parameters ℓ and λ. We define the family $\{A_i\}_{i \in \mathbb{N}^*}$ as follows:

$$\begin{aligned} A_1 &= B^2 \\ A_2 &= B \otimes B \\ \forall i > 2: \quad A_i &= \left(A_{\lceil \frac{i-1}{2} \rceil} \otimes A_{\lfloor \frac{i-1}{2} \rfloor} \right)^2 ⓩ B. \end{aligned}$$

It can be checked that A_i is a $[\ell^{8i}, \ell^2, \mu_i]$-graph, in which $\mu_i = \lambda + O(\lambda^2)$. So by picking λ small enough to start with, we can ensure that there is $\mu < 1$ for which $\mu_i \leq \mu$ for all i, and therefore that $\{A_i\}_{i \in \mathbb{N}^*}$ is an expander family.

We see the usual trade-off between degree and expansion in the choice of the initial graph B. Getting A_i to have a small second eigenvalue requires λ to be small, which in turn means that the degree ℓ of B must be large, and this means that the degree ℓ^2 of our family will also be larger.

This method does not enable the construction of a family of d-regular Ramanujan graphs (this would require $\mu = O(d^{-1/2})$). With the normal zig-zag product the best we can hope for is a second eigenvalue of $O(d^{-1/4})$, but through the *derandomized zig-zag product* (also in [61]) one can obtain a family of d-regular graph with $\mu = O(d^{-1/3})$.

5.6.5 Derandomized Squaring

The derandomized squaring operation was first presented by Rozenman and Vadhan [64]. Recall that squaring an $[n, d, \lambda_A]$-graph A consisted in taking all paths of length 2. This led to improved expansion properties ($\lambda_A \to \lambda_A^2$), but increased the degree considerably ($d \to d^2$). The idea of *derandomized squaring* is to take only a subset of the paths of length 2. By cleverly choosing which of these paths to include we can get a considerably smaller degree than A^2, at the cost of only slightly worse expansion.

The idea is to use another graph C with parameters $[d, t, \lambda_C]$. Let (u, v, w) be a path of length 2 in A. Let $i, k \in [d]$ be the labels of the edges from u to v and from v to w respectively. Then in $A \circledS C$ we keep only those paths of length 2 for which i and k are connected in C. The expansion properties of the resulting graph will depend on the expansion properties of both A and C. Formally:

Definition 5.31. Let A be an $[n, d]$-graph, and let C be an $[d, t]$-graph. The derandomized square $A \circledS C$ of A with respect to C is the $[n, dt]$-graph with vertex set $[n]$ and a labeling in $[d_1] \times [t]$. For any $a \in [n]$ and $(i, j) \in [d] \times [t]$ we have

$$a[i, j] = a[i][i[j]]. \tag{5.31}$$

Notice that if K_d is the complete graph on d vertices then we obtain the standard squaring operation:

$$A \circledS K_d = A^2. \tag{5.32}$$

Derandomized squaring does not preserve half-colorings or colorings. The expansion properties of $A \circledS C$ will be analyzed in Section 6.3. The term "derandomized" comes from the fact that performing a random walk on $A \circledS C$ requires fewer random bits than a random walk on A^2. Indeed since the degree is smaller, there are fewer choices to be made at each step.

5.6.6 Projection

The concept of graph projection will be essential to our proofs in the next chapter. Whenever we have a graph whose vertices are divided into clouds, we can "collapse" each cloud into a single vertex, while keeping all the edges.

Definition 5.32. Suppose we have an $[nm, d]$-graph A with vertex set $[n] \times [m]$. $\mathrm{P}_n[A]$ is an $[n, md]$-graph, with vertex set $[n]$ and a labeling in $[m] \times [d]$, in which for any $a \in [n], (b, k) \in [m] \times [d]$:

$$a[b, k] = u, \tag{5.33}$$

where $(u, v) \in [n] \times [m]$ is the unique vertex of A for which $(a, b)[k] = (u, v)$.

Notice that $\mathrm{P}_n[A]$ has the same number of edges as A. We can interpret the $nm \times nm$ transition matrix M_A of an $[nm, d]$-graph A as a block matrix, consisting of $n \times n$ blocks, each of size $m \times m$. For $i, j \in [n]$ and $k, \ell \in [m]$ we use the notation

$$(M_A)_{ik,j\ell} \tag{5.34}$$

to refer to entry (k, ℓ) of block (i, j). The $n \times n$ transition matrix of $\mathrm{P}[A]$ is then equal to

$$\left(M_{\mathrm{P}_n[A]}\right)_{ij} = \frac{1}{m} \sum_{k=1}^{m} \sum_{\ell=1}^{m} (M_A)_{ik,j\ell}. \tag{5.35}$$

So each block is replaced with a single entry whose value is equal to the sum of all the entries in the block divided by m. The factor $\frac{1}{m}$ ensures that $M_{\mathrm{P}_n[A]}$ is stochastic.

Example 5.33. Consider the following $[18, 2]$-graph A,

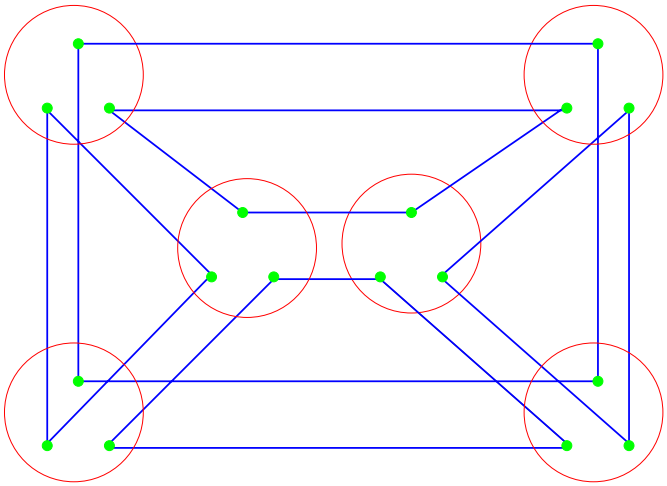


Figure 5.1: The graph A.

A has 18 vertices divided into 6 clouds. The projection $\mathrm{P}[A]$ $(= P_6[A])$ of A is then the following graph:

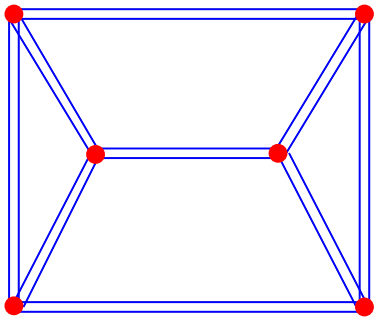


Figure 5.2: The graph $\mathrm{P}[A]$.

So $\mathrm{P}[A]$ has 6 vertices, and the same number of edges as A (namely 18).

5.6.7 De-Projection

While projection collapsed each cloud into a single vertex, *de-projection* is a sort of reverse operation that expands each vertex into a cloud.

Definition 5.34. Let A be a d-regular graph with vertex set $[n]$ and a labeling in $[d]$. We define the de-projection $\mathrm{DP}[A]$ of A as the $[nd, 1]$-graph with vertex set $[n] \times [d]$, in which each vertex (i, k) has a unique neighbor:

$$(i, k)[1] = (i[k], \ell), \tag{5.36}$$

where ℓ is the unique element of $[d]$ for which $i[k][\ell] = i$.

Notice that the graph DP$[A]$ is a matching (i.e. no two edges are adjacent). If the labeling of A is a half-coloring, then for each $i \in [n]$ and each $k \in [d]$, the element ℓ from (5.36) is equal to the partner color $\rho(k)$ of k, so that

$$\forall (i,k) \in [n] \times [d] : \quad (i,k)[1] = (i[k], \rho(k)). \tag{5.37}$$

Likewise if the labeling is an edge coloring then $\ell = k$ so that

$$\forall (i,k) \in [n] \times [d] : \quad (i,k)[1] = (i[k], k). \tag{5.38}$$

Notice that DP$[A]$ has the same number of edges as A, and that there are no edges within the clouds, only between clouds. Also, because it has degree 1, one step along DP$[A]$ induces a permutation of the vertices (in fact an involution).

We can write the transition matrix of DP$[A]$ as follows:

$$\big(M_{\mathrm{DP}[A]}\big)_{ik,jl} = \begin{cases} 1 & \text{if } \ j = i[k] \ \text{ and } \ i = j[\ell] \\ 0 & \text{otherwise.} \end{cases} \tag{5.39}$$

The de-projection operation is defined for any $[n,d]$-graph, whereas the projection $\mathrm{P}_n[A]$ is defined only for graphs whose vertices have been divided into n clouds. The relationship between the two operations can be described as follows: For any $[n,d]$-graph A, we have

$$\mathrm{P}_n\big[\mathrm{DP}[A]\big] = A. \tag{5.40}$$

Example 5.35. To illustrate the de-projection operation, we consider the $[6,3]$-graph A given below. For simplicity, the labeling in this example is an edge coloring.

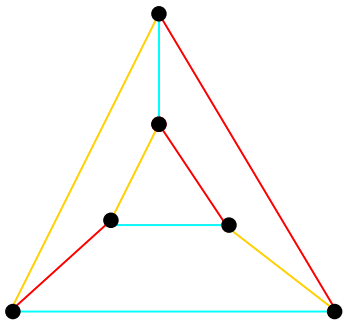


Figure 5.3: The graph A.

So the edges of A are assigned one of three possible colors. Its de-projection DP$[A]$ can then be drawn as:

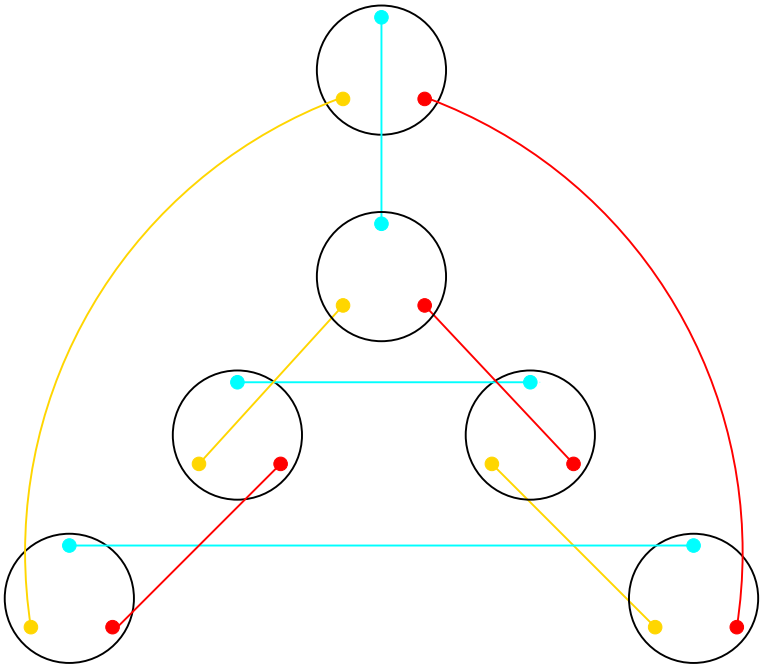


Figure 5.4: The graph DP$[A]$.

We see that the vertices of A are replaced by clouds. Each cloud has one vertex for each color it used in A. We also see that DP$[A]$ has degree 1, and there are edges only between clouds, not within the clouds.

5.7 The Spectrum of Biregular Bipartite Graphs

Although the expansion properties of biregular bipartite graphs have been widely used (expander codes, for example, are based entirely on these), there appears to be very little mention of the spectrum of such graphs, and how it can be related to their expansion properties. Nevertheless, there is a similar link to that found in non-bipartite regular graphs. For lack of reference, these links are derived in this section, along with the results needed in the next chapter.

We had previously defined the *second eigenvalue* only for non-bipartite graphs. In this section we extend this definition to cover biregular bipartite graphs, which will be used for our proofs in the next chapter. Our aim is then first to prove the results that will be needed, and also to establish the relationship between the second eigenvalue and combinatorial expansion in the biregular bipartite case. We will show that the second eigenvalue also governs the rate of convergence of random walks (Proposition 5.45), and that a modified version of the Expander Mixing Lemma holds (Lemma 5.48).

Throughout this section we will suppose that C is a biregular bipartite graph with d_1 left vertices and d_2 right vertices and of left and right degrees ℓ and r.

5.7.1 Notation

Recall that for any $n \in \mathbb{N}$, $[n]$ was defined as $\{1, \ldots, n\}$, and $1_n \in \mathbb{R}^n$ was the all one vector. We will also have the following:

Definition 5.36. If d_1 and d_2 are the numbers of left and right vertices of C, then

$$\begin{array}{rcl} D_1 & = & \{1, \ldots, d_1\}. \\ D_2 & = & \{d_1 + 1, \ldots, d\}. \\ [d] & = & \{1, \ldots, d\} = D_1 \sqcup D_2. \end{array} \tag{5.41}$$

When we consider vectors in $\mathbb{R}^d$ they will often have non-zero entries only in those positions in D_1 or only in D_2, and will therefore work with the following spaces:

Definition 5.37. We define the subspaces R_1 and R_2 of $\mathbb{R}^d$ as follows:

$$\begin{array}{rcl} R_1 & = & \{v \in \mathbb{R}^d \mid v_i = 0 \; \forall i \in D_2\}. \\ R_2 & = & \{v \in \mathbb{R}^d \mid v_i = 0 \; \forall i \in D_1\}. \end{array}$$

We therefore have $\mathbb{R}^d = R_1 \oplus R_2$.

Definition 5.38. e_1 and e_2 denote the following vectors in $\mathbb{R}^d$:

$$(e_1)_i = \begin{cases} 1 & \text{if } i \in D_1 \\ 0 & \text{if } i \in D_2. \end{cases} \tag{5.42}$$

$$(e_2)_i = \begin{cases} 0 & \text{if } i \in D_1 \\ 1 & \text{if } i \in D_2. \end{cases} \tag{5.43}$$

Notice that $e_1 + e_2 = 1_d$, $\langle e_1, e_1 \rangle = d_1$, $\langle e_2, e_2 \rangle = d_2$, and $\langle e_1, e_2 \rangle = 0$.

Recall that in general $1_n^{\parallel}$ and $1_n^{\perp}$ denote the spaces of vectors respectively parallel and perpendicular to 1_n. We have corresponding definitions for e_1 and e_2, where the spaces will be embedded into $\mathbb{R}^d$.

Definition 5.39. For $i = 1, 2$, we have:

$$\begin{array}{l} e_i^{\parallel} = \{\beta \cdot e_i \mid \beta \in \mathbb{R}\}. \\ e_i^{\perp} = \{v \in R_i \mid \langle v, e_i \rangle = 0\}. \end{array} \tag{5.44}$$

5.7.2 Transition Matrix

Let C be a biregular bipartite graph, with left and right vertex sets D_1, D_2, and of left and right degrees ℓ and r respectively. The number of edges of C can be expressed in two different ways:

$$|E(C)| = d_1 \ell = d_2 r, \tag{5.45}$$

which leads to the following equality

$$\frac{d_1}{d_2} = \frac{r}{\ell}. \tag{5.46}$$

The adjacency matrix of C has the form

$$\text{Adj}(C) = \left(\begin{array}{c|c} 0 & X \\ \hline X^{\dagger} & 0 \end{array} \right), \tag{5.47}$$

where X is a $d_1 \times d_2$ matrix. The rows of X have weight ℓ, while its columns have weight r. $\mathrm{Adj}(C)$ is symmetric and therefore has d real eigenvalues and an orthonormal set of eigenvectors. The first problem we encounter is how to define the normalized adjacency matrix of C. Indeed since C is not regular there is no degree by which to divide $\mathrm{Adj}(C)$. Instead we define M_C so that it describes one step of a random walk on C. This requires it to be stochastic (each column must be a probability vector), which leads to the following definition:

Definition 5.40. Let C be a biregular bipartite graph as describe above, with $\mathrm{Adj}(C)$ as in (5.47). Then the normalized adjacency matrix (or *transition matrix)* M_C of C is defined as

$$M_C = \left(\begin{array}{c|c} 0 & \frac{1}{r} \cdot X \\ \hline \frac{1}{\ell} \cdot X^\top & 0 \end{array}\right). \tag{5.48}$$

Because M_C is stochastic, its eigenvalues are all between -1 and 1. However M_C is not symmetric, and therefore many of the properties we showed in the previous section for regular graphs no longer hold (for example its eigenvectors are not necessarily pairwise orthogonal).

We start by presenting some characteristics of the spectrum and eigenvectors of $\mathrm{Adj}(C)$, which will then relate to those of M_C. Throughout this section, all vectors of the form

$$\left(\frac{x}{y}\right) \tag{5.49}$$

will be elements of $\mathbb{R}^d$, in which $x \in \mathbb{R}^{d_1}$ represents the top d_1 components and $y \in \mathbb{R}^{d_2}$ the bottom d_2 components. The next proposition states that the eigenvectors of $\mathrm{Adj}(C)$ with non-zero eigenvalues come in pairs.

Proposition 5.41. *If*

$$\left(\frac{x}{y}\right) \tag{5.50}$$

is an eigenvector of $\mathrm{Adj}(C)$ *with eigenvalue* λ*, then*

$$\left(\frac{x}{-y}\right) \tag{5.51}$$

is also an eigenvector of $\mathrm{Adj}(C)$ *with eigenvalue* $-\lambda$*.*

Proof: See appendix C. ■

Next, we relate the spectrum of $\mathrm{Adj}(C)$ to that of M_C. We can also deduce a bijection between the sets of eigenvectors of the two matrices.

Proposition 5.42. *Let* $u, v \in \mathbb{R}^d$ *be vectors written as follows:*

$$u = \left(\frac{x}{y}\right), \qquad v = \left(\frac{x}{\sqrt{\ell/r} \cdot y}\right). \tag{5.52}$$

u *is an eigenvector of* $\mathrm{Adj}(C)$ *with eigenvalue* λ *if and only if* v *is an eigenvector of* M_C *with eigenvalue* $\frac{\lambda}{\sqrt{\ell r}}$*.*

Proof: See appendix C. ■

5.7.3 Eigenvalues and Eigenvectors

The first consequence of Proposition 5.42 is that M_C also has d real eigenvalues. We let $\lambda_0 \geq \ldots \geq \lambda_{d-1}$ be these eigenvalues, and $v_0, \ldots, v_{d-1}$ be the corresponding normalized eigenvectors. It can be checked that

$$e_1 + \frac{r}{\ell} e_2 \qquad \text{and} \qquad e_1 - \frac{r}{\ell} e_2 \tag{5.53}$$

are eigenvectors of M_C, with respective eigenvalues 1 and -1. Since $-1 \leq \lambda_i \leq 1$ for all i, we deduce that $\lambda_0 = 1$ and $\lambda_{d-1} = -1$. Normalizing the vectors in (5.53) gives us

$$\lambda_0 = 1, \qquad v_0 = \frac{\sqrt{d_2/d_1} \cdot e_1 + \sqrt{d_1/d_2} \cdot e_2}{\sqrt{d}} \tag{5.54}$$

and

$$\lambda_{d-1} = -1, \qquad v_{d-1} = \frac{\sqrt{d_2/d_1} \cdot e_1 - \sqrt{d_1/d_2} \cdot e_2}{\sqrt{d}}. \tag{5.55}$$

Next, if we call $u_0, \ldots, u_{d-1}$ the normalized eigenvectors of $\mathrm{Adj}(C)$ (ordered in the usual way), we know first of all that they form an orthonormal basis of $\mathbb{R}^d$. Using Proposition 5.42, we can also deduce from (5.54) and (5.55) (and after normalizing) that

$$u_0 = \frac{e_1}{\sqrt{2d_1}} + \frac{e_2}{\sqrt{2d_2}}, \qquad u_{d-1} = \frac{e_1}{\sqrt{2d_1}} - \frac{e_2}{\sqrt{2d_2}}, \tag{5.56}$$

and their corresponding eigenvalues are $\sqrt{\ell r}$ and $-\sqrt{\ell r}$ respectively.

5.7.4 Random Walks

A random walk on a biregular bipartite graph does not converge to the uniform distribution. Indeed it is clear that if we start our walk on the left side, then after t steps we will be on the right side for odd t and back on the left side for even t.

For an initial distribution $x \in \mathbb{R}^d$, let

$$p_1 = \sum_{i \in D_1} x_i, \qquad \text{and} \qquad p_2 = \sum_{i \in D_2} x_i \tag{5.57}$$

denote the probabilities of starting on the left and right sides respectively (so $p_1 + p_2 = 1$). Then when t is even, the distribution $A^t x$ will converge to

$$w_{\text{even}} = p_1 \cdot \frac{e_1}{d_1} + p_2 \cdot \frac{e_2}{d_2}, \tag{5.58}$$

and to

$$w_{\text{odd}} = p_2 \cdot \frac{e_1}{d_1} + p_1 \cdot \frac{e_2}{d_2} \tag{5.59}$$

when t is odd. Intuitively this is saying that when t is even it is uniform over the left nodes with probability p_1 and uniform over the right nodes with probability p_2 (and vice versa when t is odd).

5.7.5 The Second Eigenvalue

Because $\lambda_{d-1} = -1$, if we used for biregular bipartite graphs Definition 5.8 of the second eigenvalue, then it would be 1 for every such graph. However we know that some bipartite graphs are better expanders than others, and would like to have a definition that reflects this.

For a non-bipartite graph A with eigenvalues $\lambda_0 \geq \ldots \geq \lambda_{n-1}$ and corresponding eigenvectors $v_0, \ldots, v_{n-1}$, the definition of the second eigenvalue of A as

$$\lambda_C = \max\left(|\lambda_1|, |\lambda_{n-1}|\right) \tag{5.60}$$

was partly motivated in terms of convergence of random walks. The stationary distribution of a random walk on A is a multiple of v_0. An initial distribution $x \in \mathbb{R}^n$ can be expressed in the basis given by the eigenvectors as

$$x = \sum_{i=0}^{n-1} \alpha_i v_i. \tag{5.61}$$

Under a random walk on A it will converge to its first component $\alpha_0 v_0$, and λ_C therefore describes the rate at which the other components get killed.

With a biregular bipartite graph C as above the situation is similar, though we must consider walks of even or odd length separately to get convergence. In both cases the distributions to which the walks converge are in Span(v_0, v_{d-1}), and so this time it is

$$\max\left(|\lambda_1|, |\lambda_{d-2}|\right) \tag{5.62}$$

that describes the rate at which the remaining components get killed. This leads to to the following definition:

Definition 5.43. Let C be a biregular bipartite graph with transition matrix M_C, and let $\lambda_0 \geq \ldots \geq \lambda_{d-1}$ be its eigenvalues. The *second eigenvalue* of C is defined as

$$\lambda_C = \max\left(|\lambda_1|, |\lambda_{d-2}|\right). \tag{5.63}$$

Our aim in the rest of this section is to give some properties of λ_C we will need, and also to see how it can be related to the expansion properties of C.

5.7.6 Results We Will Need

We saw in previous sections that in an $[n, d]$ non-bipartite graph A, for any $x \in 1_n^{\perp}$:

$$\|M_A \cdot x\| \leq \lambda_A \cdot \|x\|. \tag{5.64}$$

The following proposition presents the corresponding property of the second eigenvalue of a biregular bipartite graph.

Proposition 5.44. *Let C be a biregular bipartite graph, with transition matrix M_C and second eigenvalue λ_C.*

- *For any $x \in e_1^{\perp}$, we have*

$$\left\|M_C \cdot x\right\| \leq \sqrt{\frac{d_1}{d_2}} \cdot \lambda_C \cdot \|x\|. \tag{5.65}$$

- *For any $x \in e_2^{\perp}$, we have*

$$\big\|M_C \cdot x\big\| \leq \sqrt{\frac{d_2}{d_1}} \cdot \lambda_C \cdot \|x\|. \tag{5.66}$$

Proof: We will show only the first part (the second part follows by symmetry). We know from Proposition 5.42 that if $\lambda_0 \geq \ldots \geq \lambda_{d-1}$ are the eigenvalues of M_C then $\sqrt{\ell r}\cdot\lambda_0, \ldots, \sqrt{\ell r}\cdot\lambda_{d-1}$ are the eigenvalues of $\mathrm{Adj}(C)$. We also let $u_0, \ldots, u_{d-1}$ be the corresponding eigenvectors of $\mathrm{Adj}(C)$. These form an orthonormal basis of $\mathbb{R}^d$ (since $\mathrm{Adj}(C)$ is symmetric). Once again, we decompose x with respect to this basis:

$$x = \sum_{i=0}^{d-1} \alpha_i u_i. \tag{5.67}$$

Recall from (5.56) that

$$u_0 = \frac{e_1}{\sqrt{2d_1}} + \frac{e_2}{\sqrt{2d_2}}, \qquad\qquad u_{d-1} = \frac{e_1}{\sqrt{2d_1}} - \frac{e_2}{\sqrt{2d_2}}, \tag{5.68}$$

Now because $x \in e_1^{\perp}$ we have

$$\alpha_0 = \langle x, u_0\rangle = 0, \qquad\qquad \alpha_{d-1} = \langle x, u_{d-1}\rangle = 0. \tag{5.69}$$

Also recall that

$$\mathrm{Adj}(C) = \left(\begin{array}{c|c} 0 & X \\ \hline X^\top & 0 \end{array}\right), \qquad \text{and} \qquad M_C = \left(\begin{array}{c|c} 0 & \frac{1}{r}\cdot X \\ \hline \frac{1}{\ell}\cdot X^\top & 0 \end{array}\right). \tag{5.70}$$

The important thing to notice next is that because $x \in R_1$, we have

$$M_C \cdot x = \left(\begin{array}{c|c} 0 & \frac{1}{r}\cdot X \\ \hline \frac{1}{\ell}\cdot X^\top & 0 \end{array}\right) \cdot \left(\frac{x_1}{0}\right) = \left(\frac{\frac{1}{\ell}\cdot X^\top x_1}{0}\right) = \frac{1}{\ell}\cdot \mathrm{Adj}(C)\cdot x. \tag{5.71}$$

Since $\lambda_i\sqrt{\ell r}$ is the eigenvalue of $\mathrm{Adj}(C)$ corresponding to u_i (see Proposition 5.42) and $\alpha_0 = \alpha_{d-1} = 0$ (see (5.69)) this leads to

$$M_C \cdot x \;=\; \frac{1}{\ell}\cdot \mathrm{Adj}(C)\cdot \sum_{i=1}^{d-2}\alpha_i\cdot u_i \;=\; \frac{1}{\ell}\cdot\sum_{i=1}^{d-2}\alpha_i\cdot\sqrt{\ell r}\cdot\lambda_i\cdot u_i. \tag{5.72}$$

Therefore by the definition of λ_C we have

$$\big\|M_C\cdot x\big\|^2 \;=\; \big\|\frac{1}{\ell}\cdot\sum_{i=1}^{d-2}\alpha_i\cdot\sqrt{\ell r}\cdot\lambda_i\cdot u_i\big\|^2 \;\leq\; \frac{\ell r}{\ell^2}\cdot\lambda_C^2\cdot\overbrace{\big\|\sum_{i=1}^{d-2}\alpha_i\cdot u_i\big\|^2}^{\|x\|^2}. \tag{5.73}$$

And so recalling that $\frac{r}{\ell} = \frac{d_1}{d_2}$, this gives us

$$\big\|M_C\cdot x\big\| \leq \sqrt{\frac{r}{\ell}}\cdot\lambda_C\cdot\|x\| = \sqrt{\frac{d_1}{d_2}}\cdot\lambda_C\cdot\|x\|, \tag{5.74}$$

as required. ■

5.7.7 Convergence of Random Walks

First of all, we see from the definition of M_C (5.48) that

$$M_C^2 = \left(\begin{array}{c|c} 0 & \frac{1}{r}\cdot X \\ \hline \frac{1}{\ell}\cdot X^\top & 0 \end{array}\right)^2 = \left(\begin{array}{c|c} \frac{1}{\ell r}XX^\top & 0 \\ \hline 0 & \frac{1}{\ell r}X^\top X \end{array}\right). \tag{5.75}$$

C^2 is a regular graph of degree ℓr with two connected components, and M_C^2 is its transition matrix. Let $\mu_0 \geq \ldots \geq \mu_{d-1}$ be its eigenvalues with corresponding eigenvectors $w_0, \ldots, w_{d-1}$. The w_i's form an orthonormal basis of $\mathbb{R}^d$. It can easily be checked that

$$M_C^2 \cdot e_1 = e_1, \qquad M_C^2 \cdot e_2 = e_2. \tag{5.76}$$

Therefore normalizing these we see that

$$\mu_0 = 1, \qquad w_0 = \frac{e_1}{\sqrt{d_1}} \tag{5.77}$$

and

$$\mu_1 = 1, \qquad w_1 = \frac{e_2}{\sqrt{d_2}}. \tag{5.78}$$

In general, if λ is an eigenvalue of a matrix M then λ^2 is an eigenvalue of M^2. So here if $\lambda_0 \geq \ldots \geq \lambda_{d-1}$ are the eigenvalues of M_C then

$$\{\mu_0, \ldots, \mu_{d-1}\} = \{\lambda_0^2, \ldots, \lambda_{d-1}^2\}. \tag{5.79}$$

Since $\mu_0 = \lambda_0^2$ and $\mu_1 = \lambda_{d-1}^2$, the remaining μ_i's are in $\{\lambda_1^2, \ldots, \lambda_{d-2}^2\}$, and therefore by the definition of λ_C we have

$$\mu_2, \ldots, \mu_{d-1} \leq \lambda_C^2. \tag{5.80}$$

Now let $x \in \mathbb{R}^d$ be an initial distribution on the vertices of C. We stated above that a random walk of length t on C will converge to different distributions depending on whether t is even or odd. In this subsection we show this formally, and prove that the rate of convergence is given by the second eigenvalue λ_C of C.

Proposition 5.45. *Suppose we take an even number t of steps of a random walk on C from an initial distribution $x \in \mathbb{R}^d$. Then*

$$\|M_C^t x - w_{\text{even}}\| \leq \lambda_C^t, \tag{5.81}$$

where

$$w_{\text{even}} = p_1 \cdot \frac{e_1}{d_1} + p_2 \cdot \frac{e_2}{d_2}. \tag{5.82}$$

Proof: t is even, so we let $t = 2s$. The eigenvectors $w_0, \ldots, w_{d-1}$ of M_C^2 form an orthonormal basis of $\mathbb{R}^d$. We decompose x with respect to this basis as

$$x = \sum_{i=0}^{d-1} \alpha_i w_i. \tag{5.83}$$

Recall that p_1 and p_2 were defined in (5.54) as the probabilities of starting on the left and right vertices of C:

$$p_1 = \sum_{i \in D_1} x_i, \qquad \text{and} \qquad p_2 = \sum_{i \in D_2} x_i. \tag{5.84}$$

We have

$$\alpha_0 = \langle x, u_0 \rangle = \langle x, \frac{e_1}{\sqrt{d_1}} \rangle = \frac{p_1}{\sqrt{d_1}}, \tag{5.85}$$

and likewise

$$\alpha_1 = \frac{p_2}{\sqrt{d_2}}. \tag{5.86}$$

Now recalling that $t = 2s$ gives us

$$\begin{aligned} M_C^t x &= (M_C^2)^s \textstyle\sum_{i=0}^{d-1} \alpha_i w_i \\ &= \textstyle\sum_{i=0}^{d-1} \alpha_i \mu_i^s w_i \\ &= \alpha_0 w_0 + \alpha_1 w_1 + \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i. \end{aligned} \tag{5.87}$$

From the expression for α_0, α_1 (5.85), (5.86) and for w_0, w_1 (5.77), (5.78) we obtain

$$\begin{aligned} M_C^t x &= p_1 \tfrac{e_1}{d_1} + p_2 \tfrac{e_2}{d_2} + \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \\ &= w_{\text{even}} + \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \end{aligned} \tag{5.88}$$

Therefore

$$\begin{aligned} \|M_C^t x - w_{\text{even}}\| &= \big\| \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \big\| \\ &= \sqrt{\textstyle\sum_{i=2}^{d-1} \mu_i^{2s} \alpha_i^2} \\ &\leq \lambda_C^{2s} \sqrt{\textstyle\sum_{i=2}^{d-1} \alpha_i^2} \qquad \text{(using (5.80))} \\ &< \lambda_C^t \cdot \|x\| \\ &\leq \lambda_C^t. \end{aligned} \tag{5.89}$$

■

Proposition 5.46. *Suppose we take an odd number of steps $t = 2s+1$ of a random walk on C from an initial distribution x.*

1. *If $x \in R_1$ then*

$$\|M_C^t x - \frac{e_2}{d_2}\| \leq \sqrt{\frac{d_1}{d_2}} \lambda_C^t. \tag{5.90}$$

2. If $x \in R_2$ then

$$\|M_C^t x - \frac{e_1}{d_1}\| \leq \sqrt{\frac{d_2}{d_1}}\lambda_C^t. \tag{5.91}$$

Proof: We will show only the first part, the second part will then follow by symmetry. Recall that M_C^2 has two connected components, so it really is the concatenation of two separate subgraphs, one on the vertices in D_1 and the other on the vertices in D_2. This is reflected in its eigenvectors w_i which can be divided into two categories: d_1 of them in R_1 (for the first subgraph) and the remaining d_2 in R_2 (for the second subgraph). So if we express x in basis $w_0, \ldots, w_{d-1}$ as

$$x = \sum_{i=0}^{d-1} \alpha_i w_i, \tag{5.92}$$

then we will have $\alpha_i = 0$ for all $w_i \in R_2$.

Recall from (5.57) that p_1 and p_2 were defined as

$$p_1 = \sum_{i \in D_1} x_i, \qquad \text{and} \qquad p_2 = \sum_{i \in D_2} x_i. \tag{5.93}$$

Therefore in this case since $x \in R_1$ (and x is a distribution), we have

$$p_1 = 1, \qquad p_2 = 0. \tag{5.94}$$

It can be checked that

$$M_C \cdot \frac{e_1}{d_1} = \frac{e_2}{d_2}. \tag{5.95}$$

Now,

$$\begin{aligned}
M_C^t x &= M_C^{2s+1} \cdot \textstyle\sum_{i=0}^{d-1} \alpha_i w_i \\
&= M_C \cdot (M_C^2)^s \cdot \textstyle\sum_{i=0}^{d-1} \alpha_i w_i \\
&= M_C \cdot \left(p_1 \tfrac{e_1}{d_1} + p_2 \tfrac{e_2}{d_2} + \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \right) \quad \text{(from (5.87))} \\
&= M_C \cdot \left(\tfrac{e_1}{d_1} + \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \right) \quad \text{(from (5.94))} \\
&= \tfrac{e_2}{d_2} + M_C \cdot \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \quad \text{(from (5.95))}.
\end{aligned} \tag{5.96}$$

Because

$$\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \in e_1^{\perp}, \tag{5.97}$$

Proposition 5.44 tells us that

$$\big\| M_C \cdot \sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \big\| \leq \sqrt{\frac{d_1}{d_2}} \cdot \lambda_C \cdot \big\| \sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i \big\|. \tag{5.98}$$

This leads to

$$\begin{aligned}
\|M_C^t x - \tfrac{e_2}{d_2}\| &= \left\|M_C \cdot \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i\right\| \\
&\leq \sqrt{\tfrac{d_1}{d_2}} \cdot \lambda_C \cdot \left\| \textstyle\sum_{i=2}^{d-1} \mu_i^s \alpha_i w_i\right\| \quad \text{(from (5.98))} \\
&= \sqrt{\tfrac{d_1}{d_2}} \cdot \lambda_C \cdot \sqrt{\textstyle\sum_{i=2}^{d-1} \mu_i^{2s} \alpha_i^2} \\
&\leq \sqrt{\tfrac{d_1}{d_2}} \cdot \lambda_C \cdot \lambda_C^{2s} \cdot \sqrt{\textstyle\sum_{i=2}^{d-1} \alpha_i^2} \quad \text{(from (5.80))} \\
&< \sqrt{\tfrac{d_1}{d_2}} \cdot \lambda_C^{2s+1} \cdot \|x\| \\
&\leq \sqrt{\tfrac{d_1}{d_2}} \cdot \lambda_C^t.
\end{aligned} \tag{5.99}$$

■

5.7.8 The Expander Mixing Lemma

We start by stating the *expander mixing lemma,* (due to Alon and Chung [3]):

Theorem 5.47. ***The Expander Mixing Lemma***
Let A be a d-regular graph on vertex set $[n]$. Let $S, T \subseteq [n]$. Then

$$\left||E(S,T)| - \frac{d \cdot |S| \cdot |T|}{n}\right| \leq \lambda_A \cdot d \cdot \sqrt{|S| \cdot |T|}.$$

This is saying that the number of edges between S and T is close to its expected value in a random setting, namely $\frac{d \cdot |S| \cdot |T|}{2}$. The second eigenvalue λ_A of A determines how close. We show below an analogue of the expander mixing lemma for biregular bipartite graphs. The only difference is that one of the sets must contain only left nodes and the other one only right nodes.

Theorem 5.48. ***The Bipartite Expander Mixing Lemma***
Let C be a biregular bipartite graph, with left and right vertex sets D_1 and D_2 respectively, and left and right degrees ℓ and r. For any $S \subseteq D_1$ and $T \subseteq D_2$ we have

$$\left||E(S,T)| - \frac{\ell \cdot |S| \cdot |T|}{d_1}\right| \leq \lambda_C \cdot \sqrt{\ell r} \cdot \sqrt{|S| \cdot |T|}.$$

Proof: Let $\chi_S \in \mathbb{R}^d$ denote the characteristic vector of S:

$$(\chi_S)_i = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{otherwise,} \end{cases} \tag{5.100}$$

and likewise let χ_T denote the characteristic vector of T.

If $\lambda_0 \geq \ldots \geq \lambda_{d-1}$ are the eigenvalues of M_C, then Proposition 5.42 tells us that

$$\sqrt{\ell r} \cdot \lambda_0 \geq \ldots \geq \sqrt{\ell r} \cdot \lambda_{d-1} \tag{5.101}$$

are the eigenvalues of $\mathrm{Adj}(C)$. We then let $u_0, \ldots, u_{d-1}$ be the corresponding normalized eigenvectors of $\mathrm{Adj}(C)$, which form an orthonormal basis of $\mathbb{R}^d$. Once again, we express χ_S and χ_T in this basis:

$$\chi_S = \sum_{i=0}^{d-1} \alpha_i \cdot u_i, \qquad \chi_T = \sum_{i=0}^{d-1} \beta_i \cdot u_i.$$

A little maniplulation shows that

$$\big|E(S,T)\big| = \chi_S^\top \cdot \mathrm{Adj}(C) \cdot \chi_T, \tag{5.102}$$

which implies

$$\begin{aligned}
\big|E(S,T)\big| &= \chi_S^\top \cdot \mathrm{Adj}(C) \cdot \chi_T \\
&= \big(\textstyle\sum_{i=0}^{d-1} \alpha_i u_i^\top\big) \cdot \mathrm{Adj}(C) \cdot \big(\textstyle\sum_{i=0}^{d-1} \beta_i u_i\big) \\
&= \big(\textstyle\sum_{i=0}^{d-1} \alpha_i u_i^\top\big) \cdot \big(\textstyle\sum_{i=0}^{d-1} \beta_i \sqrt{\ell r} \lambda_i u_i\big) \\
&= \sqrt{\ell r} \cdot \textstyle\sum_{i=0}^{d-1} \alpha_i \beta_i \lambda_i \qquad \text{(since the } u_i\text{'s are orthonormal)} \\
&= \sqrt{\ell r} \cdot \big(\alpha_0\beta_0\lambda_0 + \alpha_{d-1}\beta_{d-1}\lambda_{d-1} + \textstyle\sum_{i=1}^{d-2} \alpha_i\beta_i\lambda_i\big) \\
&= \sqrt{\ell r} \cdot \big(\alpha_0\beta_0 - \alpha_{d-1}\beta_{d-1} + \textstyle\sum_{i=1}^{d-2} \alpha_i\beta_i\lambda_i\big),
\end{aligned} \tag{5.103}$$

where the last equality follows from the fact that $\lambda_0 = 1$ and $\lambda_{d-1} = -1$. Recall from (5.56) that

$$u_0 = \frac{e_1}{\sqrt{2d_1}} + \frac{e_2}{\sqrt{2d_2}}, \qquad u_{d-1} = \frac{e_1}{\sqrt{2d_1}} - \frac{e_2}{\sqrt{2d_2}}. \tag{5.104}$$

This means that

$$\alpha_0 = \langle \chi_S, u_0 \rangle = \frac{|S|}{\sqrt{2d_1}}, \qquad \text{and} \qquad \alpha_{d-1} = \langle \chi_S, u_{d-1} \rangle = \frac{|S|}{\sqrt{2d_1}}. \tag{5.105}$$

Likewise we obtain

$$\beta_0 = \frac{|T|}{\sqrt{2d_2}}, \qquad \text{and} \qquad \beta_{d-1} = -\frac{|T|}{\sqrt{2d_2}}. \tag{5.106}$$

Combining (5.103), (5.105) and (5.106) leads to

$$\begin{aligned}
\big|E(S,T)\big| &= \sqrt{\ell r} \cdot \left(\tfrac{|S||T|}{\sqrt{4d_1d_2}} - \tfrac{-|S||T|}{\sqrt{4d_1d_2}} + \textstyle\sum_{i=1}^{d-2} \alpha_i\beta_i\lambda_i\right) \\
&= \sqrt{\ell r} \cdot \tfrac{2|S||T|}{2\sqrt{d_1d_2}} + \sqrt{\ell r} \cdot \textstyle\sum_{i=1}^{d-2} \alpha_i\beta_i\lambda_i \\
&= \tfrac{\ell|S||T|}{d_1} + \sqrt{\ell r} \cdot \textstyle\sum_{i=1}^{d-2} \alpha_i\beta_i\lambda_i \qquad \text{(since } \tfrac{\ell}{d_1} = \tfrac{r}{d_2}\text{)}.
\end{aligned} \tag{5.107}$$

Finally,

$$\begin{aligned}
\left||E(S,T)| - \tfrac{\ell|S||T|}{d_1}\right| &= \left|\sqrt{\ell r}\cdot\textstyle\sum_{i=1}^{d-2}\alpha_i\beta_i\lambda_i\right| \\
&\leq \sqrt{\ell r}\cdot\textstyle\sum_{i=1}^{d-2}|\alpha_i|\cdot|\beta_i|\cdot|\lambda_i| \\
&\leq \sqrt{\ell r}\cdot\lambda_C\cdot\textstyle\sum_{i=1}^{d-2}|\alpha_i|\cdot|\beta_i| \quad \text{(by the definition of } \lambda_C\text{)} \\
&\leq \sqrt{\ell r}\cdot\lambda_C\cdot\textstyle\sum_{i=0}^{d-1}|\alpha_i|\cdot|\beta_i|.
\end{aligned}$$

Now define the vectors $\alpha', \beta' \in \mathbb{R}^d$ as $\alpha'_i = |\alpha_i|$ and likewise for β'. We obtain

$$\begin{aligned}
\left||E(S,T)| - \tfrac{\ell|S||T|}{d_1}\right| &\leq \sqrt{\ell r}\cdot\lambda_C\cdot\langle\alpha',\beta'\rangle \\
&\leq \sqrt{\ell r}\cdot\lambda_C\cdot\|\alpha'\|\cdot\|\beta'\| \quad \text{(by the Cauchy-Schwartz inequality)} \\
&= \sqrt{\ell r}\cdot\lambda_C\cdot\|\alpha\|\cdot\|\beta\| \\
&= \sqrt{\ell r}\cdot\lambda_C\cdot\|\chi_S\|\cdot\|\chi_T\| \\
&= \sqrt{\ell r}\cdot\lambda_C\cdot\sqrt{|S|\cdot|T|},
\end{aligned}$$

as required. ■

Chapter 6

Derandomization Through Expander Graphs

6.1 Introduction

The derandomized square introduced by Rozenman and Vadhan in [64] enabled the derandomization of a standard graph product, leading to graphs of smaller degree at the cost of slightly worsening the expansion properties. The derandomized square of a graph A is taken with respect to another graph C, and the authors obtained in [64] a bound on its spectral expansion as a function of the second eigenvalues of A and C, which they then improved in [65]. They also used this product to obtain an alternative proof that S-T connectivity in undirected graphs can be solved in deterministic logspace.

In this chapter we introduce derandomized versions of another standard graph product (tensoring), and of a code product (concatenation). These are based on the ideas presented in [64], and are also taken with respect to another graph on whose expansion their properties will depend. We will first derive the improved bound on the expansion of the derandomized square from [65] using a different method. We can then use these techniques to analyze and bound the expansion of the derandomized tensor product. This will require some of the tools introduced in the previous chapter.

The derandomization technique essentially involves taking a graph and removing certain edges. Which edges are removed is determined by another graph. In derandomized code concatentaion, we apply an analogous technique to the world of codes, whereby a code is punctured with a pattern given by an expander graph. This is interesting in the sense that constructing good codes can essentially be reduced to finding good puncturing patterns, indeed almost any code can be seen as a puncturing of the dual of a Hamming code. Likewise, an AG-code is really a puncturing of a product of two or more Reed-Solomon codes.

We start with some standard definitions and results that will constitute the background for the subsequent proofs. We then obtain in Section 6.3 the bound on the spectral expansion of the derandomized square. The derivation of our bound on the second eigenvalue of the derandomized tensor product is rather technical. We give only an outline in Section 6.4 and include the full proof in Appendix B. The analysis is effectively an extension of that in Section 6.3, though considerably longer. Finally in Section 6.5 we introduce and study derandomized code concatenation.

6.2 Background

We will be interacting often with vector spaces, tensor products and inner products. We therefore start by giving some definitions and standard results.

Definition 6.1.
• If $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$ then $u \otimes v$ is a vector in $\mathbb{R}^{nm}$ (i.e., we identify $R^n \otimes \mathbb{R}^m$ with $\mathbb{R}^{nm}$). If we index its entries with the set $[n] \times [m]$ then we have

$$(u \otimes v)_{ij} = u_i \cdot v_j.$$

• If $G \in \mathbb{R}^{n \times n}$ and $H \in \mathbb{R}^{m \times m}$ then $G \otimes H$ is a matrix in $\mathbb{R}^{nm \times nm}$. If we index its rows and columns with the set $[n] \times [m]$ then we have

$$(G \otimes H)_{ik,j\ell} = G_{ij} \cdot H_{k\ell}.$$

• If U and V are subspaces of $\mathbb{R}^n$, then $U \otimes V$ is the vector space defined as

$$U \otimes V = \text{Span}\big\{u \otimes v \mid u \in U,\ \ v \in V\big\}.$$

The basic properties of tensor and inner products we will use are given below:

Proposition 6.2.
• *If $G \in \mathbb{R}^{n \times n}$, $H \in \mathbb{R}^{m \times m}$, $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$ then*

$$(G \otimes H) \cdot (u \otimes v) = (Gu) \otimes (Hv).$$

• *Tensoring is distributive over vector addition: If $u_1, u_2 \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$ then*

$$\begin{array}{rcl} (u_1 + u_2) \otimes v & = & u_1 \otimes v \ + \ u_1 \otimes v \\ v \otimes (u_1 + u_2) & = & v \otimes u_1 \ + \ v \otimes u_2. \end{array}$$

• *If $\{u_1, \ldots, u_n\}$ is a basis of U and $\{v_1, \ldots, v_m\}$ is a basis of V, then*

$$\big\{u_i \otimes v_j \ \mid \ i \in [n],\ j \in [m]\big\}$$

is a basis of $U \otimes V$. As a consequence we have

$$\dim(U \otimes V) = \dim(U) \cdot \dim(V).$$

• *If $u_1, \ldots, u_n$ form a basis of $\mathbb{R}^n$ then for any $x \in \mathbb{R}^n$ we have*

$$x = \alpha_1 u_1 + \ldots + \alpha_n u_n, \tag{6.1}$$

where for all $i \in [n]$:

$$\alpha_i = \frac{\langle x, u_i \rangle}{\langle u_i, u_i \rangle}. \tag{6.2}$$

• *If $u_1, u_2 \in \mathbb{R}^n$ and $v_1, v_2 \in \mathbb{R}^m$ then*

$$\langle u_1 \otimes v_1, u_2 \otimes v_2 \rangle = \langle u_1, u_2 \rangle \cdot \langle v_1, v_2 \rangle.$$

In particular if either $u_1 \perp u_2$ or $v_1 \perp v_2$ then $(u_1 \otimes v_1) \perp (u_2 \otimes v_2)$.

• *If $u_1, \ldots, u_k \in \mathbb{R}^n$ are pairwise orthogonal then*

$$\|u_1 + \ldots + u_k\|^2 = \|u_1\|^2 + \ldots + \|u_k\|^2.$$

• *Suppose $u_1, \ldots, u_n \in \mathbb{R}^n$ form an orthonormal basis of $\mathbb{R}^n$, and an element $x \in \mathbb{R}^n$ can be expressed as*

$$x = \alpha_1 u_1 + \ldots + \alpha_n u_n, \tag{6.3}$$

where $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$. Then we have

$$\|x\|^2 = \alpha_1^2 + \ldots + \alpha_n^2. \tag{6.4}$$

Proof: These are all standard results. ■

6.3 Derandomized Squaring

6.3.1 Introduction

We described the derandomized squaring operation in Subsection 5.6.5. In their original conference paper [64], Rozenman and Vadhan obtained an upper bound on the second eigenvalue of the derandomized square by interpreting it as a projection of the zig-zag product, and using the bound from [61] (Theorem 5.30).

With a more careful analysis, a tighter bound can be found, as was done by the same authors in [65], and independently in [16] using a different method. The latter is the derivation we present in this section.

Recall that for an $[n, d, \lambda_A]$-graph A and a $[d, t, \lambda_C]$-graph C, the derandomized square $A Ⓢ C$ is defined (using the notation from Definition 5.22) as the $[n, dt]$-graph with

$$a[i, j] = a[i][i[j]].$$

For the rest of this section we suppose that we have two graphs A and C with the parameters above. Our aim is to find an upper bound on $\lambda_{A Ⓢ C}$ as a function of λ_A and λ_C. Throughout this chapter, we will often abuse notation and write G to denote both a graph and its transition matrix M_G. Recall from Theorem 5.10 that in general for a graph G on vertex set $[n]$ we have

$$\lambda_G = \max_{x \in 1_n^\perp} \frac{\left|\langle Gx, x\rangle\right|}{\langle x, x\rangle}, \tag{6.5}$$

so we need to look at the effect of the transition matrix (or equivalently, at the effect of one step of a random walk) on the anti-uniform vectors.

6.3.2 $A Ⓢ C$ as a Projection

We will consider $A Ⓢ C$ as a projection of a larger graph. The key point about projections is that analyzing $\mathrm{P}[G]$ over anti-uniform vectors is the same as considering G itself over vectors that are anti-uniform overall, but uniform over each cloud:

Proposition 6.3. *Let G be a graph with vertex set $[n] \times [d]$ (whose vertices are grouped into n clouds of size d). Then*

$$\max_{x \in 1_n^\perp} \frac{\left|\langle \mathrm{P}_n[G] \cdot x, x\rangle\right|}{\langle x, x\rangle} = \max_{x \in 1_n^\perp \otimes 1_d^\parallel} \frac{\left|\langle Gx, x\rangle\right|}{\langle x, x\rangle}. \tag{6.6}$$

From the perspective of random walks, the intuition behind Proposition 6.3 is that taking a step in $\mathrm{P}_n[G]$ from vertex i involves choosing an edge among all those connected to cloud i in G. This choice can be broken up into first picking a vertex uniformly from all the vertices in cloud i, and then choosing an edge from this vertex. So it is equivalent to taking a step in G starting from a uniformly chosen vertex of cloud i. We prove this formally below:

Proof: Let $P = \mathrm{P}_n[G]$. Recall that we can view the $nd \times nd$ matrix G as a block matrix consisting of $n \times n$ blocks, each of size $d \times d$. We use the following indexing: For $i, j \in [n]$ and $k, \ell \in [d]$,

$$G_{ik,j\ell} \tag{6.7}$$

denotes entry (k, ℓ) of block (i, j). P is then the $n \times n$ matrix defined as defined as

$$P_{ij} = \frac{1}{d} \sum_{k=1}^{d} \sum_{\ell=1}^{d} G_{ik,j\ell}. \tag{6.8}$$

There is a natural bijection $\pi : 1_n^\perp \to \left(1_n^\perp \otimes 1_d^\parallel\right)$ defined as

$$u \mapsto u \otimes 1_d. \tag{6.9}$$

Note that π is clearly linear and injective, so since both spaces have dimension $n-1$ it must be a bijection.

We will show that for any $u \in 1_n^\perp$, if we let $w = \pi(u) = u \otimes 1_d$ then

$$\frac{\langle Pu, u\rangle}{\langle u, u\rangle} = \frac{\langle Gw, w\rangle}{\langle w, w\rangle}, \tag{6.10}$$

from which the required result (6.6) follows immediately.

$Gw = G(u \otimes 1_d)$ is a vector in $\mathbb{R}^{nd}$. Indexing its entries with the set $[n] \times [d]$ gives us

$$\big(Gw\big)_{ik} = \Big(G(u \otimes 1_d)\Big)_{ik} = \sum_{j=1}^{n}\sum_{\ell=1}^{d} G_{ik,j\ell} \cdot \big(u \otimes 1_d\big)_{j\ell} = \sum_{j=1}^{n}\sum_{\ell=1}^{d} G_{ik,j\ell} \cdot u_j. \tag{6.11}$$

So on the one hand we have:

$$\begin{aligned}
\big\langle Gw, w\big\rangle &= \big\langle G(u \otimes 1_d),\ \ u \otimes 1_d\big\rangle \\
&= \textstyle\sum_{i=1}^{n}\sum_{k=1}^{d} \big(G(u \otimes 1_d)\big)_{ik} \cdot \big(u \otimes 1_d\big)_{ik} \\
&= \textstyle\sum_{i=1}^{n}\sum_{k=1}^{d} \big(G(u \otimes 1_d)\big)_{ik} \cdot u_i \\
&= \textstyle\sum_{i=1}^{n}\sum_{k=1}^{d}\sum_{j=1}^{n}\sum_{\ell=1}^{d} G_{ik,j\ell} \cdot u_j \cdot u_i \quad \text{(from (6.11))} \\
&= d \cdot \textstyle\sum_{i=1}^{n}\sum_{k=1}^{d} P_{ij} \cdot u_j \cdot u_i \qquad\qquad \text{(from (6.8)),}
\end{aligned} \tag{6.12}$$

while on the other hand:

$$\big\langle Pu, u\big\rangle = \sum_{i=1}^{n}(Pu)_i \cdot u_i = \sum_{i=1}^{n}\sum_{j=1}^{d} P_{ij} \cdot u_j \cdot u_i. \tag{6.13}$$

So combining (6.12) and (6.13) we see that

$$\big\langle Gw, w\big\rangle = d \cdot \big\langle Pu, u\big\rangle. \tag{6.14}$$

Furthermore since $w = u \otimes 1_d$ we have

$$\langle w, w\rangle = \langle u, u\rangle \cdot \langle 1_d, 1_d\rangle = d \cdot \langle u, u\rangle. \tag{6.15}$$

So we can deduce from (6.14) and (6.15) that (6.10) holds:

$$\frac{\langle Pu, u\rangle}{\langle u, u\rangle} = \frac{\langle Gw, w\rangle}{\langle w, w\rangle}. \tag{6.16}$$

■

Next, we define the graphs $\hat{A}$ and $\hat{C}$ from which we will construct the large graph of which $A \circledS C$ is a projection.

Definition 6.4. Let $\hat{C}$ be the graph with vertex set $[n] \times [d]$, defined as

$$\hat{C} = I_n \otimes C.$$

$\hat{C}$ consists of n clouds, and each cloud is a copy of C. There are no edges between the clouds.

Definition 6.5. Let $\hat{A}$ be the graph with vertex set $[n] \times [d]$ defined as the de-projection of A:

$$\hat{A} = \text{DP}[A].$$

The de-projection operation was introduced in Subsection 5.6.7. $\hat{A}$ is 1-regular, so one step of a random walk on $\hat{A}$ is an permutation (in fact an involution). There are edges between clouds, but no edges within the clouds.

Proposition 6.6. *Let $\hat{A}$ and $\hat{C}$ be defined as above. Then*

$$A \textcircled{S} C = \text{P}_n[\hat{A}\hat{C}\hat{A}].$$

Since the degrees of $\hat{A}$ and $\hat{C}$ are 1 and t respectively, the degree of $\hat{A}\hat{C}\hat{A}$ is t. There are d nodes in each cloud, so there are dt edges leaving each cloud. When a cloud gets collapsed into a single vertex by the projection, all these edges are kept, which means that $\text{P}_n[\hat{A}\hat{C}\hat{A}]$ has degree dt.

In terms of random walks, we can get an intuition as to why this holds. A step labeled $\ell \in [t]$ from a vertex $(a, i) \in [n] \times [d]$ in $\hat{A}\hat{C}\hat{A}$ can be decomposed as

1. A substep in between clouds (in $\hat{A}$): $(a, i) \to (a[i], i)$. This is deterministic.
2. A substep within the new cloud (in $\hat{C}$): $(a[i], i) \to (a[i], i[\ell])$. There are t choices for this substep.
3. A substep in between clouds (in $\hat{A}$): $(a[i], i[\ell]) \to (a[i][i[\ell]], i[\ell])$. This is deterministic.

Now an edge $(a, i) \to (a[i][i[\ell]], i[\ell])$ in $\hat{A}\hat{C}\hat{A}$ will become an edge $a \to a[i][i[\ell]]$ in the projected graph.

Relationship with the zig-zag product

Interestingly, as was shown in the original zig-zag product paper [61], we have the following equality:

$$A \textcircled{Z} C = \hat{C}\hat{A}\hat{C}. \tag{6.17}$$

This also leads to an interpretation of the following equality from the original paper on derandomized squaring [64]:

$$d^2 \cdot A \textcircled{S} (C^2) = \text{P}_n[(A \textcircled{Z} C)^2]. \tag{6.18}$$

The multiplication by d^2 means that each edge is duplicated d^2 times. The left hand side is equal to $\text{P}_n[\hat{A}\hat{C}^2\hat{A}]$, while the right hand side is equal to $\text{P}_n[(\hat{C}\hat{A}\hat{C})^2] = \text{P}_n[\hat{C}\hat{A}\hat{C}^2\hat{A}\hat{C}]$. But since $\hat{C}$ has edges only within the clouds, the first and last $\hat{C}$ will have no effect on the projected ("collapsed") graph, and so both sides are equal.

6.3.3 Bounding the Second Eigenvalue

To bound the second eigenvalue of $A \textcircled{S} C$ we will apply Proposition 6.3 and analyze $\hat{A}\hat{C}\hat{A}$. We will be using the following definition throughout this chapter:

Definition 6.7. Whenever w is a vector in $\mathbb{R}^{nd}$, we can view w as consisting of n blocks of size d. We index its entries with the set $[n] \times [d]$, so that w_{ij} indexes the j^{th} entry in block i. We define the map $\mathcal{M} : \mathbb{R}^{nd} \to \mathbb{R}^n$ as

$$\big(\mathcal{M}_n(w)\big)_i = \sum_{j=1}^{d} w_{ij}.$$

So if we have a graph consisting of n clouds of size d (i.e., with vertex set $[n] \times [d]$), and if w is a probability distribution on the set of vertices, then $\mathcal{M}_n(w)$ is the marginal distribution on the set of clouds. This operation can be seen as a projection for vectors.

The following lemma establishes a useful relationship between A and $\hat{A} = \text{DP}[A]$:

Lemma 6.8. *Let $\hat{A} = \text{DP}[A]$ be defined as above. Then for any $\sigma \in \mathbb{R}^n$ we have*

$$\mathcal{M}_n\big(\hat{A}(\sigma \otimes \frac{1_d}{d})\big) = A\sigma. \tag{6.19}$$

Proof: See Appendix C. ■

We are now ready to prove the main result of this section.

Theorem 6.9. *Let A and C be as above. Then we have*

$$\lambda_{A \circledS C} \leq \lambda_A^2 + \lambda_C \cdot (1 - \lambda_A^2).$$

Proof: From Proposition 6.3 we know that

$$\lambda_{A \circledS C} = \max_{x \in 1_n^{\perp} \otimes 1_d^{\parallel}} \frac{\big|\langle \hat{A}\hat{C}\hat{A}x, x\rangle\big|}{\langle x, x\rangle}.$$

Let $x \in 1_n^{\perp} \otimes 1_d^{\parallel}$. We define

$$\gamma = \hat{A}x, \tag{6.20}$$

and then let

$$\gamma^{\parallel} = \mathcal{M}_n(\gamma) \otimes \frac{1_d}{d}, \qquad \text{and} \qquad \gamma^{\perp} = \gamma - \gamma^{\parallel}. \tag{6.21}$$

So $\gamma^{\parallel}$ is uniform over each cloud, $\gamma^{\perp}$ is anti-uniform over each cloud, and $\gamma = \gamma^{\parallel} + \gamma^{\perp}$. We will use the two following claims:

Claim 1: $\hat{C}\gamma^{\parallel} = \gamma^{\parallel}$.
Proof: We have

$$\hat{C}\gamma^{\parallel} = \big(I_n \otimes C\big) \cdot \big(\mathcal{M}_n(\gamma) \otimes \frac{1_d}{d}\big) = \mathcal{M}_n(\gamma) \otimes C \cdot \frac{1_d}{d}. \tag{6.22}$$

Now because C is doubly stochastic, $C \cdot \frac{1_d}{d} = \frac{1_d}{d}$. Therefore

$$\hat{C}\gamma^{\parallel} = \gamma^{\parallel}. \tag{6.23}$$

□

Claim 2: $\left|\langle \hat{C}\gamma^{\perp}, \gamma^{\perp}\rangle\right| \leq \lambda_C \cdot \langle \gamma^{\perp}, \gamma^{\perp}\rangle$.
Proof: We can decompose $\gamma^{\perp} \in \mathbb{R}^{nd}$ as

$$\gamma^{\perp} = \begin{pmatrix} \gamma_1^{\perp} \\ \vdots \\ \gamma_n^{\perp} \end{pmatrix}, \tag{6.24}$$

where $\gamma_1^{\perp}, \ldots, \gamma_n^{\perp} \in 1_d^{\perp}$. So since $\hat{C} = I_n \otimes C$ this gives us

$$\hat{C}\gamma^{\perp} = \begin{pmatrix} C\gamma_1^{\perp} \\ \vdots \\ C\gamma_n^{\perp} \end{pmatrix}. \tag{6.25}$$

Therefore

$$\begin{aligned} \left|\langle \hat{C}\gamma^{\perp}, \gamma^{\perp}\rangle\right| &= \left|\textstyle\sum_{i=1}^{n} \langle C\gamma_i^{\perp}, \gamma_i^{\perp}\rangle\right| \\ &\leq \textstyle\sum_{i=1}^{n} \left|\langle C\gamma_i^{\perp}, \gamma_i^{\perp}\rangle\right| \\ &\leq \textstyle\sum_{i=1}^{n} \lambda_C \cdot \left|\langle \gamma_i^{\perp}, \gamma_i^{\perp}\rangle\right| \quad \text{(by the definition of } \lambda_C\text{)} \\ &\leq \lambda_C \cdot \langle \gamma^{\perp}, \gamma^{\perp}\rangle. \end{aligned} \tag{6.26}$$

□

Continuing with our proof, we have

$$\begin{aligned} \left|\langle \hat{A}\hat{C}\hat{A}x, x\rangle\right| &= \left|\langle \hat{C}\hat{A}x, \hat{A}x\rangle\right| \qquad \text{(since } \hat{A} \text{ is symmetric)} \\ &= \left|\langle \hat{C}\gamma, \gamma\rangle\right| \\ &= \left|\langle \hat{C}\gamma^{\|}, \gamma^{\|}\rangle + \overbrace{\langle \hat{C}\gamma^{\|}, \gamma^{\perp}\rangle}^{0} + \overbrace{\langle \hat{C}\gamma^{\perp}, \gamma^{\|}\rangle}^{0} + \langle \hat{C}\gamma^{\perp}, \gamma^{\perp}\rangle\right| \\ &\leq \left|\langle \hat{C}\gamma^{\|}, \gamma^{\|}\rangle\right| + \left|\langle \hat{C}\gamma^{\perp}, \gamma^{\perp}\rangle\right| \\ &= \left|\langle \gamma^{\|}, \gamma^{\|}\rangle\right| + \left|\langle \hat{C}\gamma^{\perp}, \gamma^{\perp}\rangle\right| \qquad \text{(from Claim 1).} \end{aligned}$$

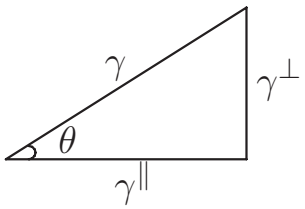
And so we deduce:

$$\frac{\left|\langle \hat{A}\hat{C}\hat{A}x, x\rangle\right|}{\langle x, x\rangle} \leq \frac{\|\gamma^{\|}\|^2}{\|x\|^2} + \frac{\left|\langle \hat{C}\gamma^{\perp}, \gamma^{\perp}\rangle\right|}{\|x\|^2}.$$

Now $\hat{A}$ is a permutation, which means that it is length-preserving. Therefore $\|\gamma\| = \|\hat{A} \cdot x\| = \|x\|$, and furthermore, from Claim 2 we have $|\langle \hat{C}\gamma^{\perp}, \gamma^{\perp}\rangle| \leq \lambda_C \langle \gamma^{\perp}, \gamma^{\perp}\rangle$. This leads to:

$$\frac{\left|\langle \hat{A}\hat{C}\hat{A}x, x\rangle\right|}{\langle x, x\rangle} \leq \frac{\|\gamma^{\|}\|^2}{\|\gamma\|^2} + \lambda_C \cdot \frac{\|\gamma^{\perp}\|^2}{\|\gamma\|^2}. \tag{6.27}$$

If we let θ be the angle between γ and $\gamma^{\|}$, then we have the following diagram



and so (6.27) becomes

$$\begin{aligned} \frac{\left|\langle \hat{A}\hat{C}\hat{A}x, x\rangle\right|}{\langle x,x\rangle} &\leq \cos^2(\theta) + \lambda_C \cdot \sin^2(\theta) \\ &= \cos^2(\theta) + \lambda_C \cdot \big(1 - \cos^2(\theta)\big) \\ &= (1-\lambda_C)\cdot \cos^2(\theta) + \lambda_C. \end{aligned} \tag{6.28}$$

Now clearly, since $1 - \lambda_C \geq 0$, this expression will be maximal when $\cos^2(\theta)$ is maximal.

Claim: $\cos(\theta) \leq \lambda_A$
Proof: We have:

$$\gamma^{\|} = \mathcal{M}_n(\gamma) \otimes \frac{1_d}{d} = \mathcal{M}_n(\hat{A}x) \otimes \frac{1_d}{d}.$$

Since $x \in 1_n^{\perp} \otimes 1_d^{\|}$, there is $u \in 1_n^{\perp}$ with $x = u \otimes 1_d$:

$$\gamma^{\|} = \mathcal{M}_n\big(\hat{A}(u \otimes 1_d)\big) \otimes \frac{1_d}{d}.$$

Using Lemma 6.8 we obtain:

$$\gamma^{\|} = d \cdot (Au) \otimes \frac{1_d}{d} = (Au) \otimes 1_d. \tag{6.29}$$

So this gives us

$$\|\gamma^{\|}\| = \|Au\| \cdot \|1_d\| \leq \lambda_A \cdot \|u\| \cdot \frac{1}{\sqrt{d}}, \tag{6.30}$$

by the definition of λ_A, since $u \in 1_n^{\perp}$. Recall that $\hat{A}$ is a permutation, it is therefore length preserving and so we obtain:

$$\|\gamma\| = \|\hat{A}x\| = \|x\| = \|u \otimes 1_d\| = \|u\| \cdot \frac{1}{\sqrt{d}}. \tag{6.31}$$

Combining (6.30) and (6.31) yields

$$\cos(\theta) = \frac{\|\gamma^{\|}\|}{\|\gamma\|} \leq \lambda_A,$$

as required. □

Combining this Claim with (6.28) gives us

$$\lambda_{A \circledS C} \leq \lambda_A^2 + \lambda_C \cdot (1 - \lambda_A^2).$$

■

6.4 Derandomized Tensoring

6.4.1 The Product

The tensor product of two graphs A and B enables the construction of a larger graph with expansion properties no worse than A and B but at the cost of a large increase in the degree.

$A Ⓢ C$ effectively consisted in taking A^2 and removing some edges in a clever way. Which edges to remove was determined by a second graph C. In the same way, the idea of derandomized tensoring is to remove edges from $A \otimes B$ based on a third graph C, which in this case will be a *bipartite* graph. We will see that the derandomized tensor product can in some cases reduce the degree of the resulting graph while preserving its expansion properties.

Let A be an $[n, d_1]$-graph, and B be an $[m, d_2]$-graph. Furthermore let C be a bipartite graph with d_1 left nodes and d_2 rights nodes. Keeping the notation from Subsection 5.7.1, we have $d = d_1 + d_2$ and we label the edges of A and B with the sets D_1 and D_2 respectively, where

$$\begin{aligned} D_1 &= \{1, \ldots, d_1\} \\ D_2 &= \{d_1 + 1, \ldots, d\}, \end{aligned}$$

so that $[d] = D_1 \sqcup D_2$.

The tensor product $A \otimes B$ has vertex set $[n] \times [m]$, which we interpret as n clouds of size m. Recall from its description in Section 5.6.3 that a step in $A \otimes B$ can be decomposed into two parts: first a step between clouds, and then a step within the new cloud (we presented them in this order, though they could also be done the other way round). The first step has a label $i \in D_1$, while the second step has a label $j \in D_2$. In the *derandomized tensor product* of A and B with respect to C, denoted $A \otimes_C B$ we take only the steps (i, j) for which i and j are connected in C.

We can describe $A \otimes_C B$ as the graph with vertex set $[n] \times [m]$, and in which there is an edge from (a, b) to (u, v) if and only if the following conditions hold:

1. There is an edge from a to u in A: There is $i \in D_1$ with $a[i] = u$.
2. There is an edge from b to v in B: There is $j \in D_2$ with $b[j] = v$.
3. i and j are connected in C.

Notice that if we remove the third condition then we obtain the normal tensor product. Also, if C is the complete bipartite graph then whenever the first two conditions are verified then so is the third one, and so in this case the product is also equal to the normal tensor product.

The degree of $A \otimes_C B$ is equal to the number of edges in C. Although this product is defined for any bipartite graph C of the right dimensions, we will be concerned only with the cases in which C is *biregular*. The spectrum and expansion properties of such graphs were discussed in Section 5.7.

A biregular bipartite graph of left and right degrees ℓ and r respectively can be labeled in the following way: the edges are labeled with elements of $[\ell]$ at their left ends, and with elements of $[r]$ at their right ends, in such a way that the edges adjacent to a given node all have distinct labels. For a left node $i \in D_1$ and a label $k \in [\ell]$, $i[k] \in D_2$ denotes the k^{th} neighbor of i (so $i[k]$ is a right node). Because the degree of the

derandomized tensor product is equal to the number of edges in C, we have

$$\deg\big(A \otimes_C B\big) = d_1\ell = d_2 r. \tag{6.32}$$

We give the formal definition only for the case in which C is biregular.

Definition 6.10. Let A be an $[n, d_1]$-graph, B be an $[n, d_2]$-graph and C be a biregular bipartite graph with d_1 left nodes, d_2 right nodes and of left and right degrees ℓ and r respectively. The derandomized tensor product $A \otimes_C B$ of A and B with respect to C is an $[nm, d_1\ell]$-graph with vertex set $[n] \times [m]$ and a labeling in $D_1 \times [\ell]$. For any $(a, b) \in [n] \times [m]$ and $(i, k) \in D_1 \times [\ell]$ we have

$$(a, b)[i, k] = (a[i], b[i[k]]).$$

We could of course have presented an equivalent definition with a labeling in $D_2 \times [r]$. If the labelings of A and B are colorings then $A \otimes_C B$ will be undirected. However, if the labelings are only half-colorings then to ensure that the product is undirected C also needs to have the following property: for any $i \in D_1, j \in D_2$, i and j are connected in C if and only if $\rho_A(i)$ and $\rho_A(j)$ are connected in C.

Our aim is to analyze the expansion properties of $G = A \otimes_C B$, more precisely to upper bound its second eigenvalue λ_G as a function of λ_A, λ_B and λ_C.

The main result of this section is the following theorem:

Theorem 6.11. *Let A, B and C be graphs as described above, in which the labelings of A and B are half-colorings. Suppose without loss of generality that $\lambda_B \leq \lambda_A$. If $G = A \otimes_C B$ then*

$$\lambda_G \leq \max\left(\lambda_A, \lambda_B, m(\lambda_A, \lambda_B, \lambda_C)\right),$$

where $f(a, b, c) = ab + c\sqrt{(1 - a^2)(1 - b^2)}$, $g(b, c) = \frac{1}{\sqrt{\frac{c^2}{b^2} - c^2 + 1}}$, and

$$m(a, b, c) = f\big(\ \min(a, g(b, c)), b, c)\ \big). \tag{6.33}$$

Notice that if C is the complete bipartite graph, then $\lambda_C = 0$, and so $g(\lambda_B, \lambda_C) = 1 \geq \lambda_A$, and therefore our bound becomes $\max\big(\lambda_A, \lambda_B, \lambda_A\lambda_B\big) = \max\big(\lambda_A, \lambda_B\big)$, which is the same as that of the normal tensor product, as would be expected.

Also, if

$$m(\lambda_A, \lambda_B, \lambda_C) \leq \max(\lambda_A, \lambda_B) = \lambda_{A\otimes B} \tag{6.34}$$

then $A \otimes_C B$ has expansion properties at least as good as those of $A \otimes B$, but with a smaller degree.

If $\lambda_A = \lambda_B$ then we always have $\lambda_A \leq g(\lambda_B, \lambda_C)$, and so we obtain the simpler expression:

Theorem 6.12. *Suppose that $\lambda_A = \lambda_B$. If $G = A \otimes_C B$ then*

$$\lambda_G \leq \max\left(\lambda_A, F(\lambda_A, \lambda_C)\right),$$

where $F(a, c) = a^2 + c \cdot (1 - a^2)$.

Interestingly, $F(a, c)$ is the same bound as for derandomized squaring.

The idea behind the analysis is conceptually the same as that in the previous section, namely to view $A \otimes_C B$ as the projection of a larger graph, and study this larger graph. However, while previously we could index our vertices with a two-dimensional array $[n] \times [d]$, in this case three dimensions will be required, which will make both the notation and the proofs rather technical.

6.4.2 Notation

All the notations defined in Subsection 5.7.1 will still hold. We summarize them below:

• If d_1 and d_2 are the degrees of A and B as above, then we let $d = d_1 + d_2$ and define

$$\begin{array}{rcl} D_1 & = & \{1, \ldots, d_1\} \\ D_2 & = & \{d_1 + 1, \ldots, d\} \\ [d] & = & \{1, \ldots, d\} = D_1 \cup D_2. \end{array} \tag{6.35}$$

• We define the subspaces R_1 and R_2 of $\mathbb{R}^d$ as follows:

$$\begin{array}{rcl} R_1 & = & \{v \in \mathbb{R}^d \mid v_i = 0 \; \forall i \in D_2\} \\ R_2 & = & \{v \in \mathbb{R}^d \mid v_i = 0 \; \forall i \in D_1\}. \end{array} \tag{6.36}$$

• e_1 and e_2 denote the following vectors in $\mathbb{R}^d$:

$$(e_1)_i = \begin{cases} 1 & \text{if } i \in D_1 \\ 0 & \text{if } i \in D_2. \end{cases} \tag{6.37}$$

$$(e_2)_i = \begin{cases} 0 & \text{if } i \in D_1 \\ 1 & \text{if } i \in D_2. \end{cases} \tag{6.38}$$

Definition 6.13. For $i = 1, 2$, we have:

$$\begin{array}{l} e_i^{\|} = \{\beta \cdot e_i \mid \beta \in \mathbb{R}\}. \\ \\ e_i^{\perp} = \{v \in R_i \mid \langle v, e_i \rangle = 0\}. \end{array} \tag{6.39}$$

6.4.3 Definitions

As explained above, we will analyze the expansion of $A \otimes_C B$ by viewing it as a projection of a larger graph H. In this subsection we give the formal definitions required for the construction of H.

$A \otimes_C B$ has vertex set $[n] \times [m]$, and H will have vertex set $[n] \times [m] \times [d]$. We will suppose throughout this section that the labelings of A and B are half-colorings, for which

$$\rho_A : D_1 \to D_1, \qquad \text{and} \qquad \rho_B : D_2 \to D_2 \tag{6.40}$$

denote the partner mappings for A and B. ρ_A and ρ_B are involutions.

We start by defining the graphs $\hat{A}$, $\hat{B}$, $\hat{X}$ and $\hat{C}$.

Definition 6.14. $\hat{A}$ is a graph with vertex set $[n] \times [m] \times [d]$. Each vertex (a, b, c) has either one or no neighbors:

- If $c \in D_1$ then there is an edge from (a, b, c) to $(a[c], b, \rho_A(c))$.
- If $c \in D_2$ then (a, b, c) has no neighbors.

$\hat{B}$ is defined analogously:

Definition 6.15. $\hat{B}$ is a graph with vertex set $[n] \times [m] \times [d]$. Each vertex (a, b, c) has either one or no neighbors:

- If $c \in D_1$ then (a, b, c) has no neighbors.
- If $c \in D_2$ then there is an edge from (a, b, c) to $(a, b[c], \rho_B(c))$.

So $\hat{A}$ and $\hat{B}$ are not regular graphs. Notice that the two graphs "complement" each other in the sense that every vertex in $[n] \times [m] \times [d]$ has an edge either in $\hat{A}$ or in $\hat{B}$, but not in both. They can be naturally combined as follows:

Definition 6.16. $\hat{X}$ is the graph with vertex set $[n] \times [m] \times [d]$ defined as

$$\hat{X} = \hat{A} + \hat{B}.$$

So $\hat{X}$ is regular, it is a $[nmd, 1]$-graph. Since $\hat{X}$ has degree 1, one step of a random walk on $\hat{X}$ is an involution.

Definition 6.17. $\hat{C}$ is the graph with vertex set $[n] \times [m] \times [d]$ defined as

$$\hat{C} = I_n \otimes I_m \otimes C.$$

$\hat{C}$ can be interpreted as nm copies of C. We are now ready to characterize $A \otimes_C B$ as a projection:

Proposition 6.18. *Suppose we have graphs $A, B, C, \hat{X}$, and $\hat{C}$ defined as above. Then*

$$A \otimes_C B = \mathrm{P}_{nm}[\hat{X}\hat{C}\hat{X}]. \tag{6.41}$$

The large graphs we are considering have vertex set $[n] \times [m] \times [d]$. We can see this as nm copies of $[d]$, which we refer to as "C-clouds". The C-clouds get collapsed in the projection, leading to a graph with vertex set $[n] \times [m]$.

Example 6.19. We illustrate these constructions with the following graphs A, B and C:

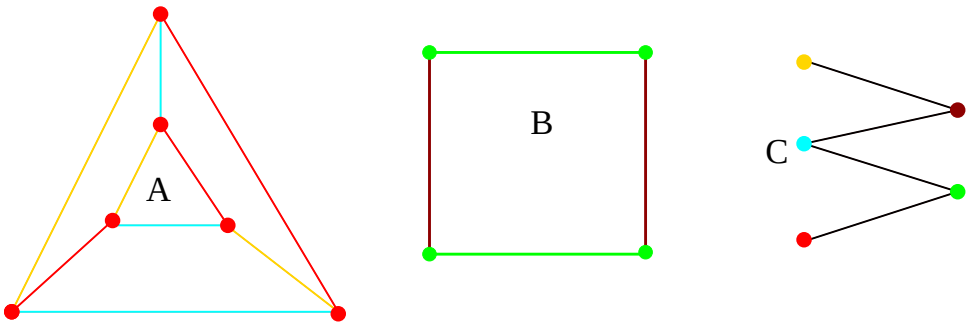


Figure 6.1: The graphs A, B and C.

So A is an $[n, d_1] = [6, 3]$-graph, B an $[m, d_2] = [4, 2]$-graph and C a bipartite graph with d_1 left vertices and d_2 right vertices (so $d = 5$ vertices in total). For this example C is not biregular. We suppose that the labelings of A and B are edge colorings to simplify the illustration (so each left vertex of C corresponds to a color of A and each right vertex to a color of B).

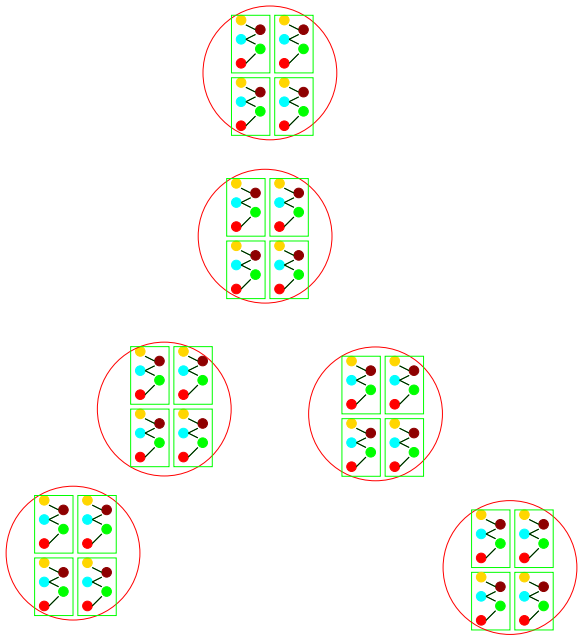


Figure 6.2: The graph $\hat{C}$.

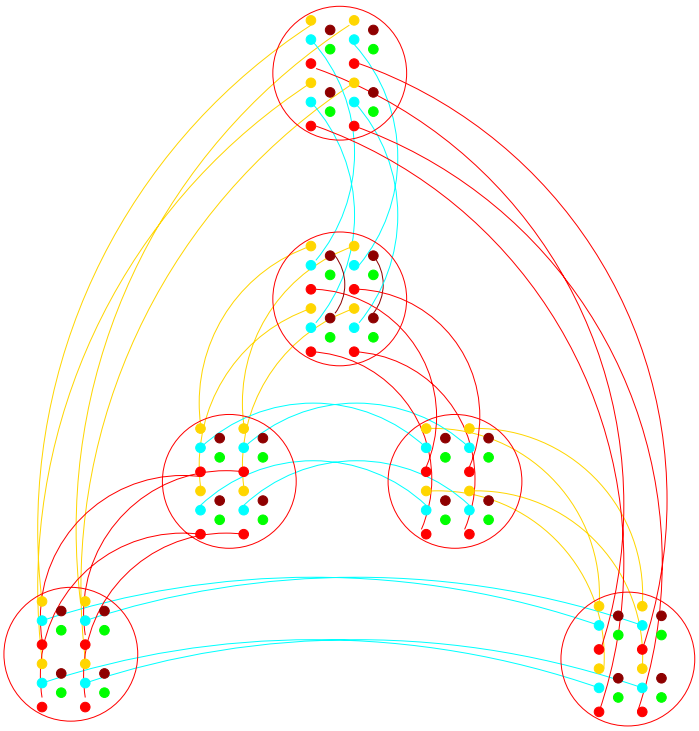


Figure 6.3: The graph $\hat{A}$.

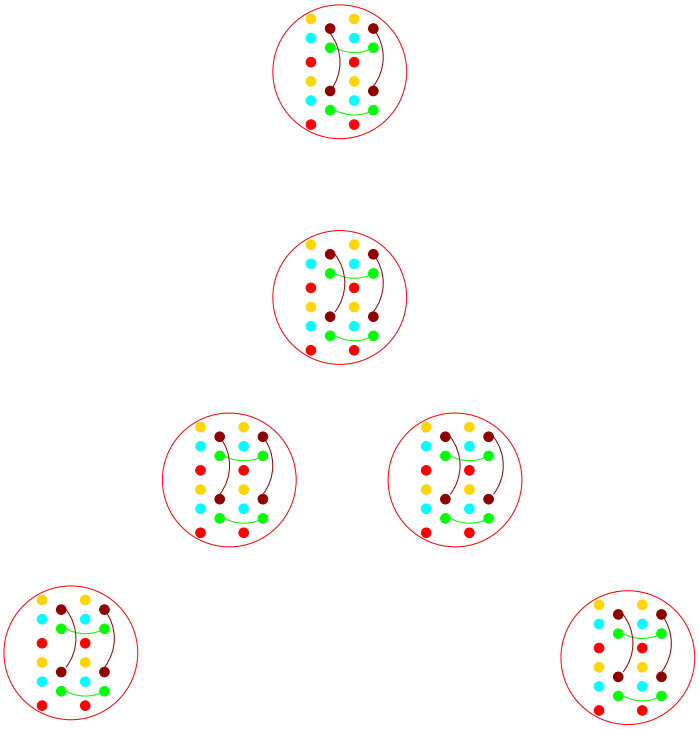


Figure 6.4: The graph $\hat{B}$.

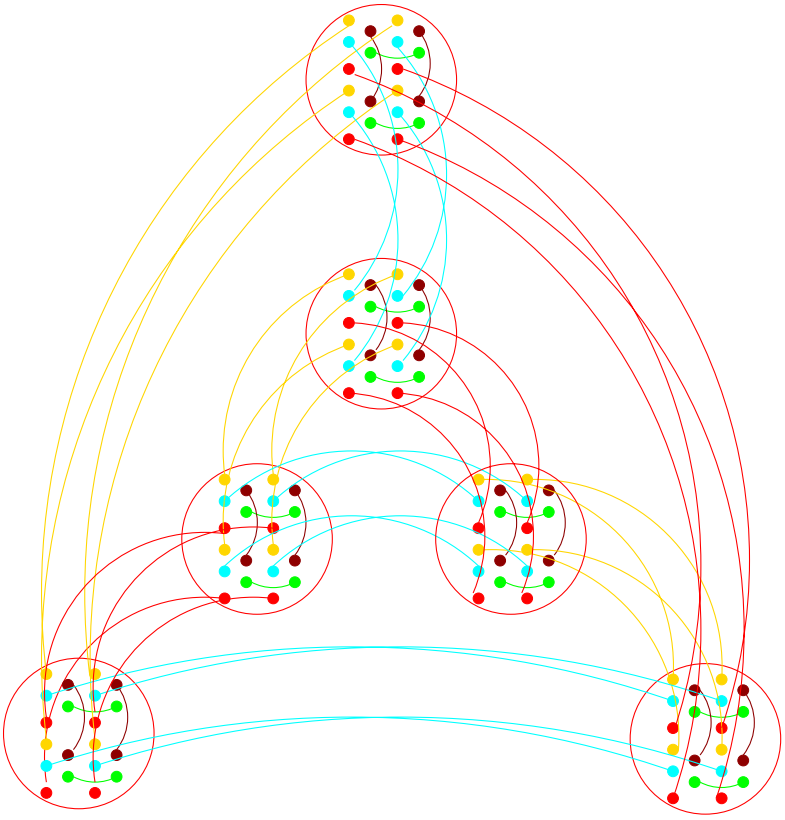


Figure 6.5: The graph $\hat{X}$.

The graphs $\hat{A}$ and $\hat{B}$ are related to the de-projections of A and B. We explore this relationship by first defining the following graphs:

Definition 6.20. $\overline{A}$ is a graph with vertex set $[n] \times [d]$. Each vertex (a, c) has either one or no neighbors:
- If $c \in D_1$ then there is an edge from (a, c) to $(a[c], \rho_A(c))$.
- If $c \in D_2$ then (a, c) has no neighbors.

Definition 6.21. $\overline{B}$ is a graph with vertex set $[m] \times [d]$. Each vertex (b, c) has either one or no neighbors:
- If $c \in D_1$ then (b, c) has no neighbors.
- If $c \in D_2$ then there is an edge from (b, c) to $(b[c], \rho_B(c))$.

$\overline{B}$ can be seen as the de-projection DP$[B]$ of B (which would have vertex set $[m] \times D_2$) to which are added some edgeless vertices, namely all those in $[m] \times D_1$.

It can be checked that

$$\hat{B} = I_n \otimes \bar{B}. \tag{6.42}$$

Example 6.22. If we use the graphs A, B and C from Example 6.19 above then $\overline{B}$ is the following graph:

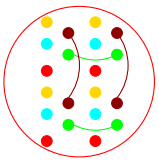


Figure 6.6: The graph $\overline{B}$.

We also see from Figure 6.4 that $\hat{B}$ consists of $n = 6$ copies of $\overline{B}$, so that the equality

$$\hat{B} = I_n \otimes \bar{B} \tag{6.43}$$

is verified. DP$[B]$, given below is the same but without the edgeless vertices:

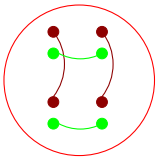


Figure 6.7: The graph DP$[B]$.

$\hat{B}$ is in a certain sense "two steps away" from DP$[B]$. The differences are:

1) There are some extra edgeless vertices (this corresponds to the difference between DP$[B]$ and $\overline{B}$). See Figures 6.6 and 6.7 of Example 6.22 for an illustration

2) There are n copies of this graph $\overline{B}$ (since $\hat{B} = I_n \otimes \overline{B}$).

There is a similar relationship between A, $\overline{A}$ and $\hat{A}$. However with our current tensor product notation it can only be expressed after an appropriate permutation of the indices (algebraically this is just a change of basis). Informally, by indexing the vertices with $[m] \times [n] \times [d]$ instead of $[n] \times [m] \times [d]$ we can write $\hat{A}$ as

$$I_m \otimes \bar{A}. \tag{6.44}$$

The relationship between $\hat{A}$ and DP$[A]$ is analogous to that between $\hat{B}$ and DP$[B]$.

Change of basis

Although it is intuitively quite clear that the change of basis we require is that in which $u \otimes v \otimes w$ becomes $v \otimes u \otimes w$, writing so formally is a little tedious. We include it nevertheless for completeness. Since the change of basis we need to do is just a permutation of the basis elements, the basis change matrix is a permutation matrix.

Definition 6.23. Consider the following $nm \times nm$ block matrix

$$L = \begin{pmatrix} Q_{11} & \cdots & Q_{1m} \\ \vdots & \vdots & \vdots \\ Q_{n1} & \cdots & Q_{nm} \end{pmatrix},$$

where $Q_{ij} \in \mathbb{R}^{m \times n}$ is defined as

$$(Q_{ij})_{k\ell} = \begin{cases} 1 & \text{if } k = j \text{ and } \ell = i \\ 0 & \text{otherwise.} \end{cases}$$

Let $P \in \mathbb{R}^{nmd \times nmd}$ be the matrix defined as

$$P = L \otimes I_d.$$

Notice that P is a permutation matrix, so it is invertible, and is therefore valid basis change matrix. the only basis change for $\mathbb{R}^{nmd}$ we will do is that given by P.

We will use the notation $x \sim_P y$ to denote the fact that y is the expression of x in the new basis ($y = Px$). So for example for any $u \in \mathbb{R}^n, v \in \mathbb{R}^m, w \in \mathbb{R}^d$ we have

$$u \otimes v \otimes w \sim_P v \otimes u \otimes w. \tag{6.45}$$

Clearly, if $x \sim_P x'$ and $y \sim_P y'$ then

$$\langle x, y \rangle = \langle x', y' \rangle. \tag{6.46}$$

$A \otimes_C B$ as a derandomized square

An alternative construction of $\hat{X}$ is to let $X = (A \otimes I_m) + (I_n \otimes B)$, and then set $\hat{X} = \text{DP}[X]$. The graph X is interesting in that it enables us to express $A \otimes_C B$ as a derandomized square. X has vertex set $[n] \times [m]$ and degree $d_1 + d_2 = d$. Its edges are given by

$$(a, b)[i] = \begin{cases} (a[i], b) & \text{if } i \in D_1 \\ (a, b[i]) & \text{if } i \in D_2. \end{cases}$$

Combining our description of the derandomized square as a graph projection in Section 6.3 and (6.41), we can deduce that

$$A \otimes_C B = X \textcircled{S} C. \tag{6.47}$$

A first approach to bounding the expansion of $A \otimes_C B$ would be to compute λ_X, and use the bound on $\lambda_{X \textcircled{S} C}$ from Theorem 6.9. However, this would not take into account the special structure of X, in terms of how it is constructed from A and B, and therefore how λ_X depends on λ_A and λ_B. A better bound can be obtained by making more use of the structure of X.

6.4.4 Proof Outline

In this subsection we outline the proof of Theorem 6.11. The full details can be found in Appendix B. Let $G = A \otimes_C B$. Since

$$G = \mathrm{P}_{nm}[\hat{X}\hat{C}\hat{X}],$$

we use Proposition 6.3 to deduce that

$$\lambda_G = \max_{x \in 1_{nm}^{\perp}} \frac{\left|\langle Gx, x\rangle\right|}{\langle x, x\rangle} = \max_{x \in 1_{nm}^{\perp} \otimes 1_d^{\parallel}} \frac{\left|\langle \hat{X}\hat{C}\hat{X}x, x\rangle\right|}{\langle x, x\rangle}. \tag{6.48}$$

Recalling that $\hat{X}$ was defined as $\hat{A} + \hat{B}$, we obtain:

$$\begin{aligned} |\langle \hat{X}\hat{C}\hat{X}x, x\rangle| &= |\langle \hat{C}\hat{X}x, \hat{X}x\rangle| && \text{(since } \hat{X} \text{ is symmetric)} \\ &= |\langle \hat{C}(\hat{A}+\hat{B})x, (\hat{A}+\hat{B})x\rangle| \\ &\leq \overbrace{|\langle \hat{C}\hat{A}x, \hat{A}x\rangle|}^{0} + |\langle \hat{C}\hat{A}x, \hat{B}x\rangle| + |\langle \hat{C}\hat{B}x, \hat{A}x\rangle| + \overbrace{|\langle \hat{C}\hat{B}x, \hat{B}x\rangle|}^{0}. \end{aligned}$$

Indeed $\hat{A}x \in R_1$ and $\hat{C}\hat{A}x \in R_2$, and likewise $\hat{B}x \in R_2$ and $\hat{C}\hat{B}x \in R_1$, which leads to

$$\langle \hat{C}\hat{A}x, \hat{A}x\rangle = 0, \qquad \langle \hat{C}\hat{B}x, \hat{B}x\rangle = 0.$$

We therefore have

$$|\langle \hat{X}\hat{C}\hat{X}x, x\rangle| \leq |\langle \hat{C}\hat{A}x, \hat{B}x\rangle| + |\langle \hat{C}\hat{B}x, \hat{A}x\rangle|. \tag{6.49}$$

Since $\hat{A}$ and $\hat{B}$ are symmetric this leads to

$$|\langle \hat{X}\hat{C}\hat{X}x, x\rangle| \leq |\langle \hat{B}\hat{C}\hat{A}x, x\rangle| + |\langle \hat{A}\hat{C}\hat{B}x, x\rangle|. \tag{6.50}$$

(In fact we have $\hat{X}\hat{C}\hat{X} = \hat{A}\hat{C}\hat{B} + \hat{B}\hat{C}\hat{A}$).

Now (6.48) tells us that we must consider vectors in the space $S = 1_{nm}^{\perp} \otimes 1_d^{\parallel}$, which we can decompose as

$$S = \underbrace{\left(1_n^{\perp} \otimes 1_m^{\parallel} \otimes 1_d^{\parallel}\right)}_{S_1} \oplus \underbrace{\left(1_n^{\parallel} \otimes 1_m^{\perp} \otimes 1_d^{\parallel}\right)}_{S_2} \oplus \underbrace{\left(1_n^{\perp} \otimes 1_m^{\perp} \otimes 1_d^{\parallel}\right)}_{S_3}. \tag{6.51}$$

This means that for any $x \in S$ there are unique $x_1 \in S_1$, $x_2 \in S_2$, and $x_3 \in S_3$ with

$$x = x_1 \ + \ x_2 \ + \ x_3. \tag{6.52}$$

Note that S_1, S_2 and S_3 have respective dimensions $n-1$, $m-1$ and $(n-1)(m-1)$. These add up to $nm - 1 = \dim(S)$, as expected.

We will first show (Lemma B.6) that

$$\langle \hat{A}\hat{C}\hat{B}x, x\rangle = \langle \hat{A}\hat{C}\hat{B}x_1, x_1\rangle + \langle \hat{A}\hat{C}\hat{B}x_2, x_2\rangle + \langle \hat{A}\hat{C}\hat{B}x_3, x_3\rangle. \tag{6.53}$$

The intuition is that the images of the spaces S_1, S_2 and S_3 under the linear transformation defined by $\hat{A}\hat{C}\hat{B}$ are also pairwise orthogonal. Therefore when we expand the left hand side of (6.53) only the terms on the right hand side will remain. Likewise we will have

$$\langle \hat{B}\hat{C}\hat{A}x, x\rangle = \langle \hat{B}\hat{C}\hat{A}x_1, x_1\rangle + \langle \hat{B}\hat{C}\hat{A}x_2, x_2\rangle + \langle \hat{B}\hat{C}\hat{A}x_3, x_3\rangle. \tag{6.54}$$

Next, letting $m(a, b, c)$ be the function defined in Theorem 6.11, we will upper bound terms from (6.53) and (6.54) as follows:

$$\begin{aligned}
|\langle \hat{B}\hat{C}\hat{A}x_1, x_1\rangle| \ + \ |\langle \hat{A}\hat{C}\hat{B}x_1, x_1\rangle| \ &\leq \ \lambda_A \cdot \langle x_1, x_1\rangle. \\
\big|\langle \hat{B}\hat{C}\hat{A}x_2, x_2\rangle\big| \ + \ \big|\langle \hat{A}\hat{C}\hat{B}x_2, x_2\rangle\big| \ &\leq \ \lambda_B \cdot \langle x_2, x_2\rangle. \\
\big|\langle \hat{B}\hat{C}\hat{A}x_3, x_3\rangle\big| \ + \ \big|\langle \hat{A}\hat{C}\hat{B}x_3, x_3\rangle\big| \ &\leq \ m(\lambda_A, \lambda_B, \lambda_C) \cdot \langle x_3, x_3\rangle.
\end{aligned} \tag{6.55}$$

Because x_1, x_2 and x_3 are pairwise orthogonal, (see (6.51)) and $x = x_1 + x_2 + x_3$, we have

$$\langle x, x\rangle \ = \ \langle x_1, x_1\rangle \ + \ \langle x_2, x_2\rangle \ + \ \langle x_3, x_3\rangle. \tag{6.56}$$

We know from (6.50) that

$$|\langle \hat{X}\hat{C}\hat{X}x, x\rangle| \ \leq \ |\langle \hat{B}\hat{C}\hat{A}x, x\rangle| \ + \ |\langle \hat{A}\hat{C}\hat{B}x, x\rangle|. \tag{6.57}$$

So combining (6.53), (6.54) with the inequalities in (6.55) leads to

$$\begin{aligned}
|\langle \hat{X}\hat{C}\hat{X}x, x\rangle| \ &\leq \ \lambda_A \cdot \langle x_1, x_1\rangle + \lambda_B \cdot \langle x_2, x_2\rangle + m(\lambda_A, \lambda_B, \lambda_C) \cdot \langle x_3, x_3\rangle \\
&\leq \ \max\Big(\lambda_A, \lambda_B, m(\lambda_A, \lambda_B, \lambda_C)\Big) \cdot \Big(\langle x_1, x_1\rangle \ + \ \langle x_2, x_2\rangle \ + \ \langle x_3, x_3\rangle\Big) \\
&= \ \max\Big(\lambda_A, \lambda_B, m(\lambda_A, \lambda_B, \lambda_C)\Big) \cdot \big\langle x, x\big\rangle \qquad \text{(using (6.56)).}
\end{aligned} \tag{6.58}$$

The result of Theorem 6.11 then follows immediately:

$$\frac{|\langle \hat{X}\hat{C}\hat{X}x, x\rangle|}{\langle x, x\rangle} \leq \max\Big(\lambda_A, \lambda_B, m(\lambda_A, \lambda_B, \lambda_C)\Big). \tag{6.59}$$

For the full proof, see Appendix B.

6.5 Derandomized Code Concatenation

6.5.1 Introduction

The operation of *code concatenation* was first presented by Forney in [25]. Suppose we have a finite field $\mathbb{F}_{q_1}$ and an extension $\mathbb{F}_{q_2}$ of $\mathbb{F}_{q_1}$ of degree m. An *outer code* $\mathcal{C}_1$ over $\mathbb{F}_{q_2}$ can be concatenated with an *inner code* $\mathcal{C}_1$ over $\mathbb{F}_{q_1}$ of dimension m, which will yield a longer $\mathbb{F}_{q_1}$-code.

Concatenated codes are particularly useful for the construction of explicit families of good codes. For the binary case, the constructions of Justesen [37], Zyablov [95], Bloch-Zyablov [9] (using *multilevel concatenation*) and Katsman-Tsfasman-Vladut [38] are all concatenations. This last construction, obtained by concatenating a very good fixed binary code (the inner code) with a family of Algebraic-Geometric codes beyond the Gilbert-Varshamov bound (the outer code), yields some of the best known explicit families of binary codes.

An improved version of the concatenation operation is therefore potentially very interesting in the quest for better explicit families. The derandomization presented in this section improves the rate of the concatenated code at the cost of decreasing its relative distance (by how much depends on how good the expander we employ is).

It is a recurring theme in coding theory that random constructions yield good codes with high probability, but doing so explicitly is much more difficult (i.e., finding a way of *guaranteeing* a good code). Although the *expected* code is good, there will be some variance in the experiment which also makes very bad codes possible (very far from the expected result). Expander graphs have the remarkable property that they enable fairly good "simulations" of random behavior. More precisely one can in some contexts use expanders to achieve deterministic (i.e., non-random) behavior within a close range of the expectation of a random experiment. So if the expectation is good, then the result is guaranteed to be almost as good.

In this section we essentially use the fact that randomly puncturing a code will improve its rate, while keeping its *expected* relative distance fixed. So we employ an expander graph to simulate this random puncturing. In general, it is an interesting problem to find good ways of puncturing codes. For example an AG-code can be seen as a puncturing of a product of two or more Reed-Solomon codes, and it would be interesting to study the properties of the corresponding puncturing pattern.

6.5.2 Definitions

We will consider only binary inner codes. We will suppose throughout this section that we have an $[n_1, k_1, d_1]_q$-code $\mathcal{C}_1$, and an $[n_2, k_2, d_2]_2$-code $\mathcal{C}_2$, where $q = 2^{k_2}$ (so $\mathbb{F}_q$ is an extension of $\mathbb{F}_2$ of degree k_2). We will also suppose that we have a fixed basis of $\mathbb{F}_q$ over $\mathbb{F}_2$, which leads to a natural bijection

$$\sigma : \mathbb{F}_q \to \mathbb{F}_2^{k_2}. \tag{6.60}$$

We start by recalling the definition of code concatenation:

Definition 6.24. Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be as above. We define their *concatenation* $\mathcal{C} = \mathcal{C}_1 \Diamond \mathcal{C}_2$ as the $[n_1 n_2, k_1 k_2]_2$-code whose encoding map $E : \mathbb{F}_2^{k_1 k_2} \to \mathbb{F}_2^{n_1 n_2}$ can be decomposed as follows:

$$\mathbb{F}_2^{k_1 k_2} \longrightarrow^{\sigma^{-1}} \mathbb{F}_q^{k_1} \longrightarrow^{E_1} \mathbb{F}_q^{n_1} \longrightarrow^{\sigma} \mathbb{F}_2^{n_1 k_2} \longrightarrow^{E_2} \mathbb{F}_2^{n_1 n_2},$$

where E_1 and E_2 are the encoding maps of $\mathcal{C}_1$ and $\mathcal{C}_2$.

This amounts to first interpreting the message vector $u \in \mathbb{F}_2^{k_1 k_2}$ as a message vector in $\mathbb{F}_q^{k_1}$ for $\mathcal{C}_1$ and encoding it to a codeword $c_1 \in \mathcal{C}_1$. Then each of the n_1 components of c_1 are interpreted as message vectors of $\mathcal{C}_2$, and encoded to codewords in $\mathcal{C}_2$.

Notice that there is a bijection between $\mathcal{C}_1$ and $\mathcal{C}$ (but they have different dimensions since they use different alphabets). $\mathcal{C}$ consists of n_1 codewords of $\mathcal{C}_2$. We index the $n_1 n_2$ components of $c \in \mathcal{C}$ with the set $[n_1] \times [n_2]$ in the canonical way. We can write a codeword $c \in \mathcal{C}_1 \Diamond \mathcal{C}_2$ as

$$c = (x_1, \ldots, x_{n_1}), \tag{6.61}$$

where $x_j \in \mathcal{C}_2$ for all j.

We will now puncture $\mathcal{C}$, with a pattern that will be given by a bipartite expander graph. We suppose throughout this section that we have a biregular bipartite graph H with left vertex set $[n_1]$, right vertex set $[n_2]$, and of left and right degrees ℓ and r, respectively.

Definition 6.25. Let $\mathcal{C}_1, \mathcal{C}_2$ and H be as above. We construct the *derandomized concatenation* $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$ of $\mathcal{C}_1$ and $\mathcal{C}_2$ with respect to H by taking their normal concatenation $\mathcal{C}_1 \Diamond \mathcal{C}_2$, and then performing the following puncturing: we remove all components $(j, k) \in [n_1] \times [n_2]$ that are not connected in H.

The resulting code $\mathcal{C}$ will have length $n_1 \ell = n_2 r$ (which is equal to the number of edges in H).

6.5.3 The Rate of $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$

Proposition 6.26. *Let H be a biregular bipartite graph as above. Let $\mathcal{C}_1$ and $\mathcal{C}_2$ also be as above, and let R_1 and R_2 denote their respective rates. Let R be the rate of $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$.*

If $\ell > n_2 - d_2$ then

$$R = R_1 R_2 \frac{n_2}{\ell}. \tag{6.62}$$

Proof: We will show that $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$ has the same dimension as $\mathcal{C}_1 \Diamond \mathcal{C}_2$, namely $k_1 k_2$. It will suffice to show that no non-zero-codeword of $\mathcal{C}_1 \Diamond \mathcal{C}_2$ becomes the zero-codeword after the puncturing (since puncturing is a linear operation, this makes it injective). Let c be a codeword of $\mathcal{C}_1 \Diamond \mathcal{C}_2$, which we write as

$$c = (x_1, \ldots, x_{n_1}). \tag{6.63}$$

Now each non-zero $x_j \in \mathcal{C}_2$ gets punctured $n_2 - \ell$ times. Since x_j has weight at least d_2, as long as

$$n_2 - \ell < d_2 \tag{6.64}$$

x_j will not become the zero-codeword. In particular c cannot become the zero-codeword through this puncturing.

Since $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$ has dimension $k_1 k_2$ and length $n_1 \ell$, we can deduce that it has rate

$$R = \frac{k_1 k_2}{n_1 \ell} = \frac{k_1}{n_1} \frac{k_2}{n_2} \frac{n_2}{\ell} = R_1 R_2 \frac{n_2}{\ell}. \tag{6.65}$$

■

So as long as $\ell > n_2 - d_2$, by performing this puncturing we increase the rate of $\mathcal{C}_1 \Diamond \mathcal{C}_2$ by a factor of $\frac{n_2}{\ell}$. Notice that since $n_1 \ell = n_2 r$, we can write (6.62) as

$$R = R_1 R_2 \frac{n_2}{\ell} = R_1 R_2 \frac{n_1}{r} = R_1 \frac{k_2}{\ell} = R_2 \frac{k_1}{r}. \tag{6.66}$$

6.5.4 The Relative Distance of $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$

We start by giving the following bound on the relative distance of $\mathcal{C}_1 \Diamond \mathcal{C}_2$:

Proposition 6.27. *Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be as above. Then the minimum distance of $\mathcal{C}_1 \Diamond \mathcal{C}_2$ is at least $d_1 d_2$.*

Proof: Let c be a codeword of $\mathcal{C} = \mathcal{C}_1 \Diamond \mathcal{C}_2$. c can be constructed by taking some $c_1 \in \mathcal{C}_1$, interpreting each of its n_1 components as a $\mathcal{C}_2$-message vector, and then encoding these to obtain n_1 codewords $x_1, \ldots, x_{n_1}$ in $\mathcal{C}_2$. So we have $c = (x_1, \ldots, x_{n_1})$.

Note that $\text{wgt}(c)$ is equal to the sum of the weights of the x_i's. Since at least d_1 of these are non-zero (c_1 must have at least that many non-zero components), and each non-zero codeword of $\mathcal{C}_2$ has weight at least d_2, we have

$$\text{wgt}(c) \geq d_1 d_2, \tag{6.67}$$

as required. ■

Recall that $q = 2^{k_2} = |\mathcal{C}_2|$. We label the q codewords in $\mathcal{C}_2$ as

$$\mathcal{C}_2 = \{c_2^{(0)}, \ldots, c_2^{(q-1)}\}, \tag{6.68}$$

where $c_2^{(0)}$ denotes the zero-codeword.

Fix a codeword $\overline{c} \in \overline{\mathcal{C}} = \mathcal{C}_1 \Diamond_H \mathcal{C}_2$. $\overline{c}$ was obtained by puncturing some codeword c of $\mathcal{C}_1 \Diamond \mathcal{C}_2$. As seen above, c consists of n_1 codewords of $\mathcal{C}_2$:

$$c = \big(x_1, \ldots, x_{n_1}\big), \tag{6.69}$$

where $x_j \in \mathcal{C}_2$. For all $i \in [q-1]$ we define the sets S_i and T_i as

$$S_i = \big\{ j \in [n_1] \;\big|\; x_j = c_2^{(i)} \big\} \qquad \text{and} \qquad T_i = \text{supp}\big(c_2^{(i)}\big). \tag{6.70}$$

So we have $S_i \subseteq [n_1]$ and $T_i \subseteq [n_2]$. Notice that the S_i's depend on $c \in \mathcal{C}$ (and therefore on the codeword $\overline{c} \in \overline{\mathcal{C}}$ we fixed above), but the T_i's do not. This enables us to obtain the following expression for the weight of $\overline{c}$:

Lemma 6.28. *Let $\overline{c} \in \mathcal{C}_1 \Diamond_H \mathcal{C}_2$. If S_i and T_i are defined as above then*

$$\text{wgt}(\overline{c}) = \sum_{i=1}^{q-1} \big|\text{E}_H(S_i, T_i)\big|. \tag{6.71}$$

Proof: Let c be the codeword in $\mathcal{C}_1 \Diamond \mathcal{C}_2$ from which $\overline{c}$ was constructed, and let c_1 be the $\mathcal{C}_1$-codeword from which c was constructed. So $S_1, \ldots, S_{q-1}$ form a partition of $\text{supp}(c_1)$.

For each $i \in [q-1]$ we let $a_i = |S_i|$, and notice that

$$\sum_{i=1}^{q-1} a_i = \text{wgt}(c_1) \geq d_1. \tag{6.72}$$

Next, we let $b_i = |T_i| = \text{wgt}\big(c_2^{(i)}\big)$ so that for all $i \in [q-1]$ we have:

$$b_i \geq d_2. \tag{6.73}$$

In c there are a_i copies of the codeword $c_2^{(i)}$. So if there was no puncturing (or equivalently if H was the complete bipartite graph), the total weight of all the copies of $c_2^{(i)}$ would be $a_i b_i$, and so we would get

$$\text{wgt}(\overline{c}) = \sum_{i=1}^{q-1} a_i b_i \geq d_2 \sum_{i=1}^{q-1} a_i \geq d_1 d_2, \tag{6.74}$$

as expected.

When puncturing does occur, notice first of all that S_i is a subset of the left vertices of H, and T_i a subset of the right vertices of H. The non-zero components of c are exactly those whose index (j,k) is in $S_i \times T_i$ for some $i \in [q-1]$. Furthermore any component of c whose index $(j,k) \in [n_1] \times [n_2]$ is not an edge in H will be punctured out. So the total weight of all the copies of $c_2^{(i)}$ after puncturing will be equal to the number of edges between S_i and T_i. This gives us

$$\text{wgt}(\overline{c}) = \sum_{i=1}^{q-1} \big|\text{E}_H(S_i, T_i)\big|. \tag{6.75}$$

■

We have the intuition that if H is a good expander, then $\big|\text{E}_H(S_i, T_i)\big|$ should be close to its expected value in a random setting (i.e., when H is constructed randomly). This however is only true if S_i and T_i are not too small, for example nothing can be said of the case $|S_i| = 1$. We know that $|T_i|/n_2 \geq \delta_2$, but we have no guarantee on the size of S_i. Once again the intuition behind a large S_i is that we would like to apply many different puncturing patterns to the same codeword (here $c_2^{(i)}$) to ensure that the resulting average weight will be good.

If n_1 is much larger than q, then we are dividing a large set (namely $\text{supp}(c_1)$, of size $\delta_1 n_1$) into few subsets S_i (q of them), and so most elements of $\text{supp}(c_1)$ will be in a large subset, which is what we are looking for. So we can ensure that $\big|\text{E}(S_i, T_i)\big|$ will be close to its expected value in a random setting as long as q is not too large, or equivalently, as long as the rate of $\mathcal{C}_2$ is small (since $q = 2^{k_2}$).

The Expander Mixing Lemma (see Section 5.7.8) formalized the idea that in a good expander the number of edges between two sets of vertices is close to what would be expected in a random setting (the second eigenvalue of H determines how close). We restate the bipartite version below (see Section 5.7.8 for a proof).

Lemma 6.29. ***(The Bipartite Expander Mixing Lemma).***
Let H be a biregular bipartite graph, with left and right vertex sets $[n_1]$ and $[n_2]$ respectively, and left and right degrees ℓ and r. Let λ_H denote the second eigenvalue of H. For any $S \subseteq [n_1]$ and $T \subseteq [n_2]$ we have

$$\left| |E(S,T)| - \frac{\ell \cdot |S| \cdot |T|}{n_1} \right| \leq \lambda_H \cdot \sqrt{\ell r} \cdot \sqrt{|S| \cdot |T|}. \tag{6.76}$$

The main result of this section is the following theorem:

Theorem 6.30. *Let $\mathcal{C}_1, \mathcal{C}_2$ and H be as above. Suppose furthermore that $\lambda_H \leq \frac{1}{\sqrt{\ell r}}$, and $\ell > \frac{n_2}{2\sqrt{d_2}}$. If δ is the relative distance of $\mathcal{C}_1 \lozenge_H \mathcal{C}_2$ then*

$$\delta \geq \delta_1\delta_2 - \lambda_H \cdot \sqrt{q-1} \cdot \sqrt{\delta_1\delta_2}. \tag{6.77}$$

Requiring that $\lambda_H \leq \frac{1}{\sqrt{\ell r}}$ is saying that we want H to be quite a good expander (otherwise the puncturing could deviate too much from its expected behavior, opening up the possibility of "worst- case" scenarios about which we can say nothing).

Proof: Let $\overline{c} \in \mathcal{C}_1 \lozenge_H \mathcal{C}_2$ be a codeword constructed from $c_1 \in \mathcal{C}_1$ (as in the analysis above). Suppose $S_1, \ldots S_{q-1}$ and $T_1, \ldots, T_{q-1}$ are defined as above. We know from Lemma 6.28 that

$$\text{wgt}(\overline{c}) = \sum_{i=1}^{q-1} \big|\text{E}(S_i, T_i)\big|.$$

Setting $a_i = |S_i|$ and $b_i = |T_i|$, we deduce from the expander mixing lemma that

$$\text{wgt}(\overline{c}) \geq \sum_{i=1}^{q-1} \left(\frac{\ell a_i b_i}{n_2} - \lambda_H \cdot \sqrt{\ell r} \cdot \sqrt{a_i b_i} \right) = \overbrace{\frac{\ell}{n_2} \sum_{i=1}^{q-1} (a_i b_i)}^{A} - \overbrace{\lambda_H \cdot \sqrt{\ell r} \cdot \sum_{i=1}^{q-1} \sqrt{a_i b_i}}^{B}. \tag{6.78}$$

Notice that A corresponds to the "expected behavior" whereas B corresponds to the "error" (i.e., the variance). We know that

$$\sum_{i=1}^{q-1} a_i \geq d_1 \qquad \text{and} \qquad \forall i \in [q-1] : b_i \geq d_2, \tag{6.79}$$

We use this to deduce that

$$A = \frac{\ell}{n_2} \cdot \sum_{i=1}^{q-1} (a_i b_i) \geq \frac{\ell}{n_2} \cdot d_2 \cdot \sum_{i=1}^{q-1} a_i \geq \frac{\ell}{n_2} \cdot d_2 d_1. \tag{6.80}$$

This corresponds to the expected behavior in the "worst-case" scenario (i.e., c_1 has weight d_1 and all non zero $\mathcal{C}_2$-codewords have weight d_2).

Claim 1: $A - B$ is minimal when $b_i = d_2 \ \forall i \in [q-1]$.
Proof: d_2 is the smallest possible value each b_i can take. We will show that increasing any b_i cannot decrease the value of $A - B$. First note that if $a_i = 0$ then $A - B = 0$ for all values of b_i, so the statement holds. Suppose now that $a_i \geq 1$. For each $i = 1, \ldots, q-1$, we have

$$\frac{\partial}{\partial b_i}\big(A - B\big) = \frac{\ell}{n_2} \cdot a_i - \lambda_H \cdot \sqrt{\ell r} \cdot \frac{\sqrt{a_i}}{2} \cdot \frac{1}{\sqrt{b_i}}. \tag{6.81}$$

We therefore have:

$$\begin{aligned} \tfrac{\partial}{\partial b_i}\big(A - B\big) \geq 0 \quad &\iff \quad \lambda_H \cdot \sqrt{\ell r} \cdot \frac{\sqrt{a_i}}{2} \cdot \frac{1}{\sqrt{b_i}} \leq \tfrac{\ell}{n_2} \cdot a_i \\ &\iff \quad \lambda_H \cdot \sqrt{\ell r} \cdot \frac{n_2}{\ell} \cdot \frac{1}{2\sqrt{a_i}} \leq \sqrt{b_i}. \end{aligned} \tag{6.82}$$

We will show that the last line of (6.82) is always true. We call LHS and RHS respectively the left and right hand sides of this inequality. Because $\lambda_H \leq \frac{1}{\sqrt{\ell r}}$ and $a_i \geq 1$, we have LHS $\leq \frac{n_2}{2\ell}$. Recalling that $\ell \geq \frac{n_2}{2\sqrt{d_2}}$ (see the theorem statement), we have $\frac{n_2}{2\ell} \leq \sqrt{d_2}$. Finally for all $i \in [n]$: $\sqrt{d_2} \leq \sqrt{b_i}$. Combining all this we obtain

$$\text{LHS} \leq \frac{n_2}{2\ell} \leq \sqrt{d_2} \leq \sqrt{b_i} = \text{RHS}.$$

We see in (6.82) that this is equivalent to

$$\frac{\partial}{\partial b_i}\big(A - B\big) \geq 0.$$

So increasing b_i cannot decrease $A - B$ for any $i \in [q-1]$, so $A - B$ is minimal when all b_i's are set to their minimal value d_2. □

We let B_m denote the value of B when all b_i's are set to d_2. We would now like to upper bound B_m, which along with (6.80) will give us a lower bound on $A - B_m$ (and therefore on $A - B$ by Claim 1).

$$B_m \;=\; \lambda_H \cdot \sqrt{\ell r} \cdot \sum_{i=1}^{q-1} \sqrt{a_i b_i} \;=\; \lambda_H \cdot \sqrt{\ell r} \cdot \sqrt{d_2} \cdot \sum_{i=1}^{q-1} \sqrt{a_i}. \tag{6.83}$$

Claim 2: $\sum_{i=1}^{q-1} \sqrt{a_i} \;\leq\; \sqrt{q-1} \cdot \sqrt{d_1}$.

Proof: We will use the Cauchy-Schwarz inequality. We define the vector $v = \big(\sqrt{a_1}, \ldots, \sqrt{a_{q-1}}\big)^\top \in \mathbb{R}^{q-1}$, and let $1_{q-1} \in \mathbb{R}^{q-1}$ be the all one vector. We know (Cauchy-Schwarz) that

$$\langle v, 1_{q-1} \rangle^2 \leq \|v\|^2 \cdot \|1_{q-1}\|^2. \tag{6.84}$$

This means that

$$\begin{aligned} \big(\textstyle\sum_{i=1}^{q-1} v_i\big)^2 \leq \big(\sum_{i=1}^{q-1} v_i^2\big) \cdot \big(q-1\big) \quad &\Longrightarrow \quad \big(\textstyle\sum_{i=1}^{q-1} \sqrt{a_i}\big)^2 \leq \big(\sum_{i=1}^{q-1} a_i\big) \cdot \big(q-1\big) \\ &\Longrightarrow \quad \big(\textstyle\sum_{i=1}^{q-1} \sqrt{a_i}\big)^2 \leq d_1 \cdot \big(q-1\big) \\ &\Longrightarrow \quad \textstyle\sum_{i=1}^{q-1} \sqrt{a_i} \leq \sqrt{q-1} \cdot \sqrt{d_1}. \end{aligned} \tag{6.85}$$

□

Combining (6.83) with Claim 2, we deduce that

$$B_m \leq \lambda_H \cdot \sqrt{\ell r} \cdot \sqrt{d_2} \cdot \sqrt{q-1} \cdot \sqrt{d_1}. \tag{6.86}$$

We know from (6.78) and Claim 1 that $\text{wgt}(\overline{c}) \geq A - B_m$. (6.80) and (6.86) therefore lead to

$$\text{wgt}(\overline{c}) \geq \frac{\ell d_1 d_2}{n_2} - \lambda_H \cdot \sqrt{\ell r} \cdot \sqrt{d_2} \cdot \sqrt{q-1} \cdot \sqrt{d_1}, \tag{6.87}$$

which is also a lower bound on the minimum distance d of $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$. Since the length of $\mathcal{C}_1 \Diamond_H \mathcal{C}_2$ is ℓn_1, we can now obtain a bound on its relative distance δ:

$$\begin{aligned}
\delta &\geq \frac{\ell d_1 d_2}{\ell n_1 n_2} - \frac{\lambda_H \cdot \sqrt{\ell r} \cdot \sqrt{q-1} \cdot \sqrt{d_1 d_2}}{\ell n_1} \\
&= \delta_1 \delta_2 - \frac{\lambda_H \cdot \sqrt{r} \cdot \sqrt{q-1} \cdot \sqrt{d_1 d_2}}{\sqrt{\ell} \cdot \sqrt{n_1} \cdot \sqrt{n_1}} \\
&= \delta_1 \delta_2 - \lambda_H \cdot \sqrt{q-1} \cdot \sqrt{\frac{r}{\ell}} \cdot \sqrt{\frac{d_1}{n_1}} \cdot \sqrt{\frac{d_2}{n_1}} \qquad (6.88) \\
&= \delta_1 \delta_2 - \lambda_H \cdot \sqrt{q-1} \cdot \sqrt{\frac{r}{\ell}} \cdot \sqrt{\frac{d_1}{n_1}} \cdot \sqrt{\frac{d_2}{n_2}} \cdot \sqrt{\frac{\ell}{r}} \quad (\text{since } n_1 = \frac{n_2 r}{\ell}) \\
&= \delta_1 \delta_2 - \lambda_H \cdot \sqrt{q-1} \cdot \sqrt{\delta_1 \delta_2},
\end{aligned}$$

as required. ■

Appendix A

The Extension of the Binomial Function

This appendix deals with the extension of the binomial function from natural numbers to non negative real numbers. We give the required definitions, and prove some of the properties needed in Chapter 3. In particular, we prove Proposition 3.13.

The binomial function $\binom{a}{b}$ is a map

$$\binom{\cdot}{\cdot} : \mathbb{N} \times \mathbb{N} \to \mathbb{N},$$

which we will extend to a map

$$\binom{\cdot}{\cdot}' : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0},$$

with the property that

$$a, b \in \mathbb{N} \implies \binom{a}{b} = \binom{a}{b}'.$$

We will do this using the *gamma function* Γ, see ([91]), which is defined everywhere in $\mathbb{R}_{>0}$, and has the property that

$$\begin{aligned} \Gamma(1) &= 1 \\ \Gamma(z+1) &= z \cdot \Gamma(z). \end{aligned}$$

In particular, we can deduce from this that

$$n \in \mathbb{N} \implies \Gamma(n+1) = n!.$$

Because of this slightly inconvenient relation to the factorial function, we will also use Gauss's simpler notation

$$\Pi(z) = \Gamma(z+1),$$

which is defined over all of $\mathbb{R}_{\geq 0}$ and gives us the nicer relationship:

$$n \in \mathbb{N} \implies \Pi(n) = n!.$$

So Π can be seen as an extension of the factorial function from $\mathbb{N}$ to $\mathbb{R}_{\geq 0}$. We use this to define our extension of the binomial function:

Definition A.1. For any $a, b \in \mathbb{R}_{\geq 0}$ with $a \geq b$, we define:

$$\binom{a}{b}' = \begin{cases} \frac{\Pi(a)}{\Pi(a-b)\cdot\Pi(b)} & \text{if } a \geq b \\ 0 & \text{otherwise.} \end{cases}$$

Notice that as required, when $a, b \in \mathbb{N}$ we have

$$\binom{a}{b}' = \binom{a}{b}.$$

From now on we will just write $\binom{a}{b}$ instead of $\binom{a}{b}'$. We will start with some lemmas giving general properties of the functions Γ and Π.

Lemma A.2. *The gamma function has the following property:*

$$\forall x, y \in \mathbb{R}_{>0},\ d \in \mathbb{R}_{\geq 0} :\ \ y \leq x \implies \frac{\Gamma(y+d)}{\Gamma(y)} \leq \frac{\Gamma(x+d)}{\Gamma(x)}.$$

Proof: Suppose we have a fixed $d \in \mathbb{R}_{\geq 0}$: We consider the following function:

$$f(t) = \frac{\Gamma(t+d)}{\Gamma(t)},$$

over the range $t \in \mathbb{R}_{>0}$. Differentiating we obtain:

$$f'(t) = \frac{\Gamma'(t+d)\cdot\Gamma(t) - \Gamma(t+d)\cdot\Gamma'(t)}{\Gamma(t)^2},$$

and so

$$\begin{aligned} f'(t) \geq 0 \quad &\iff \quad \Gamma'(t+d)\cdot\Gamma(t) - \Gamma(t+d)\cdot\Gamma'(t) \geq 0 \\ &\iff \quad \Gamma'(t+d)\cdot\Gamma(t) \geq \Gamma(t+d)\cdot\Gamma'(t) \\ &\iff \quad \tfrac{\Gamma'(t+d)}{\Gamma(t+d)} \geq \tfrac{\Gamma'(t)}{\Gamma(t)} \qquad\qquad \text{(since } \Gamma(t+d), \Gamma(t) > 0\text{).} \end{aligned}$$

Now in general (see [91]) we have

$$\frac{\Gamma'(z)}{\Gamma(z)} = -\left(\frac{1}{z} + \gamma + \sum_{n=1}^{\infty}\left[\frac{1/n}{1+z/n} - \frac{1}{n}\right]\right),$$

where γ is the Euler-Mascheroni constant (see [93] [39]). We see that $\frac{\Gamma'(z)}{\Gamma(z)}$ increases as z increases, and so because $t + d \geq t$, we have

$$\frac{\Gamma'(t+d)}{\Gamma(t+d)} \geq \frac{\Gamma'(t)}{\Gamma(t)},$$

which means that $f'(t) \geq 0$ for all t. So f is an increasing function, which in particular means that

$$\forall x, y \in \mathbb{R}_{>0},\ d \in \mathbb{R}_{\geq 0} :\ \ y \leq x \implies f(y) \leq f(x) \implies \frac{\Gamma(y+d)}{\Gamma(y)} \leq \frac{\Gamma(x+d)}{\Gamma(x)}.$$

■

It is clear that because $\Pi(x) = \Gamma(x+1)$, and $y \leq x \iff y+1 \leq x+1$, for any $x, y, d \in \mathbb{R}_{\geq 0}$ we also have

$$y \leq x \implies \frac{\Pi(y+d)}{\Pi(y)} \leq \frac{\Pi(x+d)}{\Pi(x)}. \tag{A.1}$$

Lemma A.3. *For any $z \in \mathbb{R}_{\geq 0}$ and $m \in \mathbb{N}$ we have*

$$\frac{\Pi(z+m)}{\Pi(z)} = \prod_{i=1}^{m}(z+i). \tag{A.2}$$

Proof: We have:

$$\begin{aligned}
\Pi\big(z+m\big) &= \big(z+m\big)\cdot\Pi\big(z+m-1\big) \\
&= \big(z+m\big)\cdot\big(z+m-1\big)\cdot\Pi\big(z+m-2\big) \\
&= \big(z+m\big)\cdot\ldots\cdot\big(z+1\big)\cdot\Pi\big(z\big) \\
&= \Big[\textstyle\prod_{i=1}^{m}(z+i)\Big]\cdot\Pi\big(z\big),
\end{aligned}$$

and so the result follows. ■

Proposition A.4. *For any $a, b \in \mathbb{R}$ with $1 \leq b \leq a$ we have*

$$\binom{a}{b} < 2^{a\cdot h(b/a)},$$

where h denotes the binary entropy function:

$$h(x) = -x\cdot\log_2(x) - (1-x)\cdot\log_2(1-x).$$

Proof: We will use *Stirling's formula* (see [51], Chapter 10): for $x \geq 1$ we have

$$\sqrt{2\pi}\cdot x^{x+\frac{1}{2}}\cdot e^{-x} < \Pi(x) < \sqrt{2\pi}\cdot x^{x+\frac{1}{2}}\cdot e^{-x+\frac{1}{12x}}. \tag{A.3}$$

We set $\lambda = \frac{b}{a}$, so that $b = \lambda a$. Since $b \leq a$ we have $\lambda \leq 1$. We also define $\overline{\lambda} = 1-\lambda$. We have:

$$\begin{aligned}
\tbinom{a}{b} &= \tfrac{\Pi(a)}{\Pi(a-b)\cdot\Pi(b)} \\
&< \Big[\sqrt{2\pi}\cdot a^{a+\frac{1}{2}}\cdot e^{-a+\frac{1}{12a}}\Big]\cdot\Big[\sqrt{2\pi}\cdot b^{b+\frac{1}{2}}\cdot e^{-b}\cdot\sqrt{2\pi}\cdot(a-b)^{a-b+\frac{1}{2}}\cdot e^{b-a}\Big]^{-1} \\
&= \tfrac{1}{\sqrt{2\pi}}\cdot\tfrac{a^{1/2}}{b^{1/2}\cdot(a-b)^{1/2}}\cdot\tfrac{a^a}{b^b\cdot(a-b)^{a-b}}\cdot\exp\big(-a+b+(a-b)+\tfrac{1}{12a}\big) \\
&= \tfrac{1}{\sqrt{2\pi}}\cdot\sqrt{\tfrac{a}{\lambda a\cdot(1-\lambda)a}}\cdot\tfrac{a^a}{(\lambda a)^{\lambda a}\cdot((1-\lambda)a)^{(1-\lambda)a}}\cdot\exp\big(\tfrac{1}{12a}\big) \\
&= \tfrac{1}{\sqrt{2\pi\cdot\lambda\overline{\lambda}a}}\cdot\tfrac{1}{\lambda^{\lambda a}\cdot\overline{\lambda}^{\overline{\lambda}a}}\cdot\exp\big(\tfrac{1}{12a}\big) \\
&< \tfrac{1}{\lambda^{\lambda a}\cdot\overline{\lambda}^{\overline{\lambda}a}} \qquad\qquad \text{(since } a \geq 1\text{)}.
\end{aligned}$$

Now we see that

$$\lambda^{\lambda a} \cdot \overline{\lambda}^{\overline{\lambda} a} = 2^{\lambda \cdot a \cdot \log_2(\lambda) + \overline{\lambda} \cdot a \cdot \log_2(\overline{\lambda})} = 2^{a \cdot (\lambda \cdot \log_2(\lambda) + (1-\lambda) \cdot \log_2(1-\lambda))} = 2^{-a \cdot h(\lambda)},$$

and since $\lambda = \frac{b}{a}$, the result follows.

■

Proposition A.5. *For any $a, b, c \in \mathbb{N}$, $\epsilon \in \mathbb{R}_{\geq 0}$. We have:*

$$b + \epsilon \leq a \implies \binom{a}{c}\binom{b}{c} \leq \binom{a-\epsilon}{c}\binom{b+\epsilon}{c}, \tag{A.4}$$

and

$$b + \epsilon \leq a \implies \binom{a}{c}\binom{b}{c+1} \leq \binom{a-\epsilon}{c}\binom{b+\epsilon}{c+1}. \tag{A.5}$$

Proof: We start with the first inequality:
• **1)** Calling LHS and RHS the left and right hand sides of inequation (A.4), we have:

$$\begin{aligned} \frac{\text{RHS}}{\text{LHS}} &= \frac{\pi(a-\epsilon)}{\pi(a-c-\epsilon)\cdot\pi(c)} \cdot \frac{\pi(b+\epsilon)}{\pi(b-c+\epsilon)\cdot\pi(c)} \cdot \left[\frac{\pi(a)}{\pi(a-c)\cdot\pi(c)} \cdot \frac{\pi(b)}{\pi(b-c)\cdot\pi(c)}\right]^{-1} \\ &= \frac{\pi(a-\epsilon)}{\pi(a-\epsilon-c)} \cdot \frac{\pi(b+\epsilon)}{\pi(b+\epsilon-c)} \cdot \frac{\pi(a-c)}{\pi(a)} \cdot \frac{\pi(b-c)}{\pi(b)}. \end{aligned} \tag{A.6}$$

Recall from Lemma A.3 that

$$\forall z \in \mathbb{R}_{\geq 0}, m \in \mathbb{N}: \quad \frac{\Pi(z+m)}{\Pi(z)} = \prod_{i=1}^{m}(z+i). \tag{A.7}$$

Now to each one of the four fractions in (A.6) we can apply (A.7). We set $m = c$, and $z = a - c - \epsilon,\ b - c + \epsilon,\ a - c$ and $b - c$ to obtain

$$\begin{aligned} \frac{\text{RHS}}{\text{LHS}} &= \prod_{i=1}^{c}(a-c+i-\epsilon)\cdot(b-c+i+\epsilon)\cdot\frac{1}{a-c+i}\cdot\frac{1}{b-c+i} \\ &= \prod_{i=1}^{c}\frac{a-c+i-\epsilon}{a-c+i}\cdot\frac{b-c+i+\epsilon}{b-c+i} \\ &= \prod_{i=1}^{c}\frac{A_i-\epsilon}{A_i}\cdot\frac{B_i+\epsilon}{B_i}, \end{aligned}$$

where $A_i = a - c + i$ and $B_i = b - c + i$. Now

$$\begin{aligned} \frac{A-\epsilon}{A}\cdot\frac{B+\epsilon}{B} \geq 1 \quad &\iff \quad (A-\epsilon)\cdot(B+\epsilon) \geq AB \\ &\iff \quad AB - B\epsilon + A\epsilon - \epsilon^2 \geq AB \\ &\iff \quad -B + A - \epsilon \geq 0 \\ &\iff \quad A \geq B + \epsilon. \end{aligned} \tag{A.8}$$

Now since we suppose in the statement that $a \geq b + \epsilon$, for all i we have $A_i \geq B_i + \epsilon$. So since we used equivalences everywhere in (A.8), we obtain

$$\prod_{i=1}^{c} \frac{A_i - \epsilon}{A_i} \cdot \frac{B_i + \epsilon}{B_i} \geq 1,$$

which means that

$$\text{LHS} \leq \text{RHS},$$

and so (A.4) holds.

• **2)** The second inequality can be proved in much the same way. We proceed exactly as in **1)**, the only difference being that in this case we have $B_i = b - c - 1 + i$. So we have an even stronger inequality between the A_i's and $B_i's$: $a \geq b + \epsilon$ implies that for all i: $A_i \geq B_i + \epsilon + 1 > B_i + \epsilon$. So (A.5) also holds. ■

Proposition A.6. *For any $a, b, c \in \mathbb{R}_{\geq 0}$ with $b < a$, we have*

$$\binom{a}{c}\binom{b}{c+1} \leq \binom{a}{c+1/2}\binom{b}{c+1/2}. \tag{A.9}$$

Proof: Calling LHS and RHS the left and right hand sides of the inequation (A.9), we have

$$\begin{aligned}
\tfrac{\text{RHS}}{\text{LHS}} &= \tfrac{\pi(a)}{\pi(a-c-1/2)\cdot\pi(c+1/2)} \cdot \tfrac{\pi(b)}{\pi(b-c-1/2)\cdot\pi(c+1/2)} \cdot \left[\tfrac{\pi(a)}{\pi(a-c)\cdot\pi(c)} \cdot \tfrac{\pi(b)}{\pi(b-c-1)\cdot\pi(c+1)}\right]^{-1} \\
&= \overbrace{\frac{\pi(a-c)}{\pi(a-c-1/2)} \cdot \frac{\pi(b-c-1)}{\pi(b-c-1/2)}}^{X_1} \cdot \overbrace{\frac{\pi(c)}{\pi(c+1/2)} \cdot \frac{\pi(c+1)}{\pi(c+1/2)}}^{X_2}.
\end{aligned}$$

Now we know from (A.1) that for $x, y, d \in \mathbb{R}_{\geq 0}$ we have

$$y \leq x \implies \frac{\Pi(y+d)}{\Pi(y)} \leq \frac{\Pi(x+d)}{\Pi(x)}, \tag{A.10}$$

So setting $y = c$, $x = c + \frac{1}{2}$ and $d = \frac{1}{2}$, we have $y < x$, and therefore

$$\frac{\Pi(c+1/2)}{\Pi(c)} \leq \frac{\Pi(c+1)}{\Pi(c+1/2)}.$$

From this we can deduce that

$$X_2 \geq 1.$$

Likewise, we apply (A.10) with $y = b - c - 1$, $x = a - c - \frac{1}{2}$ and $d = \frac{1}{2}$. Since we are supposing that $b < a$, we have $y < x$, and therefore

$$\frac{\Pi(b-c-1/2)}{\Pi(b-c-1)} \leq \frac{\Pi(a-c)}{\Pi(a-c-1/2)},$$

which leads to

$$X_1 \geq 1.$$

So we have

$$X_1 \cdot X_2 \geq 1 \implies RHS \geq LHS,$$

and so (A.9) holds. ■

We are now ready to prove Proposition 3.13 of Chapter 3, which we restate below:

Proposition 3.13. For any $n, \ell \in \mathbb{N}$, $0 < \delta < \frac{1}{2}$ with $1 \leq \ell \leq 2n\delta$, letting $\overline{\delta} = 1 - \delta$ we have

$$\sum_{d=1}^{\lfloor n\delta \rfloor} \binom{n-d}{\lfloor \ell/2 \rfloor} \binom{d-1}{\lceil \ell/2 \rceil - 1} \leq n\delta \cdot \binom{n\delta}{\ell/2} \binom{n\overline{\delta}}{\ell/2}. \tag{A.11}$$

Proof: We call LHS and RHS the left and right hand sides of (A.11). We proceed differently depending on whether ℓ is even or odd.

• **1)** ℓ is even.
Letting $x = \frac{\ell}{2} = \lfloor \frac{\ell}{2} \rfloor = \lceil \frac{\ell}{2} \rceil$, we have

$$\text{LHS} \;=\; \sum_{d=1}^{\lfloor n\delta \rfloor} \binom{n-d}{x} \binom{d-1}{x-1} \;\leq\; \sum_{d=1}^{\lfloor n\delta \rfloor} \binom{n-d}{x} \binom{d}{x},$$

where the inequality follows from the fact that everything is positive and in general $\binom{a}{b} \leq \binom{a+1}{b+1}$. We let

$$w_{n,x}(d) = \binom{n-d}{x} \binom{d}{x}.$$

We will show that for any n, x and for any $d = 1, \ldots, \lfloor n\delta \rfloor - 1$ we have $w_{n,x}(d) \leq w_{n,x}(d+1)$:

$$\begin{aligned}
\tfrac{w_{n,x}(d+1)}{w_{n,x}(d)} &= \left[\tbinom{n-d-1}{x}\tbinom{d+1}{x}\right] \cdot \left[\tbinom{n-d}{x}\tbinom{d}{x}\right]^{-1} \\
&= \tfrac{(n-d-1)!}{(n-d-x-1)!x!} \cdot \tfrac{(d+1)!}{(d+1-x)!x)!} \quad \cdot \quad \tfrac{(n-d-x)!x!}{(n-d)!} \cdot \tfrac{(d-x)!x!}{d!} \\
&= \tfrac{d+1}{d+1-x} \cdot \tfrac{n-d-x}{n-d} \\
&= \Big[\overbrace{1 - \frac{x}{d+1}}^{X_1}\Big]^{-1} \cdot \Big[\overbrace{1 - \frac{x}{n-d}}^{X_2}\Big].
\end{aligned}$$

This means that

$$\begin{aligned}
w_{n,x}(d) \leq w_{n,x}(d+1) &\iff 1 \leq X_1^{-1} \cdot X_2 \\
&\iff X_1 \leq X_2 \\
&\iff 1 - \tfrac{x}{d+1} \leq 1 - \tfrac{x}{n-d} \qquad (A.12) \\
&\iff \tfrac{d+1}{x} \leq \tfrac{n-d}{x} \\
&\iff d + 1 \leq n - d.
\end{aligned}$$

Now for any $d = 1, \ldots, \lfloor n\delta \rfloor - 1$, we have $d + 1 \leq \lfloor n\delta \rfloor < \frac{n}{2}$ (since $\delta < \frac{1}{2}$). So

$$d + 1 < \frac{n}{2} \implies 2d + 2 < n \implies d + 1 < n - d - 1 < n - d.$$

Combining this with (A.12) we deduce that

$$\forall\, d = 1, \ldots, \lfloor n\delta \rfloor - 1 : \quad w_{n,x}(d) \leq w_{n,x}(d + 1).$$

So this leads to

$$\forall\, d = 1, \ldots, \lfloor n\delta \rfloor : \quad w_{n,x}(d) \leq w_{n,x}(\lfloor n\delta \rfloor).$$

We can now deduce:

$$\text{LHS} \;=\; \sum_{d=1}^{\lfloor n\delta \rfloor} w_{n,x}(d) \;\leq\; \lfloor n\delta \rfloor \cdot w_{n,x}(\lfloor n\delta \rfloor) \;=\; \lfloor n\delta \rfloor \cdot \binom{n - \lfloor n\delta \rfloor}{x} \binom{\lfloor n\delta \rfloor}{x}. \tag{A.13}$$

Now letting $\epsilon = n\delta - \lfloor n\delta \rfloor$, we notice that $n - \lfloor n\delta \rfloor - \epsilon = n - \lfloor n\delta \rfloor - (n\delta - \lfloor n\delta \rfloor) = n - n\delta = n\overline{\delta}$. So we have

$$\begin{aligned} n - \lfloor n\delta \rfloor - \epsilon &= n\overline{\delta} \\ \lfloor n\delta \rfloor + \epsilon &= n\delta. \end{aligned} \tag{A.14}$$

Now recall from (A.4) of Proposition A.5 that for any $a, b, c \in \mathbb{N}$ with $b + \epsilon \leq a$ we have

$$\binom{a}{c} \binom{b}{c} \leq \binom{a - \epsilon}{c} \binom{b + \epsilon}{c}. \tag{A.15}$$

We notice that because $\delta < \frac{1}{2}$ we have $\lfloor n\delta \rfloor < n/2$, which means that $\lfloor n\delta \rfloor < n - \lfloor n\delta \rfloor$. So setting $a = n - \lfloor n\delta \rfloor$, $b = \lfloor n\delta \rfloor$ and $c = x$ we see first of all that $b < a$. So since a and b are integers and $0 \leq \epsilon < 1$ we have $b + \epsilon < a$ and we therefore can apply (A.15) to obtain

$$\binom{n - \lfloor n\delta \rfloor}{x} \binom{\lfloor n\delta \rfloor}{x} \leq \binom{n - \lfloor n\delta \rfloor - \epsilon}{x} \binom{\lfloor n\delta \rfloor + \epsilon}{x} = \binom{n\overline{\delta}}{\ell/2} \binom{n\delta}{\ell/2}, \tag{A.16}$$

where the last equality follows from (A.14) and the fact that x was defined as $x = \frac{\ell}{2}$. Now we combine (A.13) and (A.16) to deduce

$$LHS \leq \lfloor n\delta \rfloor \cdot \binom{n - \lfloor n\delta \rfloor}{x} \binom{\lfloor n\delta \rfloor}{x} \leq n\delta \cdot \binom{n\overline{\delta}}{\ell/2} \binom{n\delta}{\ell/2} = RHS.$$

• **2)** ℓ is odd.
We let $x = \frac{\ell - 1}{2}$. So $\lfloor \ell/2 \rfloor = x$ and $\lceil \ell/2 \rceil - 1 = (x + 1) - 1 = x$. This leads to

$$\text{LHS} \;=\; \sum_{d=1}^{\lfloor n\delta \rfloor} \binom{n - d}{x} \binom{d - 1}{x} \leq \sum_{d=1}^{\lfloor n\delta \rfloor} \binom{n - d}{x} \binom{d}{x + 1},$$

where the second inequality follows from the fact that in general $\binom{a}{b} \leq \binom{a+1}{b+1}$. As above, letting

$$w_{n,x}(d) = \binom{n - d}{x} \binom{d}{x + 1},$$

we will show that for any n, x and for any $d = 1, \ldots, \lfloor n\delta \rfloor - 1$ we have $w_{n,x}(d) \leq w_{n,x}(d+1)$:

$$\begin{aligned}
\tfrac{w_{n,x}(d+1)}{w_{n,x}(d)} &= \left[\tbinom{n-d-1}{x}\tbinom{d+1}{x+1}\right] \cdot \left[\tbinom{n-d}{x}\tbinom{d}{x+1}\right]^{-1} \\
&= \tfrac{(n-d-1)!}{(n-d-x-1)!x!} \cdot \tfrac{(d+1)!}{(d-x)!(x+1)!} \quad \cdot \quad \tfrac{(n-d-x)!x!}{(n-d)!} \cdot \tfrac{(d-x-1)!(x+1)!}{d!} \cdot \\
&= \tfrac{d+1}{d-x} \cdot \tfrac{n-d-x}{n-d} \\
&= \left[\tfrac{d+1-(x+1)}{d+1}\right]^{-1} \cdot \left[\tfrac{n-d-x}{n-d}\right] \\
&= \left[\overbrace{1 - \frac{x+1}{d+1}}^{Y_1}\right]^{-1} \cdot \left[\overbrace{1 - \frac{x}{n-d}}^{Y_2}\right].
\end{aligned}$$

We saw above in (A.12) that because $d + 1 \leq n - d$, for any $x \geq 0$ we have

$$1 - \frac{x}{d+1} \leq 1 - \frac{x}{n-d}.$$

So here

$$Y_1 = 1 - \frac{x+1}{d+1} < 1 - \frac{x}{d+1} \leq 1 - \frac{x}{n-d} = Y_2,$$

and therefore $Y_1^{-1} \cdot Y_2 \geq 1$. From this we deduce that

$$\forall\ d = 1, \ldots, \lfloor n\delta \rfloor - 1: \quad w_{n,x}(d) \leq w_{n,x}(d+1),$$

and therefore

$$\forall\ d = 1, \ldots, \lfloor n\delta \rfloor: \quad w_{n,x}(d) \leq w_{n,x}(\lfloor n\delta \rfloor).$$

As above, this leads to

$$\text{LHS} \;=\; \sum_{d=1}^{\lfloor n\delta \rfloor} w_{n,x}(d) \;\leq\; \lfloor n\delta \rfloor \cdot w_{n,x}(\lfloor n\delta \rfloor) \;=\; \lfloor n\delta \rfloor \cdot \binom{n - \lfloor n\delta \rfloor}{x}\binom{\lfloor n\delta \rfloor}{x+1}. \tag{A.17}$$

In exactly the same way as case **1)** above, setting $\epsilon = n\delta - \lfloor n\delta \rfloor$ gives us

$$\begin{aligned}
n - \lfloor n\delta \rfloor - \epsilon &= n\overline{\delta} \\
\lfloor n\delta \rfloor + \epsilon &= n\delta.
\end{aligned} \tag{A.18}$$

Recall from (A.5) of Proposition A.5 that for any $a, b, c \in \mathbb{N}$ with $b + \epsilon \leq a$ we have

$$\binom{a}{c}\binom{b}{c+1} \leq \binom{a-\epsilon}{c}\binom{b+\epsilon}{c+1}. \tag{A.19}$$

If we set $a = n - \lfloor n\delta \rfloor$, $b = \lfloor n\delta \rfloor$ and $c = x$ then as above we have $b + \epsilon \leq a$, and we can therefore apply (A.19) to obtain

$$\binom{n - \lfloor n\delta \rfloor}{x}\binom{\lfloor n\delta \rfloor}{x+1} \leq \binom{n\overline{\delta}}{x}\binom{n\delta}{x+1}. \tag{A.20}$$

Finally, we saw in Proposition A.6 that for any $a, b, c \in \mathbb{R}_{\geq 0}$ with $b < a$, we have

$$\binom{a}{c}\binom{b}{c+1} \leq \binom{a}{c+1/2}\binom{b}{c+1/2}, \tag{A.21}$$

and so applying this with $a = n\overline{\delta}$, $b = n\delta$ and $c = x$ we have $b < a$ and therefore

$$\binom{n\overline{\delta}}{x}\binom{n\delta}{x+1} \leq \binom{n\overline{\delta}}{x+1/2}\binom{n\delta}{x+1/2} = \binom{n\overline{\delta}}{\ell/2}\binom{n\delta}{\ell/2}, \tag{A.22}$$

where the second equality follows from the fact that x was defined as $x = \frac{\ell-1}{2}$. Combining (A.17), (A.20) and (A.22) gives us

$$LHS \leq \lfloor n\delta \rfloor \cdot \binom{n - \lfloor n\delta \rfloor}{x}\binom{\lfloor n\delta \rfloor}{x+1} \leq n\delta \cdot \binom{n\overline{\delta}}{\ell/2}\binom{n\delta}{\ell/2} = RHS.$$

■

Appendix B

Proof of Theorem 6.11

The proof of Theorem 6.11 was outlined in section 6.4.4, In this appendix we give proofs of the results that were stated in that section namely we will prove (6.53) and (6.54) (Lemma B.6), and the three inequalities in (6.55) (Lemmas B.7, B.8 and B.9).

Although there is a certain intuition behind these (similar to that in the section on derandomized squaring), it is not easy to convince oneself of the truth of the propositions by intuitive reasoning. We therefore make our proofs more technical and rigorous, even though this does make things more tedious to follow.

We will need some basic results on tensoring, inner products and the graphs we defined in section 6.4.3. Although with a little thought we can convince ourselves that they hold, formal proofs are technical. We therefore lay these results out in the following lemmas for reference. The proofs are included in Appendix C.

Lemma B.1.

1. $\forall \sigma \in \mathbb{R}^{nm}, \tau \in \mathbb{R}^n : \langle \sigma, \tau \otimes 1_m \rangle = \langle \mathcal{M}_n(\sigma), \tau \rangle$.
2. $\forall \sigma \in \mathbb{R}^{nd(1)}, \tau \in \mathbb{R}^n : \langle \sigma, \tau \otimes e_1 \rangle = \langle \mathcal{M}_n(\sigma), \tau \rangle$.
3. $\forall \sigma \in \mathbb{R}^{nd(2)}, \tau \in \mathbb{R}^n : \langle \sigma, \tau \otimes e_2 \rangle = \langle \mathcal{M}_n(\sigma), \tau \rangle$.

Lemma B.2.

1. $\forall \sigma \in \mathbb{R}^n : \mathcal{M}_n\big(\overline{A}(\sigma \otimes e_1)\big) = d_1 \cdot A\sigma$.
2. $\forall \sigma \in \mathbb{R}^m : \mathcal{M}_m\big(\overline{B}(\sigma \otimes e_2)\big) = d_2 \cdot B\sigma$.
3. $\forall \sigma \in \mathbb{R}^n, \forall \tau \in \mathbb{R}^n : \langle \overline{A}(\sigma \otimes e_1), \tau \otimes e_1 \rangle = d_1 \cdot \langle A\sigma, \tau \rangle$.
4. $\forall \sigma \in \mathbb{R}^m, \forall \tau \in \mathbb{R}^m : \langle \overline{B}(\sigma \otimes e_2), \tau \otimes e_2 \rangle = d_2 \cdot \langle B\sigma, \tau \rangle$.

Lemma B.3.

1. $C \cdot e_1 = \frac{d_1}{d_2} \cdot e_2$.
2. $C \cdot e_2 = \frac{d_2}{d_1} \cdot e_1$.
3. $\forall \sigma \in \mathbb{R}^{nm} : \overline{A}(\sigma \otimes 1_d) = \overline{A}(\sigma \otimes e_1)$.
4. $\forall \sigma \in \mathbb{R}^{nm} : \overline{B}(\sigma \otimes 1_d) = \overline{B}(\sigma \otimes e_2)$.
5. $\forall \sigma \in \mathbb{R}^n, \tau \in \mathbb{R}^{md} : \mathcal{M}_{nm}(\sigma \otimes \tau) = \sigma \otimes \mathcal{M}_m(\tau)$.

To obtain the required results we will first need to prove two Lemmas (B.4 and B.5). As stated in Theorem 6.11, our bound on the second eigenvalue of a derandomized tensor product required that the labellings of the graphs A and B be half-colorings. The next two lemmas are the only places we will use this.

Recall that we interpret the graph $\overline{B}$ (see Definition B.5 as consisting of m copies of the vertices of C (m clouds), namely one for each vertex of B. We refer to the vertices in $[m] \times D_1$ as the *left vertices* of B, and to those in $[m] \times D_2$ as the *right vertices*. A vector $\tau \in R^{md}$ is said to be B-uniform if

$$\mathcal{M}_m(\tau) \in 1_m^{\parallel}. \tag{B.1}$$

So if τ is a distribution over the right vertices of $\overline{B}$, it is B-uniform if and only if the marginal over the clouds is uniform (the probability of being on any given cloud is the same, namely $\frac{1}{m}$).

The intuition behind Lemma B.4 is the following: Suppose we start with a B-uniform distribution on the vertices of $\overline{B}$ of the form $1_m \otimes \sigma$ (so the distribution inside each cloud is the same). Then after one step of a walk in $\overline{B}$ it will still be B-uniform.

Lemma B.4. *If the labeling of B is a half-coloring then for any $\sigma \in R_2$,*

$$\mathcal{M}_m\big(\overline{B}(1_m \otimes \sigma)\big) = 1_m \cdot \langle \sigma, 1_d \rangle.$$

Proof: Recall from Definition B.5 that $\overline{B}$ is a graph with vertex set $[m] \times [d]$ in which each vertex (i, j) has either one or no neighbors:

- If $j \in D_1$ then (i, j) has no neighbors.
- If $j \in D_2$ then there is an edge from (i, j) to $(i[j], \rho_B(j))$.

One step of a random walk on $\overline{B}$ can be seen as an involution on the set $[m]$ of vertices (they all have degrees either 1 or 0). Multiplying a vector $\tau \in \mathbb{R}^{md}$ by $\overline{B}$ involves permuting its components. Formally, if we index the entries of τ with the set $[m] \times [d]$ then

$$(\overline{B}\tau)_{ij} = \tau_{i[j],\rho(j)}. \tag{B.2}$$

Recall that a half-coloring means that each color $j \in D_2$ has a "partner color" $p(j) \in D_2$ for which any vertex $i \in [m]$ satisfies

$$i[j][\rho(j)] = i[\rho(j)][j] = i. \tag{B.3}$$

We want to study $\overline{B}(1_m \otimes \sigma)$. Let $w = 1_m \otimes \sigma$. If we index the entries of w with the set $[m] \times D_2$ in the natural way then we have

$$w_{i,j} = \sigma_j. \tag{B.4}$$

So using (B.2) gives us

$$(\overline{B}w)_{i,j} = w_{i[j],\rho(j)} = \sigma_{\rho(j)}. \tag{B.5}$$

So for all $i \in [m]$ the vector $(\overline{B}w)_i \in R_2$ is just a permutation of σ. We have

$$\big(\mathcal{M}_m(\overline{B}w)\big)_i = \sum_{j=1}^{d} (\overline{B}w)_{i,j} = \sum_{j=1}^{d} \sigma_{\rho(j)} = \sum_{j=1}^{d} \sigma_j = \langle \sigma, e_2 \rangle.$$

Since this holds for all m entries of $\mathcal{M}_m(\overline{B}w)$, we have

$$\mathcal{M}_m(\overline{B}w) = 1_m \cdot \langle \sigma, e_2 \rangle. \tag{B.6}$$

■

A vector $\tau \in \mathbb{R}^{md}$ is said to be *B-anti-uniform* if the marginal over each C-vertex is anti-uniform. Another way of phrasing this is that if we decompose τ as

$$\tau = \begin{pmatrix} \tau_1 \\ \vdots \\ \tau_m \end{pmatrix}, \tag{B.7}$$

where $\tau_i \in \mathbb{R}^d$, then

$$\sum_{i=1}^{m} \tau_i = 0_d. \tag{B.8}$$

The intuition behind Lemma B.5 is the following: Suppose we start with a B-anti-uniform distribution on the vertices of $\overline{B}$ of the form $v \otimes 1_d$, with $v \in 1_m^{\perp}$ (so a C-uniform distribution). Then after one step of a walk in $\overline{B}$ it will still be B-anti-uniform.

Lemma B.5. *Suppose the labeling of B is a half-coloring. Let $v \in 1_m^{\perp}$, and decompose $b = \overline{B}(v \otimes 1_d) \in \mathbb{R}^{md}$ as follows:*

$$b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}, \tag{B.9}$$

where $b_i \in \mathbb{R}^d$. Then

$$\sum_{i=1}^{m} b_i = 0_d. \tag{B.10}$$

Proof: Recall that a half-coloring means that each color $j \in D_2$ has a "partner color" $\rho(j) \in D_2$ for which every vertex $i \in [m]$ satisfies

$$i[j][\rho(j)] = i[\rho(j)][j] = i. \tag{B.11}$$

We will first show that for a fixed $j \in D_2$, the mapping $\chi_j : [m] \to [m]$ defined as

$$\chi_j(i) = i[j]$$

is a bijection. For $i, \ell \in [m]$ we have

$$\begin{aligned} \chi_j(i) = \chi_j(\ell) \quad &\Longrightarrow \quad i[j] = \ell[j] \\ &\Longrightarrow \quad i[j][\rho(j)] = \ell[j][\rho(j)] \\ &\Longrightarrow \quad i = \ell \qquad\qquad \text{(by the definition of a half-coloring)}. \end{aligned} \tag{B.12}$$

So χ_j is injective and therefore (by cardinality arguments) bijective. So χ_j is a permutation of $[m]$.

Now let $w = v \otimes 1_n \in \mathbb{R}^{md}$. If we index the entries of w with the set $[m] \times [d]$ then

$$w_{i,j} = v_i. \tag{B.13}$$

Now using (B.2) we have:

$$(b_i)_j = (\overline{B}w)_{i,j} = w_{i[j],\rho(j)} = v_{i[j]}. \tag{B.14}$$

So for each $j \in [m]$:

$$\left(\sum_{i=1}^{m} b_i\right)_j = \sum_{i=1}^{m} (b_i)_j = \sum_{i=1}^{m} v_{i[j]} = \sum_{i=1}^{m} v_{\chi_j(i)}. \tag{B.15}$$

Since χ_j is a permutation of $[m]$, we can deduce that

$$\left(\sum_{i=1}^{m} b_i\right)_j = \sum_{i=1}^{m} v_i = 0, \tag{B.16}$$

where the last equality follows from the fact that $v \in 1_m^{\perp}$. So as required,

$$\sum_{i=1}^{m} b_i = 0_d. \tag{B.17}$$

■

We are now ready to prove the results we stated in our outline of the proof of Theorem 6.11. We start by giving a reminder of the definitions of the subspaces S_1, S_2 and S_3 of $\mathbb{R}^{nmd}$:

$$\begin{aligned} S_1 &= 1_n^{\perp} \otimes 1_m^{\parallel} \otimes 1_d^{\parallel}. \\ S_2 &= 1_n^{\parallel} \otimes 1_m^{\perp} \otimes 1_d^{\parallel}. \\ S_3 &= 1_n^{\perp} \otimes 1_m^{\perp} \otimes 1_d^{\parallel}. \end{aligned} \tag{B.18}$$

Lemma B.6. *Let S_1, S_2 and S_3 be the subspaces defined in (B.18). If $x_1 \in S_1$, $x_2 \in S_2$, $x_3 \in S_3$ and $x = x_1 + x_2 + x_3$ then*

$$\langle \hat{A}\hat{C}\hat{B}x, x\rangle = \langle \hat{A}\hat{C}\hat{B}x_1, x_1\rangle + \langle \hat{A}\hat{C}\hat{B}x_2, x_2\rangle + \langle \hat{A}\hat{C}\hat{B}x_3, x_3\rangle. \tag{B.19}$$

$$\langle \hat{B}\hat{C}\hat{A}x, x\rangle = \langle \hat{B}\hat{C}\hat{A}x_1, x_1\rangle + \langle \hat{B}\hat{C}\hat{A}x_2, x_2\rangle + \langle \hat{B}\hat{C}\hat{A}x_3, x_3\rangle. \tag{B.20}$$

Proof: We will show only the first part, from which the second part will follow by symmetry. We have

$$\langle \hat{A}\hat{C}\hat{B}x, x\rangle = \langle \hat{A}\hat{C}\hat{B}(x_1+x_2+x_3), (x_1+x_2+x_3)\rangle, \tag{B.21}$$

and therefore expanding this leads to

$$\langle \hat{A}\hat{C}\hat{B}x, x\rangle = \sum_{i=1}^{3}\sum_{j=1}^{3}\langle \hat{A}\hat{C}\hat{B}x_i, x_j\rangle. \tag{B.22}$$

We will show that of the nine terms in (B.22), all but the three that appear in (B.20) are zero. As explained above, the intuition is that the images of the spaces S_1, S_2 and S_3 under the linear transformation defined by $\hat{A}\hat{C}\hat{B}$ are also pairwise orthogonal.

By the definitions of S_1, S_2 and S_3, there are $w \in 1_n^{\perp}$, $y \in 1_m^{\perp}$ with

$$x_1 = w \otimes 1_m \otimes 1_d, \qquad x_2 = 1_n \otimes y \otimes 1_d, \tag{B.23}$$

and there are $u_1, \ldots, u_k \in 1_n^{\perp}$, $v_1, \ldots, v_k \in 1_m^{\perp}$ with

$$x_3 = \sum_{i=1}^{k} u_i \otimes v_i \otimes 1_d. \tag{B.24}$$

Claim 1: For any $\sigma \in 1_n^{\perp}$, $\tau \in \mathbb{R}^m$, we have

$$\langle \hat{A}\hat{C}\hat{B}x_2, \sigma \otimes \tau \otimes 1_d\rangle = 0. \tag{B.25}$$

Proof: Recall that $\hat{B} = I_n \otimes \overline{B}$ (see Section 6.4.3).

$$\hat{B}x_2 = \hat{B}(1_n \otimes y \otimes 1_d) = (I_n \otimes \overline{B})(1_n \otimes y \otimes 1_d) = 1_n \otimes \overline{B}(y \otimes 1_d) = 1_n \otimes b, \tag{B.26}$$

where $b = \overline{B}(y \otimes 1_d)$. We can decompose $b \in \mathbb{R}^{md}$ as follows:

$$b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}, \tag{B.27}$$

where $b_1, \ldots, b_m \in \mathbb{R}^d$. Now recalling that $\hat{C} = I_n \otimes I_m \otimes C$, we obtain

$$\hat{C}\hat{B}x_2 = (I_n \otimes I_m \otimes C) \cdot \left(1_n \otimes \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}\right) = 1_n \otimes \begin{pmatrix} Cb_1 \\ \vdots \\ Cb_m \end{pmatrix} = 1_n \otimes \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix}, \tag{B.28}$$

where $\forall j \in [m] : \;\; t_j = Cb_j \in \mathbb{R}^d$. Let $t \in \mathbb{R}^{md}$ be defined as

$$t = \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix}. \tag{B.29}$$

We now need to change bases (purely for notational purposes). We make the basis change given by the matrix P from Definition 6.23. This gives us

$$\hat{C}\hat{B}x_2 = 1_n \otimes t \sim_P \begin{pmatrix} 1_n \otimes t_1 \\ \vdots \\ 1_n \otimes t_m \end{pmatrix}. \tag{B.30}$$

In this new basis we have $\hat{A} \sim_P 1_m \otimes \overline{A}$, so that

$$\hat{A}\hat{C}\hat{B}x_2 \sim_P \hat{A}(1_n \otimes t) = \begin{pmatrix} \overline{A}(1_n \otimes t_1) \\ \vdots \\ \overline{A}(1_n \otimes t_m) \end{pmatrix}. \tag{B.31}$$

Again in our new basis, $\sigma \otimes \tau \otimes 1_d$ becomes

$$\sigma \otimes \tau \otimes 1_d \sim_P \tau \otimes \begin{pmatrix} \sigma \otimes 1_d \\ \vdots \\ \sigma \otimes 1_d \end{pmatrix} = \begin{pmatrix} \tau_1(\sigma \otimes 1_d) \\ \vdots \\ \tau_m(\sigma \otimes 1_d) \end{pmatrix}. \tag{B.32}$$

Recall that we need to compute

$$\langle \hat{A}\hat{C}\hat{B}x_2, \sigma \otimes \tau \otimes 1_d \rangle, \tag{B.33}$$

which according to (B.31) and (B.32) is equal to

$$\left\langle \begin{pmatrix} \overline{A}(1_n \otimes t_1) \\ \vdots \\ \overline{A}(1_n \otimes t_m) \end{pmatrix}, \begin{pmatrix} \tau_1(\sigma \otimes 1_d) \\ \vdots \\ \tau_m(\sigma \otimes 1_d) \end{pmatrix} \right\rangle. \tag{B.34}$$

We can express this as

$$\begin{aligned} \langle \hat{A}\hat{C}\hat{B}x_2, \sigma \otimes \tau \otimes 1_d \rangle &= \textstyle\sum_{j=1}^m \langle \overline{A}(1_n \otimes t_j), \tau_j(\sigma \otimes 1_d) \rangle \\ &= \textstyle\sum_{j=1}^m \tau_j \cdot \langle \overline{A}(1_n \otimes t_j), \sigma \otimes e_1 \rangle \\ &= \textstyle\sum_{j=1}^m \tau_j \cdot d \cdot \langle \mathcal{M}_n\big(\overline{A}(1_n \otimes t_j)\big), \sigma \rangle \quad \text{(Lemma B.1 (2))} \\ &= \textstyle\sum_{j=1}^m \tau_j \cdot d \cdot \langle t_j, 1_d \rangle \cdot \overbrace{\langle 1_n, \sigma \rangle}^{0} \quad \text{(Lemma B.4)} \\ &= 0 \quad \text{(since } \sigma \in 1_n^{\perp}\text{).} \end{aligned} \tag{B.35}$$

□

The next two claims are corollaries of Claim 1:

Claim 2: $\langle \hat{A}\hat{C}\hat{B}x_2, x_1 \rangle = 0$.
Proof: Recall that

$$x_1 = w \otimes 1_m \otimes 1_d, \tag{B.36}$$

where $w \in 1_n^{\perp}$. So the result follows by setting $\sigma = w$ and $\tau = 1_m$ in Claim 1. □

Claim 3: $\langle \hat{A}\hat{C}\hat{B}x_2, x_3\rangle = 0$.
Proof: Recall that

$$x_3 = \sum_{i=1}^{k} u_i \otimes v_i \otimes 1_d. \tag{B.37}$$

For each $i = 1, \ldots, k$, setting $\sigma = u_i \in 1_n^{\perp}$ and $\tau = v_i \in 1_m^{\perp}$ in Claim 1 enables us to obtain

$$\langle \hat{A}\hat{C}\hat{B}x_2, u_i \otimes v_i \otimes 1_d\rangle = 0, \tag{B.38}$$

which means that

$$\begin{aligned} \left\langle \hat{A}\hat{C}\hat{B}x_2, x_3 \right\rangle &= \left\langle \hat{A}\hat{C}\hat{B}x_2, \textstyle\sum_i u_i \otimes v_i \otimes 1_d \right\rangle \\ &= \textstyle\sum_i \left\langle \hat{A}\hat{C}\hat{B}x_2, u_i \otimes v_j \otimes 1_d \right\rangle \\ &= 0. \end{aligned} \tag{B.39}$$

□

Claim 4: $\langle \hat{A}\hat{C}\hat{B}x_3, x_1\rangle = 0$.
Proof: Recall that

$$x_1 = w \otimes 1_m \otimes 1_d, \qquad\qquad x_3 = \sum_{i=1}^{k} \overbrace{u_i \otimes v_i \otimes 1_d}^{z_i}. \tag{B.40}$$

We will show that for each i, $\langle \hat{C}\hat{B}z_i, \hat{A}x_1\rangle$ is zero. Since $\hat{B} = I_n \otimes \overline{B}$, we have

$$\hat{B}z_i = \hat{B}(u_i \otimes v_i \otimes 1_d) = u_i \otimes \overline{B}(v_i \otimes 1_d) = u_i \otimes b, \tag{B.41}$$

where $b = \overline{B}(v_i \otimes 1_d)$. We can decompose $b \in \mathbb{R}^{md}$ as follows:

$$b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}, \tag{B.42}$$

where $b_1, \ldots, b_m \in \mathbb{R}^d$. Now recalling that $\hat{C} = I_n \otimes I_m \otimes C$, we obtain

$$\hat{C}\hat{B}z_i = (I_n \otimes I_m \otimes C) \cdot \left(u_i \otimes \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} \right) = u_i \otimes \begin{pmatrix} Cb_1 \\ \vdots \\ Cb_m \end{pmatrix} = u_i \otimes \begin{pmatrix} t_1 \\ \vdots \\ t_m \end{pmatrix}, \tag{B.43}$$

where $t_j = Cb_j$. Changing the basis, (B.43) becomes

$$\hat{C}\hat{B}z_i \sim_P \begin{pmatrix} u_i \otimes t_1 \\ \vdots \\ u_i \otimes t_m \end{pmatrix}. \tag{B.44}$$

Since in this new basis $\hat{A} \sim_P I_m \otimes \overline{A}$ and $x_1 \sim_P 1_m \otimes w \otimes 1_d$, we can deduce that

$$\hat{A}x_1 \sim_P 1_m \otimes \overline{A}(w \otimes 1_d). \tag{B.45}$$

So combining (B.44) and (B.45) leads to

$$\langle \hat{C}\hat{B}z_i, \hat{A}x_1 \rangle = \left\langle \begin{pmatrix} u_i \otimes t_1 \\ \vdots \\ u_i \otimes t_m \end{pmatrix}, \begin{pmatrix} \overline{A}(w \otimes 1_d) \\ \vdots \\ \overline{A}(w \otimes 1_d) \end{pmatrix} \right\rangle. \tag{B.46}$$

This can be written as

$$\langle \hat{C}\hat{B}z_i, \hat{A}x_1 \rangle = \sum_{j=1}^{m} \left\langle u_i \otimes t_j, \overline{A}(w \otimes 1_d) \right\rangle \;\; = \;\; \left\langle u_i \otimes \sum_{j=1}^{m} t_j, \overline{A}(w \otimes 1_d) \right\rangle. \tag{B.47}$$

We know from Lemma B.5 that

$$\sum_{j=1}^{m} b_j = 0_d. \tag{B.48}$$

So

$$\sum_{j=1}^{m} t_j = \sum_{j=1}^{m} C b_j = C \cdot \overbrace{\sum_{j=1}^{m} b_j}^{0_d} = 0_d. \tag{B.49}$$

Plugging in (B.49) we see that (B.47) is equal to zero, and therefore

$$\langle \hat{C}\hat{B}z_i, \hat{A}x_1 \rangle = 0. \tag{B.50}$$

Since $\hat{A}$ is symmetric, we have

$$\left\langle \hat{A}\hat{C}\hat{B}x_3, x_1 \right\rangle = \left\langle \hat{C}\hat{B}x_3, \hat{A}x_1 \right\rangle = \left\langle \hat{C}\hat{B} \sum_{i=1}^{k} z_i, \hat{A}x_1 \right\rangle = \sum_{i=1}^{k} \overbrace{\left\langle \hat{C}\hat{B}z_i, \hat{A}x_1 \right\rangle}^{0} = 0. \tag{B.51}$$

□

Claim 5: $\langle \hat{A}\hat{C}\hat{B}x_1, x_2 \rangle = 0$.
Proof: Recall that

$$x_1 = w \otimes 1_m \otimes 1_d, \qquad\qquad x_2 = 1_n \otimes y \otimes 1_d, \tag{B.52}$$

with $w \in 1_n^{\perp}$ and $y \in 1_m^{\perp}$. Since $\overline{B}$ is a permutation on elements of $\mathbb{R}^m \times R_2$, it is clear that it fixes uniform vectors, in particular:

$$\overline{B}(1_m \otimes e_2) = 1_m \otimes e_2. \tag{B.53}$$

This leads to

$$\begin{aligned} \hat{B}x_1 &= (I_n \otimes \overline{B})(w \otimes 1_m \otimes 1_d) \\ &= w \otimes \overline{B}(1_m \otimes e_2) \\ &= w \otimes 1_m \otimes e_2 \qquad\qquad \text{(using (B.53)).} \end{aligned} \tag{B.54}$$

In the same way (though again a basis change is required to accommodate the limits of the current notation) it can be shown that

$$\hat{A}x_2 = 1_n \otimes v \otimes e_1. \tag{B.55}$$

Now (B.54) leads to

$$\begin{aligned} \hat{C}\hat{B}x_1 &= (I_{nm} \otimes C)\big(w \otimes 1_m \otimes e_2\big) \\ &= w \otimes 1_m \otimes C \cdot e_2 \\ &= w \otimes 1_m \otimes \tfrac{d_2}{d_1} \cdot e_1 \qquad \text{(from Lemma B.3 (1))}. \end{aligned} \tag{B.56}$$

So combining (B.55) and (B.56) gives us

$$\begin{aligned} \big\langle \hat{A}\hat{C}\hat{B}x_1, x_2 \big\rangle &= \big\langle \hat{C}\hat{B}x_1, \hat{A}x_2 \big\rangle && \text{(since } \hat{A} \text{ is symmetric)} \\ &= \tfrac{d_2}{d_1} \cdot \big\langle 1_n \otimes v \otimes e_1, w \otimes 1_m \otimes e_1 \big\rangle && \text{(using (B.55) and (B.56))} \\ &= \tfrac{d_2}{d_1} \cdot \overbrace{\big\langle 1_n, w \big\rangle}^{0} \cdot \langle v \otimes e_1, 1_m \otimes e_1 \rangle \\ &= 0 && \text{(since } w \in 1_n^{\perp}\text{)}. \end{aligned}$$

□

Claim 6: $\langle \hat{A}\hat{C}\hat{B}x_1, x_3 \rangle = 0$.
Proof: Recall that

$$x_1 = w \otimes 1_m \otimes 1_d, \qquad x_3 = \sum_{i=1}^{k} \overbrace{u_i \otimes v_i \otimes 1_d}^{z_i}, \tag{B.57}$$

with $w \in 1_n^{\perp}, \forall i = 1, \ldots k : u_i \in 1_n^{\perp}$ and $v_i \in 1_m^{\perp}$. We know from (B.56) that

$$\hat{C}\hat{B}x_1 = \frac{d_2}{d_1} \cdot \big(w \otimes 1_m \otimes e_1\big). \tag{B.58}$$

Now by changing basis we have $\hat{A} \sim_P I_m \otimes \overline{A}$, which means that

$$\hat{A}\hat{C}\hat{B}x_1 \sim_P \frac{d_2}{d_1} \cdot \big(1_m \otimes \overline{A}(w \otimes e_1)\big). \tag{B.59}$$

Since for all $i \in [k] :\ z_i \sim_P v_i \otimes u_i \otimes 1_d$, we obtain

$$\begin{aligned} \big\langle \hat{A}\hat{C}\hat{B}x_1, z_i \big\rangle &= \tfrac{d_2}{d_1} \cdot \big\langle 1_m \otimes \overline{A}(w \otimes e_1), v_i \otimes u_i \otimes 1_d \big\rangle && \text{(from (B.59))} \\ &= \tfrac{d_2}{d_1} \cdot \overbrace{\big\langle 1_m, v_i \big\rangle}^{0} \cdot \big\langle \overline{A}(w \otimes 1_d), u_i \otimes 1_d \big\rangle \\ &= 0 && \text{(since } v_i \in 1_m^{\perp}\text{)}. \end{aligned} \tag{B.60}$$

This leads to

$$\big\langle \hat{A}\hat{C}\hat{B}x_1, x_3 \big\rangle = \big\langle \hat{A}\hat{C}\hat{B}x_1, \sum_{i=1}^{k} z_i \big\rangle = \sum_{i=1}^{k} \overbrace{\big\langle \hat{A}\hat{C}\hat{B}x_1, z_i \big\rangle}^{0} = 0. \tag{B.61}$$

□

Claim 7: $\langle \hat{A}\hat{C}\hat{B}x_3, x_2\rangle = 0$.
Proof: Recall that

$$x_2 = 1_n \otimes y \otimes 1_d, \qquad x_3 = \sum_{i=1}^{k} \overbrace{u_i \otimes v_i \otimes 1_d}^{z_i}, \tag{B.62}$$

with $y \in 1_m^{\perp}$, $\forall i = 1, \dots k : u_i \in 1_n^{\perp}$ and $v_i \in 1_m^{\perp}$. We have:

$$\begin{aligned} \hat{C}\hat{B}z_i &= \hat{C}\big(I_n \otimes \overline{B}\big)\big(u_i \otimes v_i \otimes 1_d\big) \\ &= \hat{C}\big(u_i \otimes \overline{B}(v_i \otimes 1_d)\big) \\ &= \big(I_n \otimes I_m \otimes C\big)\big(u_i \otimes \overline{B}(v_i \otimes 1_d)\big) \\ &= \big(u_i \otimes t_i\big), \end{aligned} \tag{B.63}$$

where $t_i = \big(I_m \otimes C\big)\big(\overline{B}(v_i \otimes 1_d)\big) \in \mathbb{R}^{md}$. We can now write

$$\begin{aligned} \big\langle \hat{A}\hat{C}\hat{B}z_i, x_2\big\rangle &= \big\langle \hat{C}\hat{B}z_i, \hat{A}x_2\big\rangle && \text{since } \hat{A} \text{ is symmetric} \\ &= \big\langle u_i \otimes t_i, 1_n \otimes v \otimes e_1\big\rangle && \text{using (B.63) and (B.55)} \\ &= \overbrace{\big\langle u_i, 1_n\big\rangle}^{0} \cdot \big\langle t_i, y \otimes 1_d\big\rangle \\ &= 0 && \text{since } u_i \in 1_n^{\perp}. \end{aligned} \tag{B.64}$$

This leads to

$$\big\langle \hat{A}\hat{C}\hat{B}x_3, x_2\big\rangle = \big\langle \hat{A}\hat{C}\hat{B}\sum_{i=1}^{k} z_i, x_2\big\rangle = \sum_{i=1}^{k} \overbrace{\big\langle \hat{A}\hat{C}\hat{B}z_i, x_2\big\rangle}^{0} = 0. \tag{B.65}$$

□

Combining everything: We know from (B.22) that

$$\langle \hat{A}\hat{C}\hat{B}x, x\rangle = \sum_{i=1}^{3}\sum_{j=1}^{3} \langle \hat{A}\hat{C}\hat{B}x_i, x_j\rangle. \tag{B.66}$$

Claims 2 to 7 tell us that six of the nine terms in (B.66) are zero, so keeping only the remaining ones leads to

$$\langle \hat{A}\hat{C}\hat{B}x, x\rangle = \langle \hat{A}\hat{C}\hat{B}x_1, x_1\rangle + \langle \hat{A}\hat{C}\hat{B}x_2, x_2\rangle + \langle \hat{A}\hat{C}\hat{B}x_2, x_2\rangle, \tag{B.67}$$

as required. ■

The next 3 lemmas will prove the inequalities of (6.55).

Lemma B.7. *Let $S_2 = 1_n^{\|} \otimes 1_m^{\perp} \otimes 1_d^{\|}$. For any $x_2 \in S_2$, we have:*

$$\left|\langle \hat{B}\hat{C}\hat{A}x_2, x_2\rangle\right| \;+\; \left|\langle \hat{A}\hat{C}\hat{B}x_2, x_2\rangle\right| \leq \lambda_B \cdot \langle x_2, x_2\rangle.$$

Proof: Recall first of all that we can write x_2 as

$$x_2 = 1_n \otimes y \otimes 1_d. \tag{B.68}$$

Next, we saw in (B.55) that

$$\hat{A}x_2 = 1_n \otimes y \otimes e_1. \tag{B.69}$$

So this gives us

$$\begin{aligned} \hat{C}\hat{A}x_2 &= \hat{C}(1_n \otimes y \otimes e_1) \\ &= 1_n \otimes y \otimes C \cdot e_1 \\ &= 1_n \otimes y \otimes \tfrac{d_1}{d_2} \cdot e_2 \quad \text{(from Lemma B.3 (1)).} \end{aligned} \tag{B.70}$$

Also, since $\hat{B} = I_n \otimes \overline{B}$

$$\hat{B}x_2 \;=\; 1_n \otimes \overline{B}(y \otimes 1_d). \tag{B.71}$$

We therefore have

$$\begin{aligned} \langle \hat{B}\hat{C}\hat{A}x_2, x_2\rangle &= \langle \hat{C}\hat{A}x_2, \hat{B}x_2\rangle && (\hat{B} \text{ is symmetric}) \\ &= \tfrac{d_1}{d_2} \cdot \langle 1_n \otimes y \otimes e_2, 1_n \otimes \overline{B}(y \otimes 1_d)\rangle && \text{(from (B.70) and (B.71))} \\ &= \tfrac{d_1}{d_2} \cdot \langle 1_n, 1_n\rangle \;\cdot\; \langle y \otimes e_2,\; \overline{B}(y \otimes e_2)\rangle && \text{(from Lemma B.3 (4))} \\ &= \tfrac{d_1}{d_2} \cdot \langle 1_n, 1_n\rangle \;\cdot\; d_2 \;\cdot\; \langle y, By\rangle && \text{(from Lemma B.2 (2))} \\ &= d_1 \cdot \langle 1_n, 1_n\rangle \;\cdot\; \langle y, By\rangle. \end{aligned} \tag{B.72}$$

Now since $y \in 1_m^{\perp}$ we know by definition that

$$\left|\langle y, Bv\rangle\right| \leq \lambda_B \langle y, y\rangle. \tag{B.73}$$

We therefore have:

$$\begin{aligned} \left|\langle \hat{B}\hat{C}\hat{A}x_2, x_2\rangle\right| &= d_1 \;\cdot\; \left|\langle 1_n, 1_n\rangle\right| \;\cdot\; \left|\langle y, By\rangle\right| \\ &\leq d_1 \;\cdot\; \left|\langle 1_n, 1_n\rangle\right| \;\cdot\; \lambda_B \cdot \langle y, y\rangle \\ &= d_1 \cdot \lambda_B \;\cdot\; \langle 1_n, 1_n\rangle \;\cdot\; \langle y, y\rangle \cdot \tfrac{\langle 1_d, 1_d\rangle}{d} \\ &= \tfrac{d_1}{d} \cdot \lambda_B \cdot \langle 1_n \otimes y \otimes d, 1_n \otimes y \otimes d\rangle \\ &= \tfrac{d_1}{d} \cdot \lambda_B \cdot \langle x_2, x_2\rangle. \end{aligned} \tag{B.74}$$

Analogously, we can show that

$$\left|\langle \hat{A}\hat{C}\hat{B}x_2, x_2\rangle\right| \leq \frac{d_2}{d} \cdot \lambda_B \cdot \langle x_2, x_2\rangle. \tag{B.75}$$

Since $d_1 + d_2 = d$, we can combine (B.74) and (B.75) to obtain

$$\left|\langle \hat{C}\hat{A}x_2, \hat{B}x_2\rangle\right| + \left|\langle \hat{C}\hat{B}x_2, \hat{A}x_2\rangle\right| \leq \lambda_B \cdot \langle x_2, x_2\rangle, \tag{B.76}$$

as required. ■

Lemma B.8. *Let $S_1 = 1_n^{\perp} \otimes 1_m^{\parallel} \otimes 1_d^{\parallel}$. For any $x_1 \in S_1$, we have:*

$$\left|\langle \hat{B}\hat{C}\hat{A}x_1, x_1\rangle\right| \quad + \quad \left|\langle \hat{A}\hat{C}\hat{B}x_1, x_1\rangle\right| \leq \lambda_A \cdot \langle x_1, x_1\rangle.$$

Proof: Analogous to that of Lemma B.7 (though a basis change is required). ■

Theorem B.9. *Let $m(a, b, c)$ be the function defined in (6.33) of Theorem 6.11. Let $x_3 \in S_3$. Then*

$$\left|\langle \hat{B}\hat{C}\hat{A}x_3, x_3\rangle\right| \quad + \quad \left|\langle \hat{A}\hat{C}\hat{B}x_3, x_3\rangle\right| \leq m(\lambda_A, \lambda_B, \lambda_C) \cdot \langle x_3, x_3\rangle. \tag{B.77}$$

Proof: Since $\hat{B}$ is symmetric, we have

$$\langle \hat{B}\hat{C}\hat{A}x_3, x_3\rangle = \langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle. \tag{B.78}$$

Let $\gamma = \hat{A}x_3$ and $\mu = \hat{B}x_3$. Notice that

$$\gamma \in \mathbb{R}^n \otimes \mathbb{R}^m \otimes R_1, \qquad \mu \in \mathbb{R}^n \otimes \mathbb{R}^m \otimes R_2. \tag{B.79}$$

We define $\gamma^{\parallel}$ and $\gamma^{\perp}$ as follows:

$$\gamma^{\parallel} = \mathcal{M}_{nm}(\gamma) \otimes \frac{e_1}{d_1}, \qquad \gamma^{\perp} = \gamma - \gamma^{\parallel}.$$

So $\gamma^{\parallel}$ is uniform and $\gamma^{\perp}$ is anti-uniform over the left nodes of each C-cloud (so $\gamma^{\parallel} \perp \gamma^{\perp}$), and $\gamma = \gamma^{\parallel} + \gamma^{\perp}$. In the same way we decompose μ with respect to the right nodes of each C-cloud. Formally,

$$\mu^{\parallel} = \mathcal{M}_{nm}(\mu) \otimes \frac{e_2}{d_2}, \qquad \mu^{\perp} = \mu - \mu^{\parallel}.$$

Claim 1: $\langle \hat{C}\gamma^{\|}, \mu^{\perp}\rangle = \langle \hat{C}\gamma^{\perp}, \mu^{\|}\rangle = 0$.
Proof: $\gamma^{\|} \in \mathbb{R}^n \otimes \mathbb{R}^m \otimes e_1^{\|}$, and $\hat{C} = I_n \otimes I_m \otimes C$. Now for any $w \in e_1^{\|}$ we have $Cw \in e_2^{\|}$, so $\hat{C}\gamma^{\|} \in \mathbb{R}^n \otimes \mathbb{R}^m \otimes e_2^{\|}$. On the other hand, $\mu^{\perp} \in \mathbb{R}^n \otimes \mathbb{R}^m \otimes e_2^{\perp}$, which means that $\langle \hat{C}\gamma^{\|}, \mu^{\perp}\rangle = 0$. The second part can be shown in exactly the same way. □

Using Claim 1 we see that:

$$\begin{aligned}
\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle &= \langle \hat{C}\gamma, \mu\rangle \\
&= \langle \hat{C}\gamma^{\|} + \hat{C}\gamma^{\perp}, \mu^{\|} + \mu^{\perp}\rangle \\
&= \langle \hat{C}\gamma^{\|}, \mu^{\|}\rangle + \overbrace{\langle \hat{C}\gamma^{\|}, \mu^{\perp}\rangle}^{0} + \overbrace{\langle \hat{C}\gamma^{\perp}, \mu^{\|}\rangle}^{0} + \langle \hat{C}\gamma^{\perp}, \mu^{\perp}\rangle \qquad \text{(Claim 1)} \\
&= \langle \hat{C}\gamma^{\|}, \mu^{\|}\rangle + \langle \hat{C}\gamma^{\perp}, \mu^{\perp}\rangle \\
&= \|\hat{C}\gamma^{\|}\| \cdot \|\mu^{\|}\| \cdot \cos\left(\hat{C}\gamma^{\|}, \mu^{\|}\right) + \|\hat{C}\gamma^{\perp}\| \cdot \|\mu^{\perp}\| \cdot \cos\left(\hat{C}\gamma^{\perp}, \mu^{\perp}\right).
\end{aligned} \tag{B.80}$$

Therefore we have

$$\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle\right| \leq \|\hat{C}\gamma^{\|}\| \cdot \|\mu^{\|}\| + \|\hat{C}\gamma^{\perp}\| \cdot \|\mu^{\perp}\|. \tag{B.81}$$

Now, we know that $x_3 \in S_3$ with $S_3 = 1_n^{\perp} \otimes 1_m^{\perp} \otimes 1_d^{\|}$. Let $u_0, \ldots u_{n-1}$ and $v_0, \ldots, v_{m-1}$ be the normalized eigenvectors of A and B respectively. As usual u_0 and v_0 are uniform while $u_1, \ldots u_{n-1}$ and $v_1, \ldots, v_{m-1}$ form orthonormal bases of $1_n^{\perp}$ and $1_m^{\perp}$ respectively. Consequently,

$$\left\{u_i \otimes v_j \otimes \frac{1_d}{\sqrt{d}} \;\middle|\; i \in [n-1],\ j \in [m-1]\right\}$$

is an orthonormal basis of S_3. So there are $\alpha_{ij} \in \mathbb{R}$ with

$$x_3 = \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} \alpha_{ij} \cdot (u_i \otimes v_j \otimes \frac{1_d}{\sqrt{d}}). \tag{B.82}$$

Claim 2: $\|x_3\|^2 = \frac{d}{d_2} \cdot \|\mu\|^2$
Proof: Since the basis in which x_3 is expressed in (B.82) is orthonormal, we have

$$\|x_3\|^2 = \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} \alpha_{ij}^2. \tag{B.83}$$

Now for $i = 1, \ldots, n-1$ let $w_i \in \mathbb{R}^m$ be defined as

$$w_i = \sum_{j=1}^{m-1} \alpha_{ij} v_j, \tag{B.84}$$

so that

$$x_3 = \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} \alpha_{ij}(u_i \otimes v_j \otimes \frac{1_d}{\sqrt{d}}) = \sum_{i=1}^{n-1}(u_i \otimes w_i \otimes \frac{1_d}{\sqrt{d}}). \tag{B.85}$$

Notice that because the vectors $u_1, \ldots, u_{n-1}$ are pairwise orthogonal, the $u_i \otimes w_i \otimes 1_d$ are also pairwise orthogonal.

$$\begin{aligned}
\|\mu\|^2 &= \|\hat{B}x_3\|^2 \\
&= \|\sum_{i=1}^{n-1} \hat{B} \cdot (u_i \otimes w_i \otimes \tfrac{1_d}{\sqrt{d}})\|^2 \\
&= \|\sum_{i=1}^{n-1} u_i \otimes \overline{B} \cdot (w_i \otimes \tfrac{1_d}{\sqrt{d}})\|^2 \\
&= \sum_{i=1}^{n-1} \|u_i \otimes \overline{B} \cdot (w_i \otimes \tfrac{1_d}{\sqrt{d}})\|^2 \quad \text{(since the } u_i\text{'s are pairwise orthogonal)} \\
&= \sum_{i=1}^{n-1} \|\overline{B} \cdot (w_i \otimes \tfrac{1_d}{\sqrt{d}})\|^2 \qquad \text{(since } \|u_i\| = 1 \ \forall i\text{)}.
\end{aligned} \tag{B.86}$$

From Lemma B.3 (4) we know that

$$\overline{B} \cdot (w_i \otimes \frac{1_d}{\sqrt{d}}) = \overline{B} \cdot (w_i \otimes \frac{e_2}{\sqrt{d}}). \tag{B.87}$$

Since $\overline{B}$ is a permutation on elements of $\mathbb{R}^m \otimes R_2$, it is length preserving on these elements. This leads to

$$\|\overline{B} \cdot (w_i \otimes \frac{1_d}{\sqrt{d}})\| = \|w_i \otimes \frac{e_2}{\sqrt{d}}\| = \|w_i\| \cdot \|\frac{e_2}{\sqrt{d}}\| = \sqrt{\frac{d_2}{d}} \cdot \|w_i\|. \tag{B.88}$$

So plugging (B.88) into (B.86) gives us

$$\|\mu\|^2 = \frac{d_2}{d} \cdot \sum_{i=1}^{n-1} \|w_i\|^2. \tag{B.89}$$

Since $w_i = \sum_{j=1}^{m} \alpha_{ij} v_j$, and the v_j's are pairwise orthogonal, we obtain

$$\|w_i\|^2 = \sum_{j=1}^{m-1} \alpha_{ij}^2 \cdot \|v_j\|^2 = \sum_{j=1}^{m-1} \alpha_{ij}^2. \tag{B.90}$$

Plugging (B.90) into (B.89) gives us

$$\|\mu\|^2 = \frac{d_2}{d} \cdot \sum_{i=1}^{n-1}\sum_{j=1}^{m-1} \alpha_{ij}^2 = \frac{d_2}{d}\|x_3\|^2, \tag{B.91}$$

where the last equality follows from (B.83). □

Claim 3: $\|x_3\|^2 = \frac{d}{d_1}\|\gamma\|^2$.
Proof: Analogous to that of Claim 2. □

Claim 4: $\big\|\hat{C}\gamma^{\|}\big\| = \sqrt{\frac{d_1}{d_2}} \cdot \|\gamma^{\|}\|$
Proof: Lemma B.3 (1) tells us that

$$C \cdot \frac{e_1}{d_1} = \frac{e_2}{d_2}. \tag{B.92}$$

Now recall that $\gamma^{\|} = \mathcal{M}_{nm}(\gamma) \otimes \frac{e_1}{d_1}$, and $\hat{C} = I_{nm} \otimes C$. We therefore have

$$\hat{C}\gamma^{\|} = \mathcal{M}_{nm}(\gamma) \otimes C \cdot \frac{e_1}{d_1} = \mathcal{M}_{nm}(\gamma) \otimes \frac{e_2}{d_2}. \tag{B.93}$$

This leads to

$$\big\|\hat{C}\gamma^{\|}\big\| = \big\|\mathcal{M}_{nm}(\gamma)\big\| \cdot \big\|\frac{e_2}{d_2}\big\| = \big\|\mathcal{M}_{nm}(\gamma)\big\| \cdot \frac{1}{\sqrt{d_2}}, \tag{B.94}$$

and therefore

$$\big\|\hat{C}\gamma^{\|}\big\| = \sqrt{\frac{d_1}{d_2}} \cdot \|\gamma^{\|}\|. \tag{B.95}$$

□

Claim 5: $\|\hat{C}\gamma^{\perp}\| \leq \lambda_C \cdot \sqrt{\frac{d_1}{d_2}} \cdot \|\gamma^{\perp}\|$.
Proof: We saw in Proposition 5.44 that for any $x \in e_1^{\perp}$

$$\big\|Cx\big\| \leq \sqrt{\frac{d_1}{d_2}} \cdot \lambda_C \cdot \|x\|. \tag{B.96}$$

Now we can decompose $\gamma^{\perp} \in \mathbb{R}^{nm} \otimes e_1^{\perp}$ as

$$\gamma^{\perp} = \begin{pmatrix} \gamma_{1,1}^{\perp} \\ \vdots \\ \gamma_{n,m}^{\perp} \end{pmatrix}, \tag{B.97}$$

where $\gamma_{i,j}^{\perp} \in e_1^{\perp}$. Therefore

$$\hat{C}\gamma^{\perp} = \begin{pmatrix} C\gamma_{1,1}^{\perp} \\ \vdots \\ C\gamma_{n,m}^{\perp} \end{pmatrix}, \tag{B.98}$$

and so

$$\begin{aligned} \|\hat{C}\gamma^{\perp}\|^2 &= \textstyle\sum_{i=1}^{n}\sum_{j=1}^{m} \|\hat{C}\gamma_{i,j}^{\perp}\|^2 \\ &\leq \textstyle\sum_{i=1}^{n}\sum_{j=1}^{m} \frac{d_1}{d_2} \cdot \lambda_C^2 \cdot \|\gamma_{i,j}^{\perp}\|^2 \quad \text{(from (B.96), since } \gamma_{i,j}^{\perp} \in e_1^{\perp}\text{)} \\ &= \frac{d_1}{d_2} \cdot \lambda_C^2 \cdot \|\gamma^{\perp}\|^2. \end{aligned} \tag{B.99}$$

And so we conclude that

$$\|\hat{C}\gamma^{\perp}\| \leq \lambda_C \cdot \sqrt{\frac{d_1}{d_2}} \cdot \|\gamma^{\perp}\|. \tag{B.100}$$

□

Continuing with our proof, from (B.81) we obtain

$$\frac{\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle\right|}{\langle x_3, x_3\rangle} \leq \frac{\|\mu^{\|}\| \cdot \|\hat{C}\gamma^{\|}\|}{\|x_3\|^2} + \frac{\|\mu^{\perp}\| \cdot \|\hat{C}\gamma^{\perp}\|}{\|x_3\|^2}, \tag{B.101}$$

and so Claims 4 and 5 then give us

$$\frac{\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle\right|}{\langle x_3, x_3\rangle} \leq \sqrt{\frac{d_1}{d_2}} \cdot \frac{\|\mu^{\|}\|}{\|x_3\|} \cdot \frac{\|\gamma^{\|}\|}{\|x_3\|} + \lambda_C \cdot \sqrt{\frac{d_1}{d_2}} \cdot \frac{\|\mu^{\perp}\|}{\|x_3\|} \cdot \frac{\|\gamma^{\perp}\|}{\|x_3\|}. \tag{B.102}$$

From Claims 2 and 3 we know that

$$\|x_3\| = \sqrt{\frac{d}{d_1}}\|\gamma\|, \qquad\qquad \|x_3\| = \sqrt{\frac{d}{d_2}}\|\mu\|. \tag{B.103}$$

Therefore (B.102) and (B.103) lead to

$$\frac{\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle\right|}{\langle x_3, x_3\rangle} \leq \sqrt{\frac{d_1}{d_2}} \cdot \frac{\sqrt{d_1 d_2}}{d} \cdot \frac{\|\mu^{\|}\|}{\|\mu\|} \cdot \frac{\|\gamma^{\|}\|}{\|\gamma\|} + \lambda_C \cdot \sqrt{\frac{d_1}{d_2}} \cdot \frac{\sqrt{d_1 d_2}}{d} \cdot \frac{\|\mu^{\perp}\|}{\|\mu\|} \cdot \frac{\|\gamma^{\perp}\|}{\|\gamma\|}. \tag{B.104}$$

We now let θ_A be the angle between γ and $\gamma^{\|}$, and θ_B be the angle between μ and $\mu^{\|}$. Notice that

$$\cos(\theta_A) = \frac{\|\gamma^{\|}\|}{\|\gamma\|}, \qquad\qquad \sin(\theta_A) = \frac{\|\gamma^{\perp}\|}{\|\gamma\|}, \tag{B.105}$$

and likewise for θ_B. With these definitions, (B.104) can be reduced to

$$\frac{\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle\right|}{\langle x_3, x_3\rangle} \leq \frac{d_1}{d} \cdot \cos(\theta_A) \cdot \cos(\theta_B) + \lambda_C \cdot \frac{d_1}{d} \cdot \sin(\theta_A) \cdot \sin(\theta_B). \tag{B.106}$$

Since in general for any $\theta \in [0, \pi/2]$ we have $\sin(\theta) = \sqrt{1 - \cos^2(\theta)}$, we can deduce:

$$\frac{\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle\right|}{\langle x_3, x_3\rangle} \leq \frac{d_1}{d} \cdot \cos(\theta_A) \cdot \cos(\theta_B) + \lambda_C \cdot \frac{d_1}{d} \cdot \sqrt{(1 - \cos^2(\theta_A)) \cdot (1 - \cos^2(\theta_B))}. \tag{B.107}$$

In exactly the same way it can be shown that

$$\frac{\left|\langle \hat{C}\hat{B}x_3, \hat{A}x_3\rangle\right|}{\langle x_3, x_3\rangle} \leq \frac{d_2}{d} \cdot \cos(\theta_A) \cdot \cos(\theta_B) + \lambda_C \cdot \frac{d_2}{d} \cdot \sqrt{(1 - \cos^2(\theta_A)) \cdot (1 - \cos^2(\theta_B))}. \tag{B.108}$$

Since $d = d_1 + d_2$, combining (B.107) and (B.108) gives us

$$\frac{\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3\rangle\right|}{\langle x_3, x_3\rangle} + \frac{\left|\langle \hat{C}\hat{B}x_3, \hat{A}x_3\rangle\right|}{\langle x_3, x_3\rangle} \leq f(\cos(\theta_A), \cos(\theta_B), \lambda_C), \tag{B.109}$$

where

$$f(a, b, c) = ab + c \cdot \sqrt{(1 - a^2) \cdot (1 - b^2)}. \tag{B.110}$$

Claim 6: $\cos(\theta_B) \leq \lambda_B$
Proof: First recall from (B.82) that there are $\alpha_{ij} \in \mathbb{R}$ with

$$x_3 = \sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \alpha_{ij} \cdot (u_i \otimes v_j \otimes \frac{1_d}{\sqrt{d}}). \tag{B.111}$$

Setting $b_j = \overline{B}(v_j \otimes \frac{1_d}{\sqrt{d}}) \in \mathbb{R}^{md}$ gives us

$$\begin{aligned}
\mu &= \hat{B}x_3 \\
&= \hat{B} \cdot \big(\textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \alpha_{ij} \cdot (u_i \otimes v_j \otimes \frac{1_d}{\sqrt{d}}) \big) \\
&= \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \alpha_{ij} \cdot (I_n \otimes \overline{B}) \cdot (u_i \otimes v_j \otimes \frac{1_d}{\sqrt{d}}) \\
&= \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \alpha_{ij} \cdot (I_n \cdot u_i) \otimes \overbrace{\overline{B} \cdot (v_j \otimes \frac{1_d}{\sqrt{d}})}^{b_j} \\
&= \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \frac{\alpha_{ij}}{\sqrt{d}} \cdot (u_i \otimes b_j).
\end{aligned}$$

Recall that $\mu^{\|}$ was defined as

$$\mu^{\|} = \mathcal{M}_{nm}(\mu) \otimes \frac{e_2}{d_2}. \tag{B.112}$$

Lemma B.3 (5) tells us that

$$\mathcal{M}_{nm}(u_i \otimes b_j) = u_i \otimes \mathcal{M}_m(b_j), \tag{B.113}$$

and therefore

$$\mu^{\|} = \sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \frac{\alpha_{ij}}{\sqrt{d}} \cdot \mathcal{M}_{nm}(u_i \otimes b_j) \otimes \frac{e_2}{d_2} = \sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \frac{\alpha_{ij}}{\sqrt{d}} \cdot u_i \otimes \mathcal{M}_m(b_j) \otimes \frac{e_2}{d_2}.$$

Now $b_j = \overline{B}(v_j \otimes e_2)$, and therefore Lemma B.2 (2) tells us that

$$\mathcal{M}_m(b_j) = d_2 \cdot Bv_j. \tag{B.114}$$

This leads to

$$\begin{aligned}
\|\mu^{\|}\|^2 &= \| \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \frac{\alpha_{ij}}{\sqrt{d}} \cdot u_i \otimes \mathcal{M}_m(b_j) \otimes \frac{e_2}{d_2} \|^2 \\
&= \| \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \frac{\alpha_{ij}}{\sqrt{d}} \cdot u_i \otimes (d_2 \cdot Bv_j) \otimes \frac{e_2}{d_2} \|^2 && \text{(from (B.114))} \\
&= \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \frac{\alpha_{ij}}{\sqrt{d}} \cdot \|u_i\|^2 \cdot \|d_2 \cdot Bv_j\|^2 \cdot \| \frac{e_2}{d_2} \|^2 && \text{(since the } u_i\text{'s are pairwise orthogonal)} \\
&= \textstyle\sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \frac{\alpha_{ij}}{\sqrt{d}} \cdot d_2^2 \cdot \|Bv_j\|^2 \cdot \frac{1}{d_2} && \text{(since } \forall i: \ \|u_i\| = 1\text{).}
\end{aligned} \tag{B.115}$$

Now since $v_j \in 1_m^{\perp}$, by the definition of λ_B we have

$$\|Bv_j\| \leq \lambda_B \|v_j\| = \lambda_B. \tag{B.116}$$

So plugging this into (B.115) gives us

$$\|\mu^{\|}\|^2 \leq \lambda_B^2 \cdot \frac{d_2}{d} \cdot \sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \alpha_{ij}. \tag{B.117}$$

We also know from (B.91) that

$$\|\mu\|^2 = \frac{d_2}{d} \cdot \sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \alpha_{ij}. \tag{B.118}$$

So combining (B.117) and (B.118) leads to

$$\frac{\|\mu^{\|}\|}{\|\mu\|} \leq \lambda_B. \tag{B.119}$$

□

Claim 7: $\cos(\theta_A) \leq \lambda_A$.
Proof: Analogous to that of Claim 6. □

Combining it all: Since $\cos(\theta_A) \in [0, \lambda_A]$, and $\cos(\theta_B) \in [0, \lambda_B]$, (B.109) tells us that

$$\frac{\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3 \rangle\right|}{\langle x_3, x_3 \rangle} \;+\; \frac{\left|\langle \hat{C}\hat{B}x_3, \hat{A}x_3 \rangle\right|}{\langle x_3, x_3 \rangle} \leq M(\lambda_A, \lambda_B, \lambda_C), \tag{B.120}$$

where

$$M(\lambda_A, \lambda_B, \lambda_C) = \max\left\{ f(a, b, \lambda_C) \;\middle|\; a \in [0, \lambda_A], b \in [0, \lambda_B] \right\}. \tag{B.121}$$

Theorem B.10 below states that

$$M(\lambda_A, \lambda_B, \lambda_C) \leq m(\lambda_A, \lambda_B, \lambda_C), \tag{B.122}$$

and so we deduce

$$\left|\langle \hat{C}\hat{A}x_3, \hat{B}x_3 \rangle\right| \;+\; \left|\langle \hat{C}\hat{B}x_3, \hat{A}x_3 \rangle\right| \leq m(\lambda_A, \lambda_B, \lambda_C) \cdot \langle x_3, x_3 \rangle, \tag{B.123}$$

as required. ■

Theorem B.10. *Suppose that we have $\lambda_A, \lambda_B, c \in [0,1]$ with $\lambda_B \leq \lambda_A$. Let $f(a,b,c)$ be the function*

$$f(a,b,c) = ab + c\sqrt{(1-a^2)(1-b^2)}, \tag{B.124}$$

let $M(\lambda_A, \lambda_B, c)$ be the quantity

$$M(\lambda_A, \lambda_B, c) = \max\big\{ f(a,b,c) \;\mid\; a \in [0,\lambda_A], b \in [0,\lambda_B] \big\}, \tag{B.125}$$

and let $g(b,c)$ be the function

$$g(b,c) = \frac{1}{\sqrt{\frac{c^2}{b^2} - c^2 + 1}}. \tag{B.126}$$

Then we have:

$$M(\lambda_A, \lambda_B, c) = \begin{cases} f(\lambda_A, \lambda_B, c) & \text{if } \lambda_A \leq g(\lambda_B, c) \\ f(g(\lambda_B, c), \lambda_B, c) & \text{otherwise}. \end{cases} \tag{B.127}$$

So another way of putting this is

$$M(\lambda_A, \lambda_B, c) = f\bigg(\min\big(\lambda_A, g(\lambda_B, c)\big),\;\; \lambda_B,\;\; c \bigg) := m(\lambda_A, \lambda_B, \lambda_C), \tag{B.128}$$

which means that $m(a,b,c)$ is as defined in Theorem 6.11.

Proof: First of all, we have

$$\frac{\partial}{\partial a} f(a,b,c) = b - \frac{ac\sqrt{1-b^2}}{\sqrt{1-a^2}}. \tag{B.129}$$

Now

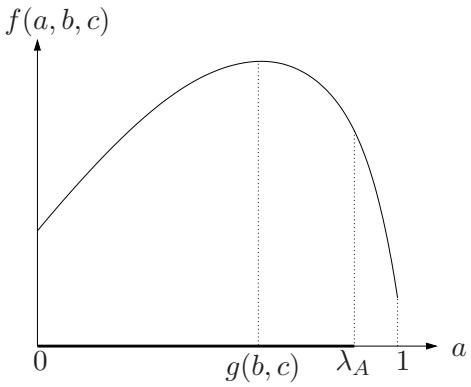
$$\begin{aligned} \tfrac{\partial}{\partial a} f(a,b,c) \geq 0 \iff b\sqrt{1-a^2} \geq ac\sqrt{1-b^2} \quad &\iff \quad \sqrt{\tfrac{1}{a^2} - 1} \geq c\sqrt{\tfrac{1}{b^2} - 1} \\ &\iff \quad \tfrac{1}{a^2} \geq c^2 \cdot (\tfrac{1}{b^2} - 1) + 1 \\ &\iff \quad a \leq \tfrac{1}{\sqrt{\frac{c^2}{b^2} - c^2 + 1}} \\ &\iff \quad a \leq g(b,c), \end{aligned}$$

where $g(b,c)$ is taken from (B.126).

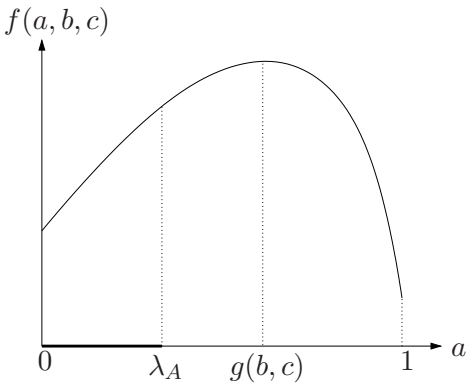
Furthermore, for any $b, c \in [0,1]$ we have

$$\frac{\partial}{\partial a} f(0,b,c) \;\; = \;\; b \;\; \geq \;\; 0, \tag{B.130}$$

which means that for fixed b and c, $f(a,b,c)$ is increasing when $a \in [0, g(b,c)]$, and decreasing when $a \in [g(b,c), 1]$. So over the range $a \in [0, \lambda_A]$ (and for fixed $b, c \in [0,1]$), depending on whether $\lambda_A \leq g(b,c)$ we have one of the two following cases:



Case 1



Case 2

The maxima of $f(a,b,c)$ over the range $a \in [0,\lambda_A]$ will therefore be at $a_0 = \min\big(\lambda_A, g(b,c)\big)$. We will consider the two values a_0 can take separately. We will show that in both cases, when $b \in [0,\lambda_B]$ and $c \in [0,1]$ we have:

$$f(a_0,b,c) \le f(a_0,\lambda_B,c), \tag{B.131}$$

from which the result follows.

Case 1: $g(b,c) < \lambda_A$. So $a_0 = g(b,c)$. Let

$$h_1(b,c) = f\big(a_0,b,c\big) = f\big(g(b,c),b,c\big) = \frac{b}{\sqrt{\frac{c^2}{b^2}-c^2+1}} + c\cdot\sqrt{(1-b^2)\cdot(1-\frac{1}{\frac{c^2}{b^2}-c^2+1})}. \tag{B.132}$$

It can then be checked that

$$\frac{\partial}{\partial b}h_1(b,c) = \frac{1-c^2}{\sqrt{(c/b)^2(1-b)^2+1}}, \tag{B.133}$$

which means that

$$\forall\, b,c \in [0,1]: \quad \frac{\partial}{\partial b}h_1(b,c) \ge 0. \tag{B.134}$$

Therefore for fixed c, $h_1(b,c)$ increases with b and so when b is in the range $[0,\lambda_B]$ it is maximal when $b = \lambda_B$:

$$\forall\, b,c \in [0,1]: \quad h_1(b,c) \le h_1(\lambda_B,c). \tag{B.135}$$

Case 2: $\lambda_A \le g(b,c)$. So $a_0 = \lambda_A$. Let

$$h_2(b,c) = f\big(a_0,b,c\big) = f\big(\lambda_A,b,c\big). \tag{B.136}$$

Now for any a,b,c we have $f(a,b,c) = f(b,a,c)$. Therefore

$$h_2(b,c) = f\big(b,\lambda_A,c\big). \tag{B.137}$$

We showed above that for fixed b and c, $f(a,b,c)$ is increasing when $a \in [0, g(b,c)]$. So applying this to (B.137) we can deduce that for fixed c, $h_2(b,c)$ is increasing for $b \in [0, g(\lambda_A, c)]$.

Claim: $\forall x, c \in [0,1]: \quad g(x,c) \geq x$.
Proof: Recall from (B.126) that $g(x,c)$ is defined as

$$g(x,c) = \frac{1}{\sqrt{\frac{c^2}{x^2} - c^2 + 1}}. \tag{B.138}$$

Now we have:

$$\frac{x}{g(x,c)} = x\sqrt{\frac{c^2}{x^2} - c^2 + 1}. \tag{B.139}$$

Therefore

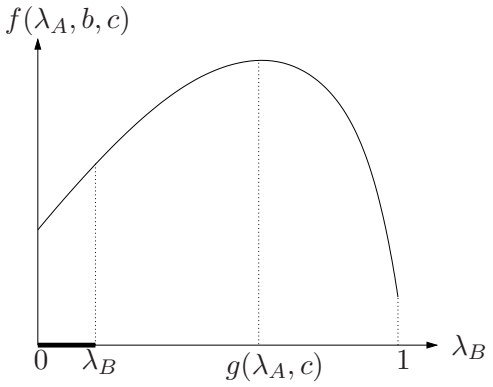
$$\begin{aligned}
\left(\tfrac{x}{g(x,c)}\right)^2 &= x^2\left(\tfrac{c^2}{x^2} - c^2 + 1\right) \\
&= c^2 - x^2c^2 + x^2 \\
&= c^2\left(1 - x^2\right) + x^2 \\
&\leq \left(1 - x^2\right) + x^2 \qquad \text{(since } c \in [0,1] \text{ and } 1 - x^2 \geq 0\text{)} \\
&= 1,
\end{aligned} \tag{B.140}$$

and the result follows immediately. □

From this claim we obtain

$$\lambda_B \leq \lambda_A \leq g(\lambda_A, c). \tag{B.141}$$

(The first inequality is an assumption we made in the statement of the proposition). We have the following situation:



So over the range $b \in [0, \lambda_B]$, $h_2(b, c)$ is maximal when $b = \lambda_B$:

$$h_2(b, c) \leq h_2(\lambda_B, c), \tag{B.142}$$

which concludes case 2. ■

Appendix C

Proofs

This appendix contains the proofs that are either too technical to feature in the main chapters, or that do not involve results of central importance.

Lemma 3.16. For any $\epsilon_1 > 0$, there are N_1, Γ_1 with

$$N \geq N_1, \gamma \geq \Gamma_1 \implies \left| P_w - \frac{1}{2} \right| \leq \epsilon_1. \tag{C.1}$$

Proof: Using the definition $\gamma = \frac{w}{n^{1-y}}$, we get

$$P_w = \frac{1}{2} - \frac{1}{2}\left(1 - \frac{2w}{Rn}\right)^{(nR)^y} = \frac{1}{2} - \frac{1}{2}\left(1 - \frac{2\gamma}{Rn^y}\right)^{(nR)^y}, \tag{C.2}$$

and therefore

$$\left| P_w - \frac{1}{2} \right| = \left| \frac{1}{2}\left(1 - \frac{2\gamma}{Rn^y}\right)^{(nR)^y} \right| = \left| \frac{1}{2}\left(1 - \frac{2\gamma}{R^{1-y} \cdot (nR)^y}\right)^{(nR)^y} \right|. \tag{C.3}$$

To make notation simpler, we let

$$x = (nR)^y, \qquad a = \frac{2\gamma}{R^{1-y}}. \tag{C.4}$$

So the expression in (C.3) is

$$\left| \frac{1}{2}\left(1 - \frac{a}{x}\right)^x \right|. \tag{C.5}$$

Studying the asymptotic properties of (C.5) is a little delicate since we need to consider the asymptotic behavior of two variables x and a. Furthermore the growth rate of one with respect to the other could behave in many different ways (corresponding to how w grows with n). We start by seeing that

$$\left(1 - \frac{a}{x}\right)^x = \exp\left[x \cdot \ln\left(1 - \frac{a}{x}\right)\right],$$

and so recalling that the Maclaurin expansion of $\ln(1-z)$ is

$$-z - \frac{1}{2} \cdot z^2 - \frac{1}{3} \cdot z^3 - \frac{1}{4} \cdot z^4 - \ldots,$$

we obtain

$$\left(1-\frac{a}{x}\right)^x = \exp\left[x\cdot\left(-\frac{a}{x}-\frac{a^2}{2x^2}-\frac{a^3}{3x^3}-\dots\right)\right] = \exp\left[-a-\frac{a^2}{2x}-\frac{a^3}{3x^2}-\dots\right]. \tag{C.6}$$

Now since $x>0$, no matter how x behaves, we will have

$$\lim_{a\to\infty}\left(-a-\frac{a^2}{2x}-\frac{a^3}{3x^2}-\dots\right) = -\infty. \tag{C.7}$$

Notice that x could have any behavior as a gets large and (C.7) would still hold. Now combining (C.6) and (C.7) we obtain

$$\lim_{a\to\infty}\left(1-\frac{a}{x}\right)^x = 0.$$

So replacing a and x according to (C.4), and combining this with (C.3), we can deduce that

$$\lim_{\gamma\to\infty}\left|P_w-\frac{1}{2}\right| = 0.$$

Formally, this means that for any $\epsilon_1>0$ there is Γ_1 with

$$\gamma\geq\Gamma_1 \implies \left|P_w-\frac{1}{2}\right|\leq\epsilon_1.$$

Now recall that the only values of w we consider are $w=1,\dots,\lfloor nR\rfloor$. So since $\gamma=\frac{w}{n^{1-y}}$, for w to get large it is also necessary that n be large enough. Although this is sort of implicit in the statement "$\gamma\geq\Gamma_1$", we make this requirement explicit, by saying there are N_1,Γ_1 with

$$n\geq N_1,\gamma\geq\Gamma_1 \implies \left|P_w-\frac{1}{2}\right|\leq\epsilon_1.$$

■

Theorem 3.22. For any b,x with $b\geq 1$ and $0\leq x\leq 1$ we have

$$1-x\leq\left(1-\frac{x}{b}\right)^b. \tag{C.8}$$

Proof: Using the Maclaurin expansion of $\ln(1-x)$, we obtain:

$$\begin{aligned}
\ln(1-x) &= -x - \frac{1}{2}x^2 - \frac{1}{3}x^3 - \frac{1}{4}x^4 - \dots \\
&= -b\cdot\frac{x}{b} - b^2\cdot\frac{1}{2}\Big(\frac{x}{b}\Big)^2 - b^3\cdot\frac{1}{3}\Big(\frac{x}{b}\Big)^2 - \dots \\
&\overbrace{\leq}^{b\geq 1} -b\cdot\frac{x}{b} - b\cdot\frac{1}{2}\Big(\frac{x}{b}\Big)^2 - b\cdot\frac{1}{3}\Big(\frac{x}{b}\Big)^2 - \dots \\
&= b\cdot\left(\frac{x}{b} - \frac{1}{2}\Big(\frac{x}{b}\Big)^2 - \frac{1}{3}\Big(\frac{x}{b}\Big)^2 - \dots\right) \\
&= b\cdot\ln\Big(1-\frac{x}{b}\Big) \\
&= \ln\Big(\big(1-\frac{x}{b}\big)^b\Big).
\end{aligned}$$

Because the function $\ln$ is increasing, we obtain

$$1 - x \leq \Big(1-\frac{x}{b}\Big)^b, \tag{C.9}$$

as required. ■

Lemma 3.33. For all $x, b \in \mathbb{R}_{>0}$ we have

$$-x\ln(bx) \leq \frac{1}{be}.$$

Proof: Suppose that $b \in R_{>0}$ is fixed, and let $t(x) = -x\ln(bx)$. Differentiating we get

$$t'(x) = \frac{\partial}{\partial x}t(x) = -\ln(bx) - \frac{x}{bx}b = -\ln(bx) - 1.$$

Now,

$$\begin{aligned}
t'(x) = 0 &\iff \ln(bx) = -1 \\
&\iff bx = \tfrac{1}{e} \\
&\iff x = \tfrac{1}{be}.
\end{aligned}$$

Furthermore, $t''(x) = -\frac{1}{bx} < 0$, so $x = \frac{1}{be}$ is a maxima for t.

$$t\big(\frac{1}{be}\big) = -\frac{1}{be}\cdot\ln\big(b\frac{1}{be}\big) = -\frac{1}{be}\cdot\ln\big(\frac{1}{e}\big) = \frac{1}{be}.$$

Now $x = \frac{1}{be}$ is a maxima, so $\forall x \in \mathbb{R}_{>0}: \;\; t(x) \leq t\big(\frac{1}{be}\big)$, and the result follows. ■

Lemma 3.45. Let $f : \mathbb{R} \to \mathbb{R}$ be a bounded function. Then for any $\epsilon > 0$ there is X with

$$x \geq X \implies \exp\big(-f(x)\big) - \epsilon \leq \Big(1 - \frac{f(x)}{x}\Big)^x \leq \exp\big(-f(x)\big).$$

Proof: The second inequality follows from the fact that for any $z \in \mathbb{R} : (1+z) \leq \exp(z)$.

Next, we have:

$$\Big(1 - \frac{f(x)}{x}\Big)^x = \exp\Big[x \cdot \ln\Big((1 - \frac{f(x)}{x}\Big)\Big].$$

Recall that the Maclaurin expandion of $\ln(1-z)$ is

$$\ln(1-z) = -z \ - \ \frac{1}{2}z^2 \ - \ \frac{1}{3}z^3 \ - \ \frac{1}{4}z^4 \ - \ \ldots .$$

This leads to

$$\Big(1 - \frac{f(x)}{x}\Big)^x = \exp\Big[- f(x) \ - \ \frac{f(x)^2}{2x} \ - \ \frac{f(x)^3}{3x^2} \ - \ \ldots\Big]. \tag{C.10}$$

Therefore since $f(x)$ is bounded, by making x large enough, we can bring (C.10) as close as necessary to $\exp\big(-f(x)\big)$. The result then follows. ■

Lemma 6.8. Let $\hat{A} = \mathrm{DP}[A]$. Then for any $\sigma \in \mathbb{R}^n$ we have

$$\mathcal{M}_n\big(\hat{A}(\sigma \otimes \frac{1_d}{d})\big) = A\sigma. \tag{C.11}$$

Proof: Recall that in the notation from definition 5.22, for a vertex $a \in [n]$ and a label $k \in [d]$, $a[k]$ denotes the k^{th} neighbor of a.

Now multiplying $\sigma \in \mathbb{R}^n$ by A can be described as follows:

$$\big(A\sigma\big)_i = \frac{1}{d} \cdot \sum_{k=1}^{d} \sigma_{i[k]}. \tag{C.12}$$

Next recall that the transition matrix of $\hat{A} = \mathrm{DP}[A]$ was defined in (5.39):

$$\big(\hat{A}\big)_{ik,jl} = \begin{cases} 1 & \text{if } \ j = i[k] \ \text{ and } \ i = j[\ell] \\ 0 & \text{otherwise.} \end{cases} \tag{C.13}$$

Therefore, we have

$$\big(\hat{A}(\sigma \otimes 1_d)\big)_{ik} = (\sigma \otimes 1_d)_{i[k]\ell} = \sigma_{i[k]}. \tag{C.14}$$

So

$$\Big(\mathcal{M}_n\big(\hat{A}(\sigma \otimes \frac{1_d}{d})\big)\Big)_i = \frac{1}{d} \cdot \sum_{k=1}^{d} \big(\hat{A}(\sigma \otimes 1_d)\big)_{ik} = \frac{1}{d} \cdot \sum_{k=1}^{d} \sigma_{i[k]}. \tag{C.15}$$

Combining (C.12) and (C.15) then leads to

$$\mathcal{M}_n\big(\hat{A}(\sigma \otimes \frac{1_d}{d})\big) = d \cdot A\sigma, \tag{C.16}$$

as required. ■

Lemma B.1

1. $\forall \sigma \in \mathbb{R}^{nm}, \tau \in \mathbb{R}^n : \langle \sigma, \tau \otimes 1_m \rangle = \langle \mathcal{M}_n(\sigma), \tau \rangle$.
2. $\forall \sigma \in \mathbb{R}^{nd(1)}, \tau \in \mathbb{R}^n : \langle \sigma, \tau \otimes 1_{d_1} \rangle = \langle \mathcal{M}_n(\sigma), \tau \rangle$.
3. $\forall \sigma \in \mathbb{R}^{nd(2)}, \tau \in \mathbb{R}^n : \langle \sigma, \tau \otimes 1_{d_2} \rangle = \langle \mathcal{M}_n(\sigma), \tau \rangle$.

Proof: We will prove only part (1), the proofs of (2) and (3) are analogous.

We index the elements of vectors in the the space $\mathbb{R}^m \otimes \mathbb{R}^d$ with the set $[m] \times [d]$. For any $i \in [n], j \in [d]$ we have

$$(\tau \otimes 1_{d_2})_{ij} = \tau_i \tag{C.17}$$

Now

$$\begin{aligned} \langle \sigma, \tau \otimes 1_{d_2} \rangle &= \textstyle\sum_{i=1}^m \sum_{j=1}^d \sigma_{ij} \cdot (\tau \otimes 1_{d_2})_{ij} \\ &= \textstyle\sum_{i=1}^m \big(\sum_{j=1}^d \sigma_{ij} \big) \cdot \tau_i \qquad \text{(from (C.17))} \\ &= \textstyle\sum_{i=1}^m \big(\mathcal{M}_m(\sigma) \big)_i \cdot \tau_i \\ &= \langle \mathcal{M}_m(\sigma), \tau \rangle. \end{aligned} \tag{C.18}$$

■

Lemma B.2

1. $\forall \sigma \in \mathbb{R}^n : \mathcal{M}_n\big(\overline{A}(\sigma \otimes 1_{d_1})\big) = d_1 \cdot A\sigma$.
2. $\forall \sigma \in \mathbb{R}^m : \mathcal{M}_m\big(\overline{B}(\sigma \otimes 1_{d_2})\big) = d_2 \cdot B\sigma$.
3. $\forall \sigma \in \mathbb{R}^n, \forall \tau \in \mathbb{R}^n : \langle \overline{A}(\sigma \otimes 1_{d_1}), \tau \otimes 1_{d_1} \rangle = d_1 \cdot \langle A\sigma, \tau \rangle$.
4. $\forall \sigma \in \mathbb{R}^m, \forall \tau \in \mathbb{R}^m : \langle \overline{B}(\sigma \otimes 1_{d_2}), \tau \otimes 1_{d_2} \rangle = d_2 \cdot \langle B\sigma, \tau \rangle$.

Proof:

• The proofs of (1) and (2) and analogous to that of Lemma 6.8.

• For (3), let $\sigma \in \mathbb{R}^n$ and $\tau \in \mathbb{R}^n$.

$$\begin{aligned} \langle \overline{A}(\sigma \otimes 1_{d_1}), \tau \otimes 1_{d_1} \rangle &= \langle \mathcal{M}_n\big(\overline{A}(\sigma \otimes 1_{d_1})\big), \tau \rangle \quad \text{(using Lemma B.1 (2))} \\ &= \langle d_1 \cdot A\sigma, \tau \rangle \qquad \text{(using (1))} \\ &= d_1 \cdot \langle A\sigma, \tau \rangle. \end{aligned} \tag{C.19}$$

• the proof of (4) if analogous to that of (3).

■

Lemma B.3

1. $C \cdot 1_{d_1} = \frac{d_1}{d_2} \cdot 1_{d_2}$.
2. $C \cdot 1_{d_2} = \frac{d_2}{d_1} \cdot 1_{d_1}$.
3. $\forall \sigma \in \mathbb{R}^{nm} : \overline{A}(\sigma \otimes 1_d) = \overline{A}(\sigma \otimes 1_{d_1})$.
4. $\forall \sigma \in \mathbb{R}^{nm} : \overline{B}(\sigma \otimes 1_d) = \overline{B}(\sigma \otimes 1_{d_2})$.
5. $\forall \sigma \in \mathbb{R}^n, \tau \in \mathbb{R}^{md} : \mathcal{M}_{nm}(\sigma \otimes \tau) = \sigma \otimes \mathcal{M}_m(\tau)$.

Proof:

• (1) and (2) follow immediately from the fact that the transition matrix of C is in the form

$$\left(\begin{array}{c|c} 0 & \frac{1}{r} \cdot X \\ \hline \frac{1}{\ell} \cdot X^T & 0 \end{array}\right), \tag{C.20}$$

where the rows and columns of X have weight r and ℓ respectively. So

$$C \cdot 1_{d_1} = \frac{r}{\ell} \cdot 1_{d_1} = \frac{d_1}{d_2} \cdot 1_{d_1}. \tag{C.21}$$

• For (3) Recall $\overline{A}$ has vertex set $[n] \times [m] \times [d]$, and that all vertices in $[n] \times [m] \times [d_2]$ are edgeless. Therefore for any $x \in \mathbb{R}^n \otimes \mathbb{R}^m \otimes \mathbb{R}^{d(2)}$ we have $\overline{A}x = 0$. Now

$$\overline{A} \cdot 1_d = \overline{A}\big(1_{d_1} + 1_{d_2}\big) = \overline{A} \cdot 1_{d_1}. \tag{C.22}$$

• The proof of (4) is analogous to that of (3).

• For (5), on the left hand side we have $\mathcal{M}_{nm}(\sigma \otimes \tau) \in \mathbb{R}^n \otimes \mathbb{R}^m$. For any $i \in [n]$, $j \in [m]$

$$\mathcal{M}_{nm}(\sigma \otimes \tau)_{ij} = \sum_{k=1}^{d} \sigma_i \cdot \tau_{jk}. \tag{C.23}$$

On the right hand side, first note that $\mathcal{M}_m(\tau) \in \mathbb{R}^m$ and

$$\big(\mathcal{M}_m(\tau)\big)_j = \sum_{k=1}^{d} \tau_{jk}. \tag{C.24}$$

Now

$$\begin{aligned} \big(\sigma \otimes \mathcal{M}_m(\tau)\big)_{ij} &= \sigma_i \cdot \big(\mathcal{M}_m(\tau)\big)_j \\ &= \sigma_i \cdot \textstyle\sum_{k=1}^{d} \tau_{jk} \\ &= \mathcal{M}_{nm}(\sigma \otimes \tau)_{ij} \quad \text{using (C.23)}, \end{aligned} \tag{C.25}$$

and so (5) follows. ■

List of Symbols

Algebra

$\mathbb{N}$	Natural numbers ($\{0, 1, 2, \ldots\}$)
$\mathbb{N}^*$	Positive natural numbers ($\{1, 2, 3, \ldots\}$)
$\mathbb{Z}$	Ring of integers
$\mathbb{Z}_n$	Ring of integers modulo n ($\mathbb{Z}/n\mathbb{Z}$)
$\mathbb{Q}$	Field of rationals
$\mathbb{R}$	Field of real numbers
$\mathbb{R}_{\geq 0}$	Set of non-negative real numbers
$\mathbb{R}_{>0}$	Set of positive real numbers
$\mathbb{C}$	Field of complex numbers
$\mathbb{F}_q$	Finite field of size q
C_ℓ	Cyclic group of size ℓ
D_ℓ	DFT matrix corresponding to C_ℓ
$[n]$	The set $\{1, \ldots, n\}$
$\lceil x \rceil$	Smallest integer not smaller than x
$\lfloor x \rfloor$	Largest integer not larger than x
$S \sqcup T$	Disjoint union of the sets S and T

Linear Algebra

$\langle x, y \rangle$	Inner product of the vectors x and y
$x \otimes y$	Tensor product of the vectors x and y
$\|x\|$	Norm of vector x
$x \parallel y$	Vectors x and y are parallel
$x \perp y$	Vectors x and y are orthogonal
1_n	The vector in $\mathbb{R}^n$ whose entries are all 1
$1_n^{\parallel}$	Space of vectors in $\mathbb{R}^n$ generated by 1_n
$1_n^{\perp}$	Space of vectors in $\mathbb{R}^n$ that are orthogonal to 1_n
$R^{n \times m}$	The set of all $n \times m$ matrices with entries in the ring R
lker(M)	Left kernel of the matrix M
rker(M)	Right kernel of the matrix M

Algebraic Geometry

$\mathcal{X}$	Algebraic curve
$\mathbb{D}(\mathcal{X})$	Divisor group of the curve $\mathcal{X}$
$\mathbb{D}^0(\mathcal{X})$	Group of divisors of degree 0
Prin$(\mathcal{X})$	Group of principal divisors
Pic$(\mathcal{X})$	$\mathbb{D}(\mathcal{X})$ modulo Prin$(\mathcal{X})$
Pic$^0(\mathcal{X})$	$\mathbb{D}^0(\mathcal{X})$ modulo Prin$(\mathcal{X})$
D	Divisor (element of $\mathbb{D}(\mathcal{X})$)
$\mathcal{L}(D)$	Linear space of the divisor D
$\dim(D)$	Dimension of the divisor D
$\deg(D)$	Degree of the divisor D
$N_q(g)$	Maximum number of points on a curve over $\mathbb{F}_q$ of genus g

Coding Theory

$\mathcal{C}$	Code
wgt(x)	Hamming weight of the vector x
d(x, y)	Hamming distance between the vectors x and y
$B_r(x)$	Ball of radius r around the vector x
Vol(r, n)	Volume of a ball of radius r in $\mathbb{F}_2^n$
$d_{\min}(\mathcal{C})$	Minimum distance of the code $\mathcal{C}$
$\delta(\mathcal{C})$	Relative distance of the code $\mathcal{C}$
$\dim(\mathcal{C})$	Dimension of the code $\mathcal{C}$
$R(\mathcal{C})$	Rate of the code $\mathcal{C}$
$\{\mathcal{C}_i\}_{i\in\mathbb{N}^*}$	Family of codes
h_q	q-ary entropy function
h	Binary entropy function

Graph Theory

K_d	Complete graph on d vertices
$N(a)$	Set of neighbors of the vertex a
$N(S)$	Set of neighbors of the set of vertices S
$a[i]$	i^{th} neighbor of vertex a
λ_A	Second eigenvalue of the graph A
P$_n[A]$	Projection of size n of the graph A
DP$[A]$	De-projection of the graph A
A^2	Square of the graph A
$A Ⓢ C$	Derandomized square of the graph A with respect to the graph C
$A \otimes B$	Tensor product of the graphs A and B
$A \otimes_C B$	Derandomized tensor product of the graphs A and B with respect to the graph C
$A Ⓩ B$	Zig-zag product of the graphs A and B

Bibliography

[1] A. Albanese, J. Blömer, J. Edmonds, M. Luby, M. Sudan. "Priority Encoding Transmission". *IEEE Trans. on Inform. Theory*, vol. 42, pp. 1737–1744, 1996.

[2] N. Alon. "Eigenvalues and Expanders". *Combinatorica*, vol. 6, no. 2, pp. 83–96, 1986.

[3] N. Alon, F. Chung. "Explicit Constructions of Linear Sized Tolerant Networks". *Discr. Math.*, vol. 72, pp. 15–19, 1988.

[4] N. Alon, A. Lubotzky, A. Wigderson. "Semi-Direct Product in Groups and Zigzag Product in Graphs: Connections and Applications". *Proc. 42nd IEEE Symp. on Foundations of Comput. Sci. (FOCS)*, Las Vegas, Nevada, pp. 630–637, 2001.

[5] N. Alon, V. Milman, "Eigenvalues, Expanders and Superconcentrators". *Proc. 25th IEEE Symp. on Foundations of Comput. Sci. (FOCS)*, Singer Island, Florida, pp. 320–322, 1984.

[6] N. Alon, V. Milman, "λ_1-Isopermitric Inequalities for Graphs, and Superconcentrators". *J. Combin. Theory Ser. B*, vol. 38, no. 1, pp. 73–88, 1985.

[7] M. Artin. *Algebra*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1991.

[8] N. Biggs, *Algebraic Graph Theory*, 2nd ed. Cambridge: Cambridge University Press, 1993.

[9] E. Blokh, V. Zyablov. "Linear Concatenated Codes". *Nauka*, Moscow, 1982.

[10] J. Blömer, M. Kalfane, R. Karp, M. Karpinski, M. Luby, D. Zuckerman. "An XOR-Based Erasure-Resilient Coding Scheme". *Technical Report TR-95-048*, International Computer Science Institute, 1995.

[11] A. Brouwer. "Bounds on the Minimum Distance of Linear Codes". `http://www.win.tue.nl/~aeb/voorlincod.html`.

[12] A. Brown, A. Shokrollahi. "Algebraic-Geometric Codes Over the Erasure Channel". *Proc. IEEE Intern. Symp. on Inform. Theory*, Chicago, p. 76, 2004.

[13] A. Brown, L. Minder, A. Shokrollahi. "Probabilistic Decoding of Interleaved Reed-Solomon Codes on the Q-ary Symmetric Channel". *Proc. IEEE Intern. Symp. on Inform. Theory*, Chicago, p. 327, 2004.

[14] A. Brown, M. Luby, A. Shokrollahi. "Repeat-Accumulate Codes that Approach the Gilbert-Varshamov Bound". *Proc. IEEE Intern. Symp. on Inform. Theory*, Adelaide, Australia, pp. 169–173, 2005.

[15] A. Brown, L. Minder, A. Shokrollahi. "Improved Decoding of Interleaved AG-Codes". *Proc. 10th IMA Conf. on Cryptography and Coding*, Cirencester, UK, Lecture Notes in Comput. Sci., Vol. 3796, pp. 37–46, 2005 .

[16] A. Brown, A. Shokrollahi. "Some Graph Products and their Expansion Properties". *Proc. IEEE Inform. Theory Workshop*, Punta del Este, Uruguay, pp. 132–134, 2006.

[17] P. Bürgisser, M. Clausen, A. Shokrollahi. *Algebraic Complexity Theory*. Springer-Verlag, 1997.

[18] P. Cameron, J. van Lint. *Designs, Graphs, Codes and their Links*. Cambridge: Cambridge University Press, 1991.

[19] J. Canon et al. "The Computer Algebra System Magma", 2003. `http://magma.maths.usyd.edu.au/magma`

[20] F. Chung. *Spectral Graph Theory*, CBMS Regional Conference Series in Mathematics, AMS, vol. 92, 1994.

[21] S-Y. Chung, D. Forney, T. Richardson, R. Urbanke. "On the Design of Low-Density Parity-Check Codes Within 0.0045 dB of the Shannon Limit". *IEEE Comm. Letters*, vol. 5, no. 2, pp. 58–60, 2001.

[22] D. Divsalar, H. Jin, R. McEliece. "Coding Theorems for 'Turbo-Like' Codes". *Proc. 36th Allerton Conf. on Communications, Control, and Computing*, Allerton House, Monticello, Illinois, pp. 201–210, 1998.

[23] J. Dodziuk. "Difference Equations, Isoperimetric Inequality and Transience of Certain Random Walks". *Trans. Amer. Math. Soc.*, vol. 284, no. 2, pp. 787–794, 1984.

[24] I. Dumer. "Concatenated Codes and their Multilevel Generalizations". *Handbook of Coding Theory*, vol. 2, V. Pless and W. Huffman, Editors, Amsterdam: Elsevier Science, pp. 1911–1988, 1998.

[25] G. Forney. *Concatenated Codes*. M.I.T. Press, Cambridge, MA, 1966.

[26] J. Friedman. "Some Geometric Aspects of Graphs and their Eigenfunctions". *Duke Math. J.*, vol. 69, no. 3, pp. 487–525, 1993.

[27] O. Gabber, Z. Galil. "Explicit Constructions of Linear-Sized Superconcentrators". *J. of Comput. and System Sci.*, vol. 22, no. 3, pp. 407–420, 1981.

[28] V. Goppa. "Codes on Algebraic Curves". *Soviet Math Dokl.*, vol. 24, no. 1, pp.170–172, 1988.

[29] V. Goppa. *Geometry and Codes*. Kluwer Academic Publishers, 1988.

[30] R. Hamming. "Error Detecting and Error Correcting Codes". *Bell Syst. Tech. J.*, vol. 29, pp. 147–160, 1950.

[31] F. Heß. "Computing Riemann-Roch Spaces in Algebraic Function Fields and Related Topics". *J. Symbolic Comput.*, vol. 11, 2001.

[32] I. Holyer. "The NP-Completeness of Edge Colorings". *SIAM J. Comput.*, vol. 10, 718-720, 1981.

[33] S. Hoory, N. Linial, A. Wigerson. "Expander graphs and their Applications". *Bulletin of the American Mathematical Society*, vol. 43, no. 4, pp. 439–561, 2006.

[34] G. James, M. Liebeck. *Representations and Characters of Groups*. Cambridge: Cambridge University Press, 1993.

[35] H. Jin, A. Khandekar, R. McEliece. "Irregular Repeat-Accumulate Codes". *Proc. 2nd Intern. Conf. on Turbo Codes and Related Topics*, Brest, France, pp. 1–8, 2000.

[36] S. Jukna. *Extremal Combinatorics: With Applications in Computer Science*. Springer-Verlag, 2001.

[37] J. Justesen. "A Class of Constructive Asymptotically Good Algebraic Codes". *IEEE Trans. on Inform. Theory*, vol. 18, pp. 652–656, 1972.

[38] G. Katsman, M. Tsfasman S. Vladut. "Modular Curves and Codes with a Polynomial Construction". *IEEE Trans. on Inform. Theory*, vol. 30, pp. 353–355, 1984.

[39] S. Krantz. "The Euler-Mascheroni Constant". *Handbook of Complex Variables*, 13.1.7, Boston: Birkhäuser, pp. 156–157, 1999.

[40] D. Leven, Z. Galil. "NP-completeness of Finding the Chromatic Index of Regular Graphs". *J. Algorithms*, vol. 4, pp. 35–44, 1983.

[41] S. Lin, D. Costello. *Error Control Coding: Fundamentals and Applications*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1983.

[42] S. Litsyn, V. Shevelev. "On Ensembles of Low-Density Parity-Check Codes: Distance Distributions". *IEEE Trans. on Inform. Theory*, vol. 49, pp. 3140–3159, 2003.

[43] A. Lubotzky, R. Phillips, P. Sarnak. "Explicit Expanders and the Ramanujan Conjectures". *Proc. 18th ACM Symp. on Theory of Comput.*, Berkeley, California, pp. 240–246, 1986.

[44] A. Lubotzky, R. Phillips, P. Sarnak. "Ramanujan Graphs". *Combinatorica*, vol 8, no. 3, pp. 261–267, 1988.

[45] M. Luby. "LT-Codes". *Proc. 43rd IEEE Symp. on Foundations of Comput. Sci. (FOCS)*, Vancouver, Canada, pp. 271–280, 2002.

[46] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. Spielman. "Efficient Erasure Correcting Codes". *IEEE Trans. on Inform. Theory*, vol. 47, pp. 569–584, 2001.

[47] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. Spielman, V. Stenman. "Practical Erasure Resilient Codes". *Proc. 29th ACM Symp. on Theory of Computing (STOC)*, pp.150–159, 1997.

[48] D. MacKay. "Good Error-Correcting Codes Based on Very Sparse Matrices". *IEEE Trans. on Inform. Theory*, vol. IT-45, no. 2, pp. 399–131, 1999.

[49] D. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge: Cambridge University Press, 2003. Available at `http://www.inference.phy.cam.ac.uk/mackay/itila/`

[50] D. MacKay, M. Davey. "Low-Density Parity Check Codes over GF(q)". *IEEE Comm. Letters*, vol. 2, no. 6, pp. 165–167, 1998.

[51] F. MacWilliams, N. Sloane. *The Theory of Error-Correcting Codes*, New York: North-Holland, 1977.

[52] G. Margulis. “Explicit Constructions of Expanders”. *Problemy Peredachi Informatsii*, vol. 9, no. 4, pp. 73–80, 1973.

[53] G. Margulis. “Explicit Constructions of Graphs Without Short Cycles and Low Density Codes”. *Combinatorica*, vol. 2, no. 1, pp. 71–78, 1982.

[54] G. Margulis. “Explicit Group-Theoretical Constructions of Combinatorial Schemes and their Application to the Design of Expanders and Superconcentrators”. *Problemy Peredachi Informatsii*, vol. 24, no. 1, pp. 39–46, 1988.

[55] O. Moreno, D. Zinoviev, V. Zinoviev. “On Several New Projective Curves Over $\mathbb{F}_2$ of Genus 3, 4 and 5”. *IEEE Trans. on Inform. Theory*, vol. 41, pp. 1643–1645, 1995.

[56] M. Morgenstern. “Existence and Explicit Constructions of q+1 Regular Ramanujan Graphs for Every Prime Power q”. *J. Combin. Theory Ser. B*, vol. 62, pp. 44–32, 1994.

[57] A. Nilli. “On the Second Eigenvalue of a Graph”. *Discrete Math.*, vol. 91, no. 2, pp. 207–210, 1991.

[58] P. Oswald, A. Shokrollahi. “Capacity-Achieving Sequences for the Erasure Channel”. *IEEE Trans. on Inform. Theory*, vol. 48, pp. 3017–3028, 2002.

[59] M. Pinsker. “On the Complexity of a Concentrator”. *7th Intern. Teletraffic Conf.*, Stockholm, Sweden, p. 318, 1973.

[60] O. Pretzel. *Error-Correcting Codes and Finite Fields*. Clarendon Press, Oxford, 1992.

[61] O. Reingold, S. Vadhan, A. Wigderson. “Entropy Waves, the Zig-Zag Product, and New Constant Degree Expanders”. *Ann. of Math.*, vol. 155, no. 1, pp. 157–187, 2002.

[62] T. Richardson, A. Shokrollahi, R. Urbanke. “Finite-Length Analysis of Various Low-Density Parity-Check Ensembles For the Binary Erasure Channel”. *Proc. IEEE Intern. Symp. on Inform. Theory*, Lausanne, Switzerland, p.1, 2002.

[63] L. Rizzo. “On the Feasibility of Software FEC”. 1997.
`http://info.iet.unipi.it/ luigi/softfec.ps`

[64] E. Rozenman, S. Vadhan. “Derandomized Squaring of Graphs”. *Proc. 8th Intern. Workshop on Randomization and Computation (RANDOM ‘05)*, Berkeley, Lecture Notes in Comput. Sci., vol. 3624, pp. 436–447, 2005.

[65] E. Rozenman, S. Vadhan. “Derandomized Squaring of Graphs”. *Electronic Colloquium on Computational Complexity*, Technical Report TR05-092, 2005.

[66] B. Segre. “Introduction to Galois Geometries”. *Atti Acad. Naz. Lincei (Mem. Cl. Sci. Fis. Mat. Natur.)*, vol. 8, pp. 133–236, 1967.

[67] J-P. Serre. “Nombre de Points des Courbes Algébriques sur $\mathbb{F}_q$”. *Séminaire de Théorie des Nombres de Bordeaux*, exp. no. 22 (*Oeuvres III*, no. 129, pp. 664–668), 1982-1983.

[68] J-P. Serre. “Quel est le Nombre Maximum de Points Rationnels que peut avoir une Courbe Algébrique de Genre g sur un Corps Fini $\mathbb{F}_q$?”. *Résumé des Cours de 1983-1984* (*Oeuvres III*, no. 132, pp. 701–705), 1983-1984.

[69] J-P. Serre. “Rational Points on Curves over Finite Fields”. *Notes of lectures at Harvard University*, 1985.

[70] J. Silverman. *The Arithmetic of Elliptic Curves*. Graduate Texts in Mathematics, vol. 106, Springer-Verlag, 1986.

[71] C. Shannon. “A Mathematical Theory of Communication”. *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 623–656, 1948.

[72] S. Shokranian, A. Shokrollahi. “Coding Theory and Bilinear Complexity”. *International Office of the Forschungszentrum*, 1993.

[73] A. Shokrollahi. “Codes on Hermitian Curves”. *Proc. 4th Intern. Symp. on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, Lecture Notes in Comput. Sci., vol. 507, pp. 168–176, 1987.

[74] A. Shokrollahi. “Minimum Distance of Elliptic Codes”. *Advances in Mathematics*, vol. 93, pp. 251–281, 1992.

[75] A. Shokrollahi. “New Sequences of Linear Time Erasure Codes Approaching the Channel Capacity”. *Proc. 13th Intern. Symp. on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, Honolulu, Lecture Notes in Comput. Sci., vol. 1719, pp. 65–76, 1999.

[76] A. Shokrollahi. “Raptor Codes”. *IEEE Trans. on Inform. Theory*, vol. 52, no. 6, pp. 2551–2567, 2006.

[77] A. Shokrollahi, H. Wasserman. “List Decoding of Algebraic Geometric Codes”. *IEEE Trans. on Inform. Theory*, vol. 45, no. 2, pp. 432–437, 1999.

[78] A. Shokrollahi. “Modern Coding Theory”. *EPFL Doctoral school course*, 2006. `http://algo.epfl.ch/index.php?p=courses_0506_MCT`

[79] M. Sipser, D. Spielman. “Expander Codes”. *IEEE Trans. on Inform. Theory*, vol. 42, no. 6, pp. 1710–1722, 1996.

[80] D. Spielman. “Linear Time Encodable and Decodable Error Correcting Codes”. *IEEE Trans. on Inform. Theory*, vol. 42, no. 6, pp. 1723–1731, 1996.

[81] H. Stichtenoth. *Algebraic Function Fields and Codes*. Springer-Verlag, 1993.

[82] M. Sudan. “Algorithmic Introduction to Coding Theory”. *MIT Graduate course*, 2002. `http://theory.lcs.mit.edu/~madhu/FT01/course.html`.

[83] M. Tanner. “A Recursive Approach to Low Complexity Codes”. *IEEE Trans. on Inform. Theory*, vol. 27, no. 5, pp. 1710–1722, 1981.

[84] M. Tanner. “Explicit Construction of Concentrators from Generalized N-gons”. *SIAM J. on Algebraic Discrete Methods*, vol. 5, no. 3, pp. 287–293, 1984.

[85] M. Tanner. “Minimum Distance Bounds by Graph Analysis”. *IEEE Trans. on Inform. Theory*, vol. 47, no. 2, pp. 808–821, 2001.

[86] M. Tsfasman, S. Vladut. *Algebraic-Geometric Codes*. Kluwer, Dordrecht, 1991.

[87] M. Tsfasman, S. Vladut, T. Zink. “Modular Curves, Shimura Curves, and Goppa Codes, Better than the Varshamov-Gilbert Bound”. *Math. Nachr.*, vol. 109, pp. 21–28, 1982.

[88] G. van der Geer. “Tables of Curves with Many Points”. `http://www.science.uva.nl/~geer`.

[89] J. van Lint. *Introduction to Coding Theory*, 3rd ed. Graduate Texts in Mathematics, vol. 86, Springer-Verlag, 1998.

[90] V. Vizing. “On an Estimate of the Chromatic Class of a Graph”. *Diskret. Anal.*, vol. 3, p. 2530, 1964.

[91] E. Weisstein. “Gamma Function”. *MathWorld–A Wolfram Web Resource*. `http://mathworld.wolfram.com/GammaFunction.html`.

[92] E. Weisstein. “Symmetric Polynomial”. *MathWorld–A Wolfram Web Resource*. `http://mathworld.wolfram.com/SymmetricPolynomial.html` .

[93] E. Weisstein. “Euler-Mascheroni Constant”. *MathWorld–A Wolfram Web Resource*. `http://mathworld.wolfram.com/Euler-MascheroniConstant.html`.

[94] D. Xiao. “The Evolution of Expander Graphs”. AB thesis, Harvard College, 2003.

[95] V. Zyablov. “An Estimate of the Complexity of Constructing Binary Linear Cascaded Codes”. *Problemy Peridachi Informatsii*, vol 15, no. 2, pp. 58–70, 1971.

[96] V. Zyablov, M. Pinsker. “Estimation of Error-Correction Complexity of Gallager Low-Density Codes”. *Problemy Peridachi Informatsii*, vol. 11, no. 1, pp. 18–28, 1976.

Andrew Brown

Address:	22, Chemin du Mottey 1020 Renens Switzerland	**Nationality:** **Phone:** **E-mail:**	British / US +41 79 486 4910 andrew.brown@epfl.ch

Education

2003-present	**Ecole polytechnique Fédérale de Lausanne (EPFL)** **PhD in Mathematics** Thesis: “Codes, graphs and graph-based codes”.	**Lausanne, Switzerland**
2001-2002	**University College, University of Oxford** **Master of Science, “Mathematics and the Foundations of Computer Science”** Dissertation: Computational algebra, “Efficient Presentations for some Finite Simple Groups”.	**Oxford, UK**
1997-2001	**Imperial College of Science, Technology and Medicine** **MSci, Mathematics and Computing, 1st class degree** Final year project: “Minimal Degrees of Primitive Permutation Groups”. Awarded best individual project prize.	**London, UK**

Awards

2001	**Morgan Stanley prize in Joint Mathematics & Computing** “Awarded to a final year student on the course for the best individual project”.	**Imperial College**
2000	**Joint Mathematics & Computing non-final year prize** “Awarded to the outstanding student for excellence in a non-final year of the course”.	**Imperial College**

Work experience (summer internships)

2005	**Bell Labs, Lucent Technologies** Summer intern in the Mathematics Center. Research on expander graphs and network coding.	**Murray Hill (NJ), USA**
2004	**Digital Fountain, Inc. (Licenses error correction technology)** Summer intern. • Worked on the development, analysis and specification of the Raptor version 10 coding technology. • Constructed Algebraic-Geometric codes and implemented (in C++) a fast encoder/decoder. Also developed a method to compute the error probabilities of these codes.	**Fremont (CA), USA**
2001	**Food For Thought (Food and drinks consultancy)** Worked on a service enabling customers to purchase data over the web.	**Geneva, Switzerland**
2000	**CERN (European Laboratory for Particle Physics)** Worked in a team of programmers developing web based tools used throughout CERN. Required good knowledge of Java and the Oracle database management system.	**Geneva, Switzerland**