The Approximability of Learning and Constraint Satisfaction Problems

Yi Wu

CMU-CS-10-142

October 7, 2010

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Ryan O'Donnell, Chair
Avrim Blum
Venkatesan Guruswami
Subhash Khot

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

This thesis is dedicated to my parents Youqun Yu, Qingchun Wu and my wife Xiaoxiao Li.

Abstract

An α-approximation algorithm is an algorithm guaranteed to output a solution that is within an α ratio of the optimal solution. We are interested in the following question: Given an NP-hard optimization problem, what is the best approximation guarantee that any *polynomial time* algorithm could achieve?

We mostly focus on studying the *approximability* of two classes of NP-hard problems: Constraint Satisfaction Problems (CSPs) and Computational Learning Problems.

For CSPs, we mainly study the approximability of MAX CUT, MAX 3-CSP, MAX 2-LIN$_\mathbb{R}$, VERTEX-PRICING, as well as serval variants of the UNIQUE-GAMES.

- The problem of MAX CUT is to find a partition of a graph so as to maximize the number of edges between the two partitions. Assuming the Unique Games Conjecture, we give a complete characterization of the approximation curve of the MAX CUT problem: for every optimum value of the instance, we show that certain SDP algorithm with RPR^2 rounding always achieve the optimal approximation curve.

- The input to a 3-CSP is a set of Boolean constraints such that each constraint contains at most 3 Boolean variables. The goal is to find an assignment to these variables to maximize the number of satisfied constraints. We are interested in the case when a 3-CSP is satisfiable, i.e., there does exist an assignment that satisfies every constraint. Assuming the d-to-1 conjecture (a variant of the Unique Games Conjecture), we prove that it is NP-hard to give a better than 5/8-approximation for the problem. Such a result matches a SDP algorithm by Zwick which gives a 5/8-approximation problem for satisfiable 3-CSP. In addition, our result also conditionally resolves a fundamental open problem in PCP theory on the optimal soundness for a 3-query nonadaptive PCP system for NP with perfect completeness.

- The problem of MAX 2-LIN$_\mathbb{Z}$ involves a linear systems of integer equations; these equations are so simple such that each equation contains at most 2 variables. The goal is to find an assignment to the variables so as to maximize the total number of satisfied equations. It is a natural generalization of the Unique Games Conjecture which address the hardness of the same equation systems over finite fields. We show that assuming the Unique Games Conjecture, for a MAX 2-LIN$_\mathbb{Z}$ instance, even that there exists a solution that satisfies $1 - \epsilon$ of the equations, it is NP-hard to find one that satisfies ϵ of the equations for any $\epsilon > 0$.

- The problem of VERTEX-PRICING involves of a set of customers each of which is interested in buying a set of items. VERTEX-PRICING$_k$ is the special case when each customer is interested in at most k of the items. All of the buyers are single minded, which means that each of the buyer would buy all the items they have interest on if the total cost of the items is within their budget. The algorithmic task is to price each item with so as to maximize the overall profit. When each item is priced positive profit margin, it is known that there is a $O(k)$-approximation algorithm for the problem. We prove that in contrast for the very simple case of VERTEX-PRICING$_3$, when the seller is allowed to price some of the items with negative profit margin (in which case more profit could possibly be achieved), there is no polynomial time approximation algorithm that gives constant approximation to the problem assuming the Unique Games Conjecture.

For the learning problems, our results mostly involve showing that learning tasks are hard for many basic function classes under the agnostic learning model. In particular, we proved that the following two results on agnostic learning monomials and low degree polynomial threshold functions:

- Our first result is about hardness of learning monomials. We prove that given a set of examples, even that there exists a monomial that is consistent with $1 - \epsilon$ of the examples, it is NP-hard to find a $(1/2 + \epsilon)$ good hypothesis even we are allowed to output a linear threshold function, for any $\epsilon > 0$. Our result rules out the possibility of using linear classifiers such as Winnow and SVM to agnostically learn monomials.

- Our second result is on the hardness of learning polynomial threshold functions (PTFs). We prove that assuming the Unique Games Conjecture, given a set of examples, even that there exists a low degree PTF that is consistent with $1 - \epsilon$ of the examples, it is NP-hard to find such a one that is $1/2 + \epsilon$ good for any $\epsilon > 0$. In the language of learning, we show there is no better-than-trivial proper learning algorithm that agnostically learns low degree PTFs.

Contents

10 Hardness of Learning Low degree PTFs   203

IV   Open Problems   225

11 Open Problems   227

List of Figures

List of Tables

Part I

Introduction

Chapter 1

Overview of the Thesis

1.1 Motivation

For a vast variety of applications in computer science and engineering, the central task is to design *efficient* algorithms for certain *optimization* problem. For example, in machine learning, one of the major goal is to find a predication rule with the *maximum* accuracy on a particular domain of data; in computer networking, a common task is to design a protocol that gives the *minimum* delay of the transmissions.

Unfortunately, for a huge class of optimization problems, it is NP-hard to find the optimum solution. Under the widely held belief that $P \neq NP$, there does not exist a polynomial time algorithm for all of these NP-hard optimization problems.

To cope with the NP-hardness, there has been a great interest of designing efficient *approximation algorithms* that return a suboptimal solution provably close to the optimum. Formally, an algorithm is called an α-approximation if it guarantees to output a solution that is within a factor α of the optimum. When $\alpha = 1$, the algorithm solves the problem exactly. Ideally, we want to design an algorithm with its approximation ratio α being as close to 1 as possible, while still require the algorithm to have an polynomial running time. This raises the following natural question:

Question *Given an* NP-*hard problem, what is the best polynomial time approximation algorithm?*

Answering the above question involves proofs from two sides: first we need to exhibit a polynomial time algorithm that has certain approximation guarantee; second we need to prove the impossibility of getting better polynomial time approximation algorithms.

This thesis is about to study the optimal approximation threshold for a variety of important and natural NP-hard optimization problems.

1.2 Problems Studied in This Thesis

To give the reader a sense of the optimization problems studied in the thesis, we list some of them here:

1. (MAX 2-LIN$_{\mathbb{Z}}$) We are given a set of linear inequations and these equations are so simple such that each equation contains at most 2 variables. Can we find an assignment to the variables so as to maximize the number of satisfied equations?

2. (MON-MA) We want to decide whether an E-mail is spam or not. A common approach is to look at whether these E-mails contain certain set of key words or not. Suppose there is a collection of key words such that with high accuracy, E-mails containing all of them are spam (and vice versa). Given a set of E-mails that are labelled with whether they are spam or not, can we find a way to classify other unlabelled E-mails with high accuracy?

3. (MAX CUT) Given a graph, can we partition it into two parts so as to maximize the total number of edges between them?

4. (MAX 3-CSP) Given a set of Boolean constraints such that each of the constraint contains at most 3 variables, can we efficiently find a solution that satisfies all of them (if there exists such a solution)?

5. (VERTEX-PRICING) A set of buyers each of which is interested in a bundle of items. These buyers are single minded such that they either buy the whole bundle if the total cost is within their budget or they will buy nothing. The question is how to price each item so as to maximize the overall profit.

Generally speaking, the problems studied in this thesis come from the following two categories: i) Constraint Satisfaction Problem (CSP); ii) Computational Learning. Below, we give a high level overview of these two classes of problems.

1.2.1   Constraint Satisfaction Problem (CSP)

Briefly speaking, a *Constraint Satisfaction Problem (CSP)* involves a system of constraints on a set of variables. Given a CSP, the natural algorithmic task, called "Max-CSP", is to find an assignment to the variables such that the total number of satisfied constraints are as large as possible.

While the above definition of CSPs is rather abstract, many natural optimization problems fall into the class of CSPs. One concrete example of a CSP is the linear equation system, which consists of a set of linear equations over a set of variables. The corresponding optimization problem is to find an assignment of the variables of the system to satisfy as many equations as possible (if not all). In addition to linear systems, we can specialize a MAX CSP by using other types of constraints to get many of the most canonical NP-hard optimization problems such as MAX CUT, MAX 3-SAT and MAX SAT.

CSPs also have a deep root in the study of theoretical computer science. The NP-hardness of MAX CUT and Max-3SAT came along with the very beginning of the NP-completeness theory [34, 88] in the seventies. Shortly after that, a seminal paper by Johnson [82], which is a foundational paper of the filed of approximation algorithms, designed algorithms for many NP-hard optimization problems including Max-SAT, Max-3SAT as well as Set Cover, Coloring and Maximum Independent Set. Since then, there has been a flurry of work that successfully designing approximation algorithms for various CSPs. Many of the early algorithms are based on Linear Programming; in a breakthrough on both theory and practice happened in 1994, Goemans and Williamson [59] gave a Semidefinite Programming (SDP) rounding algorithm achieving a 0.878 approximation guarantee for MAX CUT; it is the first algorithm with a nontrivial approximation for MAX CUT. After that, there is a tremendous interest in designing SDP based approximation algorithm for various CSPs [11, 15, 29, 32, 37, 49, 110, 123, 141, 142].

Compared with the quick development at the algorithm side, there has been a relatively slow progress on proving hardness of approximation results until the early nineties. The first major breakthrough is the celebrated PCP theorem, which is equivalent to the following statement: there exists some constant $\epsilon > 0$ such that given a 3-SAT instance that can be satisfied by some assignment, it is NP-hard to find a assignment that satisfies $1 - \epsilon$ fraction of the constraints. This implies that it is NP-hard to have a approxima-

tion better than $1 - \epsilon$ for MAX 3-SAT. Since then, people obtain many improved hardness results for various kinds of CSPs. In a seminal work by Hastad [76], he improved the hardness of approximation ratio of MAX 3-SAT from $1 - \epsilon$ to $\frac{3}{4}$. In the same work, he gave a lot of other inapproximability results which included showing that MAX 3-LIN$_q$ is hard to approximate beyond the trivial $1/q$ ratio.

We now have optimal (i.e., matching) approximation algorithms and NP-hardness-of-approximation results for many key problems:MAX k-LIN$_q$ for $k \geq 3$ [76], MAX 3-SAT [76, 87, 143], and a few other MAX k-CSP problems with $k \geq 3$ [65, 76, 138, 141]. However, many basic problems remain unresolved; for example, we do not know if 90%-approximating MAX CUT is in P or is NP-hard. Similarly, given a satisfiable 3-CSP, we do not know if satisfying 2/3 of the constraint-weight is in P or is NP-hard. To address this, the Unique Games Conjecture, along with some variants of it called d-to-1 Conjecture, were proposed by Khot [97] in 2002.

One equivalent statement of the Unqiue Games Conjecture (UGC) [99] is about the approximability of the following problem:

Definition 1.2.1. (Γ-MAX 2-LIN$_q$) *We are given a system of linear equations with variables* $\{x_i\}_{i=1}^n$ *and all the equations are of the simple form* $x_i - x_j = c_{ij} (mod\, q)$ *with the integer coefficient* $0 \leq c_{ij} \leq q - 1$. *The goal is to assign each* x_i *some value in* $\{0, 1, \ldots, q - 1\}$ *such that the maximum number of equations are satisfied.*

UGC states that it is extremely hard to approximate the Γ-MAX 2-LIN$_q$ problem in the following sense:

Conjecture 1.2.2. *(UGC) For any* $\epsilon > 0$, *there exists large enough* q *such that for* Γ-MAX 2-LIN$_q$ *instance, even there is an assignment that satisfies* $1 - \epsilon$ *fraction of the equations, it is* NP-*hard to find an assignment that satisfies more than* ϵ *fraction of the equations.*

Assuming the UGC, people have proved many optimal hardness of approximation results such as those results for MAX CUT [99, 102, 120] and Max-2Sat [13, 14] and a lot of other problems [43, 67, 103]. In a powerful work by Raghvendra [125], he obtained a very general result that for almost any CSP, the optimal approximation is achieved by certain generic SDP algorithms.

This thesis includes work that initializes this line of research as well as work that reflects the latest development of the area. In particular, we study the approximability of serval important CSPs: MAX CUT and MAX 3-CSP and VERTEX-PRICING. In addition, we will study the approximability as well as the SDP approximation for serval variants of the Γ-MAX 2-LIN$_q$ problem.

1.2.2   Computational Learning

In addition to the CSPs, we also study NP-hard optimization problems from *Computational Learning theory*, a branch of theoretical computer science that studies how to efficiently infer an unknown target function from examples under certain distributions. For example, the target function can be "whether it is going to rain tomorrow ?" and the input to the target function could be the measurement of different physical conditions of today such as temperature, humidity, and wind speed, etc. The learning algorithm has an ac-

cess to a set of labelled examples (e.g., the measurements of a certain day as well as the weather of the day after) and the goal is to infer the target function with high accuracy.

We usually assume that the target function has certain "simple" structure, as otherwise we have no way of inferring the function on any unseen examples. Some examples of classes of simple functions include: monomials (conjunctions), decision lists, majority functions, halfspaces, low degree polynomial threshold functions (PTFs), small size decision trees, DNF, CNF, Neural Networks, *et al.* .

In learning theory, researchers are mainly interested in whether these simple function classes are *learnable*. The *learnability* of a function class is defined by whether we can use a small amount of labelled examples and computation time to find a function which has a good agreement with the target function in that function class. Such a model is formalized as the PAC learning model by Valiant [140]. While the original PAC learning model assumes that certain simple target function correctly labels all the data, this model has been generalized by Haussler [78] and Kearns [90] to address the case when there is noise in the labels and examples. Under their model (which is called *agnostic learning model*), it is only known that there is some simple function that has correctly labeled a c (say $c = 0.95$) fraction of the examples, the goal of the learning to come up with a hypothesis with accuracy being close c.

All these learning problems can be viewed as an *optimization* task as we are given a set of labelled examples and our goal is to find a hypothesis with *maximum* prediction accuracy. For many important concept classes, finding the optimal hypothesis, especially when there is noise, is NP-hard. A good learning algorithm usually returns a hypothesis that *approximate* the optimal one well. In the thesis, we are particular interested in the learnability (approximability) of three common function classes: monomials, halfspaces and polynomial threshold functions under the agnostic learning model.

Comparison between Learning and Constraint Satisfaction Problems A learning problem can also be viewed as a CSP: each of the example is a constraint and the goal is to find a hypothesis, specified by a set of variables, that has the maximum agreement with all the examples.

Although the learning problems is a special CSP, in this thesis we discuss the learning problems and other CSPs seperately for the following two reasons: i) the CSPs (except for the learning problems) in this thesis all have "local" constants ; i.e., each constraint involves constant number of variables, while the constraints in the learning problems are "global', which involve many variables. ii) The techniques of proving upper and lower bounds for these two classes of problems are relatively independent.

1.3   Organization and Summary

1.3.1   Organization

While the rest of the thesis spans a variety of different problems, they are all united by the theme on understanding the *approximability* or *inapproximability* of NP-hard optimiza-

tion Problems in Learning and CSPs. The thesis are organized as follows:

In Chapter 2, we define the problems to study in thesis as well as some relevant background knowledge. In Chapter 3, we define the mathematical tools used throughout the thesis. The remaining of the thesis divides into two relatively independent parts.

In Part II, our work is mainly on understanding the approximability of various CSPs as well as the SDP algorithms for them: we study the problem of MAX CUT in Chapter 4, MAX 3-CSP in Chapter 5, a generalization of Γ-MAX 2-LIN$_q$ into integer domain in Chapter 7 the SDP formulation of several variants of Γ-MAX 2-LIN$_q$ in Chapter 6 and the problem of vertex pricing in Chapter 8.

In Part III, Our work is mainly to prove that several learning tasks are inherently hard to approximate; i.e., there is no better-than-trivial algorithm for the problem. In Chapter 9, we study the learnability of monomials under the agnostic learning model. In Chapter 10, we study the learnability of polynomial threshold functions (PTFs) under the same model.

1.3.2 Summary of Thesis Contributions

We summarize the main contributions of the thesis as follows:

- For Part II:

  - In Chapter 4, we give the a complete characterization of the approximability of the MAX CUT problem assuming the UGC. In particular, we can answer the following question: given a MAX CUT instance of optimum value c, what is the best polynomial time approximation guarantee we can achieve. To obtain such a result, we show that certain RPR^2 SDP rounding algorithm [50] is the optimal polynomial time algorithm for MAX CUT. In addition, we precisely determine the SDP gap, which is a important geometric property of SDP, for the MAX CUT problem.

  - In Chapter 5, we study the approximability of satisfiable MAX 3-CSP; i.e., given a 3-CSP such that there exists a perfect assignment satisfying all the constraint, what is the best approximation guarantee s we can get? The optimal approximation ratio of such a problem is also corresponds to a fundamental open problem in the area of PCP: What is the smallest s such that $\mathrm{NP} \subseteq \mathrm{naPCP}_{1,s}[O(\log n), 3]$?

    The previous best upper bound and lower bound for s are $20/27 + \epsilon$ by Khot and Saket [104] and $5/8$ by Zwick [141]. In this work we close the gap assuming Khot's d-to-1 Conjecture. Formally, we prove that if Khot's d-to-1 Conjecture holds for any finite constant integer d, then the optimal approximation for satisfiable MAX 3-CSP is indeed $5/8$.

  - In Chapter 6 we present SDP gap instances for three variants of the UNIQUE-GAMES: (i) 2-to-1 LABEL-COVER; (ii) 2-to-2 LABEL-COVER; (iii) α-constraint LABEL-COVER. Compared with the existing UNIQUE-GAMES SDP instance, the difference is that all of our SDP gap instances have perfect SDP solutions.

For alphabet size K, the optimal solutions have value: (i) $O(1/\sqrt{\log K})$; (ii) $O(1/\log K)$; (iii) $O(1/\sqrt{\log K})$. Prior to this work, there were no known SDP gap instances for any of these problems with perfect SDP value and integral optimum tending to 0.

- In Chapter 7, we study the hardness of solving integer linear systems of which each equation contains at most two variables. As is mentioned, the UGC is equivalent to the following statement: given a linear system with variables such as $x_i - x_j = c_{ij} \mod q$, it is NP-hard to find a ϵ-good solution to the system even if we know that there is an assignment that satisfies $1 - \epsilon$ fraction of the equations. It is natural to ask whether such a linear system is still hard when equations are evaluated over integers. Assuming the UGC, we prove that such a hardness still holds for equations over integers (or even real numbers).

- In Chapter 8, we consider the problem of pricing n items under an unlimited supply with single minded buyers, each of which is interested in at most k of the items . The meaning of "single minded" is that each buyer will either buy k of the items if the overall cost is within their budget or they will buy none of them. The goal is to price each item with profit margin $p_1, p_2, ..., p_n$ so as to maximize the overall profit. There is an $O(k)$-approximation algorithm when the price on each item must be above its margin cost; i.e., each $p_i > 0$. [26]

  We investigate the above problem when the seller is allowed to price some of the items below their margin cost. It was shown that by pricing some of the items below cost, the seller could possibly increase the maximum profit by $(\log n)$ times [26, 56]. These items sold at low prices to stimulate other profitable sales are usually called as "loss leader. It is unclear what kind of approximation guarantees are achievable when some of the items can be priced below cost. Understanding this question is posed as an open problem by Blum and Balcan [26]. We give a strong negative result for the problem of pricing loss leaders. We prove that assuming the Unique Games Conjecture, there is no constant approximation algorithm for item pricing with prices below cost allowed even when each customer is interested in at most 3 items.

  Conceptually, our result indicates that although it is possible to make more money by selling some items below their margin cost, it can be computationally intractable to do so.

- For Part III:
  - In Chapter 9, We prove the following strong hardness result for learning monomials: given a distribution of labeled examples of binary inputs such that there exists a monomial (conjunction) consistent with $(1 - \epsilon)$ of the examples, it is NP-hard to find a halfspace that is correct on $(1/2 + \epsilon)$ of the examples, for arbitrary constants $\epsilon > 0$. In learning theory terms, weak agnostic learning of monomials is hard, even if one is allowed to output a hypothesis from the much bigger concept class of halfspaces. As immediate corollaries of our result we

show that weak learning noisy decision lists and majorities are NP-hard. There are a large classes of learning algorithms that use halfspaces as their hypothesis such as SVM, Perceptron, Logistic Regression, *et al.* Our result rules out the possibility that any of these algorithms can be used to a learn the function class of monomials with noise.

- In Chapter 10, we prove two hardness results for the problem of agnostic learning low degree polynomial threshold functions (PTFs): for any constants $d \geq 1, \epsilon > 0$,

    - Assuming the UGC, it is NP-hard to find a degree-d PTF that is consistent with $(\frac{1}{2} + \epsilon)$ fraction of a given set of labeled examples in $\mathbb{R}^n \times \{-1, 1\}$, even if there exists a degree-d PTF that is consistent with a $1 - \epsilon$ fraction of the examples.

    - It is NP-hard to find a degree-2 PTF that is consistent with $(\frac{1}{2} + \epsilon)$ fraction of a given set of labeled examples in $\mathbb{R}^n \times \{-1, 1\}$, even if there exists a halfspace (degree-1 PTF) that is consistent with a $1 - \epsilon$ fraction of the examples.

These results immediately imply the following hardness of learning results: i) Assuming the UGC, there is no better-than-trivial proper learning algorithm that agnostically learns degree d PTFs under arbitrary distributions; ii) There is no better-than-trivial learning algorithm that outputs degree 2 PTFs and agnostically learns halfspaces (i.e., degree 1 PTFs) under arbitrary distributions.

Chapter 2

Background

In this chapter, we formally define the Constraint Satisfaction Problems and Learning Problems studied in the rest of this thesis. Also, we lay out the framework under which we analyze the approximation algorithms and in particular those based on SDP. In addition, we formally define the UGC and several variants of it based on which we derive many of the results. Last we introduce a gadget called Dictator Test and explain its relationship with hardness of approximation results for learning and CSPs.

2.1 Notations

First we define the symbols with their meaning used throughout the thesis.

Symbol :	Meaning
$\mathbb{R}$	Real numbers
$\mathbb{N}$	Natural numbers
$\mathbb{Z}$	Integer number
B_n	$\{x \in \mathbb{R}^n : \|x\| \leq 1\}$.
S_{n-1}	$\{x \in \mathbb{R}^n : \|x\| = 1\}$

Vector: For vector $x \in \mathbb{R}^n$ and $i \in [n]$, we use x_i to denote its i-th coordinate and write $x = (x_1, x_2, \ldots x_n)$. For any $S \subseteq [n]$, we use x_S to denote the collection of coordinates in set S.

2.2 Approximation and Hardness of Approximation

Given an NP-hard problem instance G and suppose the problem is a maximization problem. Let us fix the following notations: we denote optimum value of the problem to be $\mathrm{Opt}(G)$; for a polynomial-time algorithm A on the problem we use $\mathrm{Alg}_A(G)$ to denote the value output by A on G.

The traditional way to measure the quality of an approximation algorithm is to look at the *ratio:*

Definition 2.2.1. *(Approximation ratio) We call a algorithm A α-approximation if for every instance G of the problem,*

$$\frac{\mathrm{Alg}_A(G)}{\mathrm{Opt}(G)} \geq \alpha.$$

Correspondingly, we can define the hardness of approximation ratio:

Definition 2.2.2. *(Hardness of Approximation ratio) We call a problem α-hard to approximate if there is no polynomial time algorithm with better than α-approximation unless* P = NP.

The notions of approximation and hardness of approximation as ratios have some unsatisfactory aspects though. Instances with different optimum value can be of very different hardness of approximation ratio. Let us use the problem of solving linear systems over

real variables as an example. We know that when a linear system has a solution that satisfies all the equations, we can efficiently solve the problem exactly by Gaussian Elimination. However, if we only know that there is a solution that satisfies 99% of the equations, then it is known to be NP-hard to recover a solution that satisfies even 1% of the equations [68]. Another example is the MAX CUT problem. Goemans-Williamson (GW) [59] algorithm has a guarantee that this ratio is always at least .878. However this guarantee is not very good for graphs G with only moderately large maximum cuts. For example, if $\text{Opt}(G) = .55$, which means the optimum assignment satisfies .55 fraction of the constraints, then the GW algorithm may [4] only find a solution with value $.878 \cdot .55 < .49$, which is worse than the trivial one (1/2). On the other hand, Goemans and Williamson showed [59] that when $\text{Opt}(G) = .95$, their algorithm finds a solution with value at least .90, which is significantly *better* than $.878 \cdot 0.95$.

We think it is essential to measure the quality of approximation and hardness of approximation not with a single ratio but with a *curve*. Let us first assume that we have a *maximization* problem $\mathscr{P}$ with optimum value in the range of $[0,1]$.

Definition 2.2.3. *We say that an algorithm A achieves* approximation curve $\text{Apx}_A : [0,1] \to [0,1]$ *for problem $\mathscr{P}$ if*

$$\text{Alg}_A(G) \geq \text{Apx}_A(\text{Opt}(G)) \quad \text{for all instance } G.$$

Following definition is used to characterize the approximation guarantee at a particular optimum value of c.

Definition 2.2.4. *Assume $\mathscr{P}$ is an optimization problem and $\mathscr{A}$ is an algorithm for it. If any instance G with $\text{Opt}(G) \geq c$, $\text{Apx}_A(G) \geq s$, then we say that algorithm $\mathscr{A}$ (c,s)-approximate the problem $\mathscr{P}$*

Correspondingly, we can define the hardness for (c,s)-approximation; usually we prove such a claim by showing the NP-hardness of the following decision problem:

Definition 2.2.5. *For a optimization problem $\mathscr{P}$, we use $\mathscr{P}(c,s)$ to denote the problem of the following: given a instance G of $\mathscr{P}$ and distinguish the following two cases:*

1. *$\text{Opt}(G) \geq c$;*
2. *$\text{Opt}(G) < s$.*

Essentially, if $\mathscr{P}(c,s)$ is NP-hard, then it is NP-hard to (c,s)-approximate $\mathscr{P}$. This is because if there is a polynomial time algorithm that (c,s)-approximate $\mathscr{P}$, we can run the algorithm on instances of $\mathscr{G}$ and output "$\text{Opt}(G) \geq c$" if the algorithm outputs value above s.

2.3 CSPs and SDP

2.3.1 CSPs

A Constraint Satisfaction Problem (CSP) involves a system of constraints over variables $\{v_i\}_{i=1}^n$. A "k-CSP" is a system of constraints in which each constraint involves at most k of the variables. We also assume each constraint has a nonnegative weight, with the sum

weight:	constraint:
1/4	$v_1 \wedge \neg v_3 \wedge v_4$
1/4	IF v_3 THEN v_4 ELSE $\neg v_5$
1/2	$v_2 \neq v_5$

Figure 2.1: 3-CSP

of all weights being 1. Given a k-CSP, the natural algorithmic task, called "MAX k-CSP", is to find an assignment to the variables such that the total weight of satisfied constraints is as large as possible. We write "Opt" to denote the weight satisfied by the best possible assignment. We also say that a CSP is "satisfiable" if Opt = 1. Figure 2.3.1 is an example of 3-CSPs.

In a k-CSP, each constraint in a k-CSP is of a certain "type"; more precisely, it is a certain predicate with arity at most k over the variables. If we specialize Max-kCSP by restricting the type of constraints allowed, we get some of the most canonical NP optimization problems. For the special case when a CSP is over Boolean variable $v_1, ... v_n$. Let us use l_i to denote the literal which can represent either v_i or $\neg v_i$. Some of the important classes of Boolean CSPs are listed here:

- Max-2Sat: with predicate $l_i \vee l_j$;

- Max-3Sat: with predicate $l_i \vee l_j \vee l_j$;

- Max-2Lin: with predicates $v_i \oplus v_j$, $\neg(v_i \oplus v_j)$;

- MAX CUT: with predicate $v_i \neq v_j$.

- Max-3CSP: with all the possible 3-bit predicates $P(v_i, v_j, v_k): \{0,1\}^3 \to \{0,1\}$.

- Max-3CSP: with all the possible k-bit predicates $P(v_1, v_2, \ldots, v_k): \{0,1\}^k \to \{0,1\}$.

We also study some less familiar 3-CSPs in the thesis.

- MAX NTW: with predicate $\texttt{NTW}(l_1, l_2, l_3)$, where $\texttt{NTW}$ is the 3-arity predicate that evaluate truth if and only if $0, 1$ or 3 of its input is True; i.e. "Not Two True";

- MAX NAE: with predicate $\texttt{NAE}(l_1, l_2, l_3) = \neg(l_1 = l_2 = l_3)$.

Further, we also study CSPs over larger domain (other than Boolean value) such as $[q]$ or even $\mathbb{Z}$ and $\mathbb{R}$. Following are definitions of such CSPs that will be discussed in the rest of the thesis.

- MAX 2-LIN$_q$: $v_i \in [q]$, with predicates $av_i + bv_j = c \mod q$ for $a, b, c \in [q]$;

- MAX 2-LIN$_\mathbb{Z}$: $v_i \in \mathbb{Z}$, with predicates $av_i + bv_j = c$ for $a, b, c \in \mathbb{Z}$;

- MAX 2-LIN$_\mathbb{R}$: $v_i \in \mathbb{R}$, with predicates $av_i + bv_j = c$ for $a, b, c \in \mathbb{R}$;

- MAX 3-LIN$_q$: $v_i \in [q]$, with predicates $av_i + bv_j + cv_k = d \mod q$ for $a, b, c, d \in [q]$.

- MAX 3-LIN$_\mathbb{Z}$: $v_i \in \mathbb{Z}$, with predicates $av_i + bv_j + cv_k = d$ for $a, b, c, d \in \mathbb{Z}$;

- MAX 3-LIN$_\mathbb{R}$: $v_i \in \mathbb{R}$, with predicates $av_i + bv_j + cv_k = d$ for $a, b, c, d \in \mathbb{R}$. .

- Γ-MAX 2-LIN$_\mathbb{Z}$,Γ-MAX 2-LIN$_\mathbb{R}$,Γ-MAX 2-LIN$_q$: MAX 2-LIN$_\mathbb{Z}$,MAX 2-LIN$_\mathbb{R}$,MAX 2-LIN$_q$ with the additional constraints that each equation has the special form $v_i - v_j = a$, evaluated in the *corresponding* domain.

Each constraint in k-CSPs can be viewed as functions of the form: $f : \mathbb{R}^k \to \{0, 1\}$. A assignment satisfy the constraint f if f's value is 1. We can further relax the definition of CSPs by allowing constraints to be more generalized payoff functions that takes real values (other than $\{0, 1\}$). The goal of the optimization task is to find an assignment to maximize the weighted sum of the payoff on all of the constraints. In this thesis, we will also study a CSP called VERTEX-PRICING with the following generalized payoff function:

- VERTEX-PRICING$_k$: variables $v_1, v_2, \ldots v_k \in \mathbb{R}$ and the constraint is of the form

$$f_b(v_1, \ldots, v_k) = \mathbf{1}(\sum v_i < b) \cdot (\sum v_i)$$

for some positive constant $b \in \mathbb{R}^+$.

We will explain the problem in more details in Chapter 8.

2.3.2 SDP Gap

Most of the best approximation guarantees for CSPs currently known are achieved by algorithms using Semidefinite Programming (SDP). Generally speaking, SDP based algorithm involves two parts: *relaxation* of the original problem in to a SDP and *rounding* the solution of the SDP to an integer solution.

For the purpose of exposition, let us use the MAX CUT problem as an example. Suppose we have a MAX CUT instance G and it has input on Boolean variables $x_1, \ldots x_n \in \{-1, 1\}$ and a set constraints $x_i \neq x_j$ with positive weight w_{ij}.

Essentially, the optimum value of G is the maximum of the following integer programming problem:

$$\max_{x_i \in \{-1, 1\}} \sum_{ij} w_{ij} \frac{1 - x_i x_j}{2}.$$

Solving the above optimization problem is NP-hard as it is equivalent to the Max Cut problem. The SDP relaxation of Max Cut replace each x_i with vector variable $v_i \in B_n$ (i.e. $v_i \in \mathbb{R}^n, |v_i| = 1$) and replace the product of two integer by the inner product of two vectors. The following relaxed optimization problem is the SDP relaxation of MAX CUT and we call its optimum $\mathrm{Sdp}(G)$:

$$\max_{v_i \in S_{n-1}} \sum_{ij} w_{ij} \frac{1 - v_i \cdot v_j}{2}.$$

Apparently $\mathrm{Sdp}(G) \geq \mathrm{Opt}(G)$ as we can always set each v_i to be one dimensional unit vector (i.e., 1 or -1) to achieve the integral optimum. The utility of this relaxation is that we can actually find an essentially optimal in polynomial time [59]. Then after solving the relaxed optimization problems, we can figure out a set of x_i from the vector v_i. For example, after getting a set of vectors $v_1, v_2, \ldots, v_n$, the famous Goemans-Williamson algorithm uses a random hyperplane to cut all the vectors in to two parts and this naturally induces an assignment of the x_i. Another interpretation of the GW algorithm is that we first randomly picking a vector r and then we set $x_i = \mathrm{sgn}(r \cdot v_i)$. A simple analysis on

above rounding scheme shows that $\text{Alg}_{\text{GW}}(G) \geq 0.878 \cdot \text{Sdp}(G) \geq 0.878 \cdot \text{Opt}(G)$ and thus the Goemans-Williamson Algorithm achieves an 0.878-approximation.

For other CSPs, their SDP algorithms all have a similar framework: i) formulate the original problem as an integer programming; ii) relax and solve the corresponding SDP; iii) Round the SDP solution to an integer solution.

In evaluating SDP algorithms, we would like to compare the algorithm output to the optimum values. However doing this directly is difficult — roughly because Max-CSPs are usually hard, and therefore we do not analytically have access to the optimum. The approximation guarantees of SDP-based algorithms are actually based on comparing the value of the algorithm output to the *SDP value*:

Definition 2.3.1. *Given a SDP algorithm A, we use $Sdp(G)$ to denote the corresponding SDP value. We say that SDP algorithm A achieves* SDP-approximation curve $\text{SdpApx}_A :$ $[0,1] \rightarrow [0,1]$ *if*

$$\text{Alg}_A(G) \geq \text{SdpApx}_A(\text{Sdp}(G)) \quad \textit{for all } G.$$

There is an obvious barrier to how good SDP-approximation guarantees can be: If there exists a instance G with $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq s$ then of course no algorithm could have an SDP-approximation curve SdpApx with $\text{SdpApx}(c) > s$. The SDP gap is defined as follows:

Definition 2.3.2. *For $0 \leq s \leq c \leq 1$, we call the pair (c,s) an* SDP gap *if there exists a instance G with $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq s$. We define the* SDP gap curve *by*

$$\text{Gap}_{\text{SDP}}(c) = \inf\{s : (c,s) \text{ is an SDP gap}\}.$$

In addition, the SDP gap gives a measure of how close the SDP is to the original integer programming problem.

2.4 Learning Theory

2.4.1 Concept Classes

Computational Learning Theory establishes the theoretical framework of how can we infer an unknown target function from examples under certain distributions. We usually assume that the target function is from some simple concept class. Let us define concept class as follows (assuming we only consider binary examples and labels)

Definition 2.4.1. *(Concept Class) A concept class is a class of functions on $f : \{0,1\}^n \rightarrow \{-1,1\}$.*

Here is a list of concept classes studied in the thesis.

Definition 2.4.2. *(monomials) Suppose the input to the function is $x \in \{0,1\}^n$, suppose l_i is the literal that can represent either x_i or $\neg x_i$. A monomial is the conjunction on a subset of literals which can be represented as:*

$$\bigwedge_{i \in S} l_i.$$

for some $S \subseteq [n]$.

Definition 2.4.3. *(decision lists) A decision list f over the Boolean variables $x \in \{0,1\}^n$ is represented by a list of variable pairs $(l_1,b_1),(l_2,b_2),...,(l_k,b_k)$ and b_{k+1} where each l_i is a literal (being either x_i or $\neg x_i$) and each b_i is either -1 or 1. Given any $x \in \{0,1\}^n$, the value of $f(x)$ is b_i if i is the smallest index such that l_i is made true by x; if no l_i is true then $l(x) = b_{k+1}$.*

Definition 2.4.4. *(halfspaces) Suppose the input is $x \in \{0,1\}^n$ (or $\mathbb{R}^n$). A halfspace function $f(x) : \{0,1\}^n \to \{-1,1\}$ is the sgn of the weighted sum of all the x_i subtracted by a threshold:*

$$\operatorname{sgn}(\sum w_i x_i - \theta)[1].$$

Here $w_1,\dots,w_n,\theta \in \mathbb{R}$.

Definition 2.4.5. *(degree d PTFs) For positive integer d, we call a function $f(x) : \{0,1\}^n \to \mathbb{R}$ (or $\mathbb{R}^n \to \mathbb{R}$) a degree d polynomial function if it is of the following polynomial expansion form:*

$$\sum_{\text{multiset } S \subseteq [n], |S| \leq d} c_S \prod_{i \in S} x_i.$$

Here each $c_S \in \mathbb{R}$ is the coefficient of the polynomial. A degree d polynomial threshold function is of the form $\operatorname{sgn}(f(x))$ where $f(x)$ is a degree d polynomial function.

A relationship among all these concept classes is that:
$$\text{monomials} \subseteq \text{decision lists} \subseteq \text{halfspaces} \subseteq \text{degree } d \text{ PTFs}.$$

2.4.2 Learning Models

In learning theory, researchers study whether these common concept classes are *learnable*. The *learnability* of a concept class is defined under the PAC learning model by Valiant [140].

Definition 2.4.6. *(PAC Learning) We say an algorithm $\mathscr{A}$ efficiently learns a Boolean function class $\mathscr{F}$ if the following is true for any $\delta, \epsilon > 0$ and distribution D on $\{0,1\}^n$ and f in $\mathscr{F}$: Suppose $\mathscr{A}$ has an oracle access to example-label pairs $(x,f(x))$ for x sampled from distribution D, it will output some hypothesis h in certain concept class $\mathscr{H}$ such that with probability $1 - \delta$, $\mathbf{Pr}(h(x) = f(x)) \geq 1 - \epsilon$ with running time $\operatorname{poly}(1/\epsilon, 1/\delta, n)$. We call the learning algorithm proper if $\mathscr{F} = \mathscr{H}$.*

While the original PAC learning model assumes that some function $f \in \mathscr{F}$ perfectly labels all the data, this model has been generalized by Haussler [79] and Kearns [90] to address the noise. In addition, the new models has extended to functions over real value input: $f : \mathbb{R}^n \to \{-1,1\}$. Under their model (which is called the *agnostic learning model*), it is only known that there is some function in a particular concept class $\mathscr{F}$ that has correctly labeled a c fraction of the examples, the goal of the learning to come up with a hypothesis with accuracy being close to c.

Definition 2.4.7. *(Agnostic Learning) We say an algorithm $\mathscr{A}$ agnostically learns a concept class $\mathscr{F}$ if the following is true for any $\delta, \epsilon > 0$ and distribution D on $\{0,1\}^n$ (or even $\mathbb{R}^n$): Suppose $\mathscr{A}$ has an oracle access to example-label pairs (x,l_x) for x sampled from*

---

[1]in this thesis, we use the convention that $\operatorname{sgn}(x)$ is 1 for $x \geq 0$ and -1 for $x < 0$.

distribution D and suppose the best hypothesis in $\mathscr{F}$ has an accuracy at least c; i.e., $\max_{f \in \mathscr{F}} \mathbf{Pr}(f(x) = l_x) \geq c$, then $\mathscr{A}$ will output some hypothesis $h \in \mathscr{H}$ such that with probability at least $1 - \delta$, $\mathbf{Pr}(h(x) = l_x) \geq c - \epsilon)$ with running time $\mathrm{poly}(1/\delta, 1/\epsilon, n)$.

The agnostic model still defines learnability as whether an algorithm can find a hypothesis that has almost the optimal accuracy; in practice, come up with a hypothesis with any non-trivial (and not necessarily optimal) performance would still be useful. It is quite natural to relax the agnostic learning model to address this.

Definition 2.4.8. *((c,s) Agnostic Learning) For $0 \leq s \leq c \leq 1$, we say an algorithm $\mathscr{A}$ agnostically (c,s) learns concept class $\mathscr{F}$ if the following is true for any $\delta > 0$ and distribution D on $\{0,1\}^n$ (or even $\mathbb{R}^n$): Suppose $\mathscr{A}$ has an oracle access to example-label pairs (x, l_x) for x sampled from distribution D and suppose the best hypothesis in $\mathscr{F}$ has accuracy at least c; i.e., $\max_{f \in \mathscr{F}} \mathbf{Pr}(f(x) = l_x) \geq c$, then $\mathscr{A}$ will output some hypothesis $h \in \mathscr{H}$ such that with probability $1 - \delta$, $\mathbf{Pr}(h(x) = l_x) \geq s - \epsilon)$ with running time $\mathrm{poly}(1/\delta, 1/\epsilon, n)$.*

Uniform convergence results in Haussler's work [78] (and see also [90]) implies that for most common simple concept class[2], learnability of $\mathscr{F}$ by outputting hypothesis in $\mathscr{H}$ in the above agnostic model is equivalent to the approximability of the problem of finding hypothesis in $\mathscr{H}$ that has the same agreement rate as the best hypothesis in $\mathscr{C}$ on the given set of examples.

We use $\mathscr{F}$-$\mathscr{H}$-MA to denote the optimization problem of finding an optimal function in $\mathscr{H}$ that approximate the best function in $\mathscr{F}$ on a set of examples. If $\mathscr{F} = \mathscr{H}$, we just write it as $\mathscr{F}$-MA. We also define the following decision problem to characterize its approximability.

Definition 2.4.9. *For $0 \leq s \leq c \leq 1$, and a given set of examples, we want to distinguish the following two cases:*

1. *There is some hypothesis $f \in \mathscr{F}$ such that agrees with a c fraction of the examples.*

2. *No hypothesis in $\mathscr{H}$ agrees more than an s fraction of the examples.*

We call above decision problem $\mathscr{F}$-$\mathscr{H}$-MA (c,s)

Therefore, by the uniform convergence results, the NP-hardness of the above problem suggests the NP-hardness of (c,s) agnostically learn a hypothesis in $\mathscr{F}$ by concept class $\mathscr{H}$. And when $\mathscr{F} = \mathscr{H}$, the hardness of $\mathscr{F}$-MA implies the hardness of proper learning concept class $\mathscr{F}$.

In the thesis, we will investigate the above problems for some natural selection of $\mathscr{H}$ and $\mathscr{F}$. We are mostly interested in the cases when $c = 1 - o(1)$; i.e., we want to understand the learnability of a concept class $\mathscr{F}$ knowing that there is indeed some hypothesis in it that almost correctly labels all the examples.

For notation convenience, we use the following short name for the concept class we have defined:

- MON: monomials;
- HS: halfspaces;
- DL: decision lists;

---

[2]The requirement for the uniform convergence results to hold is that a concept class should have polynomial VC dimension, a requirement that all the concept classes we study in the thesis satisfy.

- PTF_d: degree d PTFs.

We study the approximability of MON-HS-MA in Chapter 9 and MA-PTF_d in Chapter 10.

2.4.3 Related Work

A number of hardness results for proper agnostic learning of monomials, decision lists and halfspaces have appeared in the literature. For monomials, MON-MA was shown to be NP-hard by Kearns and Li [91]. The hardness of approximating the problem within some constant factor (i.e., APX-hardness) was first shown by Ben-David *et al.* [20]. The factor was improved to 58/59 by Bshouty and Burroughs [16]. Finally, Feldman showed a tight inapproximability result [51] (see also [52]), namely that MON-MA $(1 - \epsilon, 1/2 + \epsilon)$ is NP-hard. Recently, Khot and Saket [105] proved a similar hardness result even when a t-CNF is allowed as output hypothesis for an arbitrary constant t (a t-CNF is the conjunction of several clauses, each of which has at most t literals; a monomial is thus a 1-CNF). The Maximum Agreement problem for halfspaces (HS-MA) was shown to be NP-hard to approximate by Amaldi and Kann [5], Ben-David *et al.* [20], and Bshouty and Burroughs [16] for approximation factors $\frac{261}{262}$, $\frac{415}{418}$, and $\frac{84}{85}$, respectively. An optimal inapproximability result was established independently by Guruswami and Raghavendra [68] and Feldman *et al.* [52] showing NP-hardness of HS-MA $(1 - \epsilon, 1/2 + \epsilon)$ for any $\epsilon > 0$. The reduction in [52] produced examples with real-valued coordinates, whereas the proof in [68] worked also for examples drawn from the Boolean hypercube. For the concept class of decisions lists, APX-hardness of its Maximum Agreement problem (DL-MA) was shown by Bshouty and Burroughs [16]. As for the concept class of low degree PTFs, its hardness of knowing result is not well understood before our work.

A number of hardness of approximation results are also known for the symmetric problem of minimizing disagreement for each of the above concept classes [7, 16, 51, 52, 80, 90]. Another well-known evidence of the hardness of agnostic learning of monomials is that even a non-proper agnostic learning of monomials would give an algorithm for learning DNF — a major open problem in learning theory [109]. Further, Kalai *et al.* proved that even agnostic learning of halfspaces with respect to the uniform distribution implies learning of parities with random classification noise — another long-standing open problem in learning theory and coding [84].

On the algorithmic side, monomials, decision lists, halfspaces and low degree PTFs are all known to be PAC-learnable. Monomials, decision lists and halfspaces are even known to be efficiently learnable in the presence of more benign *random* classification noise [6, 22, 33, 89, 92]. Simple online algorithms like Perceptron and Winnow learn halfspaces when the examples can be separated with a significant *margin* (as is the case if the examples are consistent with a monomial) and are known to be robust to a very mild amount of adversarial noise [12, 57, 58]. Kalai *et al.* gave the first non-trivial algorithm for agnostic learning monomials in time $2^{\tilde{O}(\sqrt{n})}$ [84]. They also gave a breakthrough result for agnostic learning of halfspaces with respect to the uniform distribution on the hypercube up to any constant accuracy (and analogous results for a number of other settings). Their algorithms output linear thresholds of parities as hypotheses. Very recent work [42]

has in fact given efficient agnostic learning algorithms for low-degree PTFs under specific distributions on examples such as Gaussian distributions or the uniform distribution over Boolean Cube.

2.5 LABEL-COVER and Khot's Conjectures

In this section, we formally state the UGC which leads to a numerous hardness of approximation results. To begin with, let us first define the LABEL-COVER Problem, of which the UNIQUE-GAMES is a special case.

Definition 2.5.1. *A* LABEL-COVER *instance $\mathscr{L}$ is defined by a tuple $(U, V, E, P, R_1, R_2, \Pi)$. Here U and V are the two vertex sets of a bipartite graph and E is the set of edges between U and V. P is an explicitly given probability distribution on E. R_1 and R_2 are integers with $1 \le R_1 \le R_2$. Π is a collection of "projections", one for each edge: $\Pi = \{\pi_e : [R_2] \to [R_1] \mid e \in E\}$. A labeling L is a mapping $L : U \to [R_1], V \to [R_2]$. We say that an edge $e = (u, v)$ is "satisfied" by labeling L if $\pi_e(L(v)) = L(u)$. We define:*

$$\text{Opt}(L) = \max_{\text{all labelling } L} \mathbf{Pr}_{e=(u,v)\sim P}[\pi_e(L(v)) = L(u)].$$

The fundamental inapproximability theorem of Raz [128] is the following statement of the hardness of approximating the LABEL-COVER problem:

Theorem 2.5.2. *There exists some positive constant η such that for every constant $\epsilon > 0$ for any $1/k^\eta \le \epsilon$ and* LABEL-COVER *instances with alphabet size k (or above),* LABEL-COVER *$(1, \epsilon)$ is NP-hard.*

In [97], Khot conjectured that several restricted forms of the LABEL-COVER problem are also NP-hard.

Definition 2.5.3. *(d-to-1 LABEL-COVER) A projection $\pi : [R_2] \to [R_1]$ is said to be "d-to-1" if for each element $i \in [R_1]$ we have $1 \le |\pi^{-1}(i)| \le d$. The d-to-1* LABEL-COVER *is the special case of* LABEL-COVER *in which each projection in Π is d-to-1.*

A Unique Games instance is the special case when $d = 1$ and sometimes it is also referred as the Unique LABEL-COVER. The UGC is that it is NP-hard to distinguish *near satisfiable* instance from instances with tiny optimum value.

Conjecture 2.5.4. *(UGC) For every constant $\epsilon > 0$ there is some constant $k(\epsilon)$ such that for* UNIQUE-GAMES *with label size greater than $k(\epsilon)$,* UNIQUE-GAMES *$(1 - \epsilon, \epsilon)$ is NP-hard.*

It is easy to see that Γ-MAX 2-LIN$_q$ is a special case of Unique Gamses with alphabet size q. By the work of [99], it is also known that Unqiue Games is equivalent to the following statement:

Conjecture 2.5.5. *(Equivalent statement of UGC) For any constant ϵ, there exists large enough q, such that Γ-MAX 2-LIN$_q$ $(1 - \epsilon, \epsilon)$ is NP-hard.*

If we want to parameterized the soundness by the size of alphabet, following statement is equivalent to UGC [99].

Conjecture 2.5.6. *For any constant ϵ, there exists large enough q, such that Γ-MAX 2-LIN$_q$ $(1 - \epsilon, 1/q^{\frac{\epsilon}{2-\epsilon}})$ is NP-hard.*

It should be note that when a Γ-MAX 2-LIN$_q$ instance has optimum value is 1, such a problem can be easily solved by Gaussian Elimination. This also is true for UNIQUE-GAMES. In comparison, LABEL-COVER is NP-hard to ϵ-approximate even when it has optimum value 1. Khot's d-to-1 Conjecture address the above difference by assuming that when $d \geq 2$, d-to-1 Label Cover is also hard for satisfiable instance.

Conjecture 2.5.7. *(d-to-1 Conjecture) For every constant $\epsilon > 0$ there is some constant $k(\epsilon)$ such that for d-to-1 LABEL-COVER instances $\mathscr{L}$ with $R_2 \geq k(\epsilon)$, d-to-1 LABEL-COVER $(1,\epsilon)$ is NP hard.*

2.5.1 UGC v.s. d-to-1 Conjecture

Since UNIQUE-GAMES does not have perfect completeness; i.e., it is easy when $opt = 1$, None of the UGC-based hardness results applies to the *satisfiable* Max-Φ problems, i.e., the $(1,s)$-approximability, by current reduction machineries. In comparison, the d-to-1 Conjecture states that it is NP-hard to distinguish whether a d-to-1 LABEL-COVER instances is *satisfiable* or *far from satisfiable*; it can be easily adapted to the reduction that address the approximability of satisfiable instance. The first application of the d-to-1 Conjecture is by Dinur et al. [43] where they use some variant of the 2-to-1 Conjecture to obtain hardness of approximation result for the 4-Coloring problems. The reason they can not use UGC is because assuming UGC, they can only obtain hardness results applies to "almost 4-colorable" graph. There has also been several other works that use the d-to-1 conjecture to derive the hardness for satisfiable instance [71, 120, 137],

Assuming the correctness of d-to-1 conjecture, we present a $(1, 5/8 + o(1))$ hardness for 3-CSP that appear in Chapter 5 in this thesis. In addition, one may also wonder is it possible to use SDP to solve satisfiable d-to-1 LABEL-COVER so as to disprove d-to-1 conjecture? We make some partial progress on understanding the SDP gap of d-to-1 LABEL-COVER in Chapter 6.

2.6 Dictator Testing

In this section, we introduce a gadget called "Dictator Testing" which is strongly motivated by its applications to proving hardness-of-approximation results for CSPs and Learning. Generally speaking, we have black-box query access to an unknown Boolean function $f : \{0,1\}^n \to \{0,1\}$ and the goal is to test the extent to which f is close to a "dictator" function; i.e., one of the n functions of the form

$$f(x_1, \ldots, x_n) = x_i.$$

Dictator Testing is in somewhat different form for learning and CSP applications and we discuss the difference in the following two sections.

2.6.1 Dictator Testing for CSPs

A "test" is a randomized algorithm which makes a very small number of queries to f and then either "accepts" or "rejects". The Dictator Testing problem was first studied

by Bellare, Goldreich, and Sudan [19], with hardness-of-approximation for CSPs as the motivation. It was later independently introduced, with the "dictator" terminology, by Parnas, Ron, and Samorodnitsky [122].

Definition 2.6.1. *A Dictator Test has* completeness *at least c if all n dictator functions are accepted with probability at least c. We say a Dictator Test has* perfect completeness *if it has completeness* 1.

The Dicator Test should also have the property of rejecting functions far from Dictator with high probability, which is also called the soundness of the test. Håstad [75, 76] introduced a notion of "quasirandom" function as one way of defining functions far from being Dictator. One can think of it as functions f which have correlation at most $o(1)$ with every "*junta*" (function depending on only $O(1)$ coordinates). Another way of thinking of these functions is that we *cannot* have a procedure of outputting an $O(1)$ list of coordinates of the functions satisfying the following property: when we permute the function, the corresponding coordinates in the list is also permuted. We refer to such tests as "Dictator-vs.-quasirandom Tests". As Håstad and others have demonstrated, Dictator-vs.-quasirandom Tests can often be used to prove optimal inapproximability results for CSPs.

Definition 2.6.2. (Informal.) *A Dictator-vs.-quasirandom Test has* soundness *at most s if every quasirandom function is accepted with probability at most s + o(1).*

Let us use the 3-CSP as an example. Suppose $\mathcal{T}$ is a 3-query Dictator-vs.-quasirandom Test on functions $f : \{0,1\}^n \to \{0,1\}$. Imagine we consider all possible random choices of $\mathcal{T}$, and in each case write down the (up to) 3 strings x, y, z queried and the predicate applied to the outcomes to decide accept/reject. The complete behavior of $\mathcal{T}$ might then look like the following:

$$
\begin{array}{llllllll}
\text{with probability } p_1, & \text{accept iff} & f(x^{(1)}) & \vee & f(y^{(1)}) & \vee & f(z^{(1)}) \\
\text{with probability } p_2, & \text{accept iff} & \neg f(x^{(2)}) & \vee & f(y^{(2)}) \\
\text{with probability } p_3, & \text{accept iff} & \neg f(x^{(3)}) & \vee & \neg f(y^{(3)}) & \vee & \neg f(z^{(3)})
\end{array}
$$

$$\cdots$$

This is precisely an instance of Max-3CSP, in which the "variables" are the $f(x)$'s. Note that the weights p_i indeed sum up to 1. More generally, if $\mathcal{T}$ makes at most k queries it can be viewed as an instance of Max-kCSP. Further, suppose that $\mathcal{T}$ "uses the predicate set Φ" — i.e., its decision to accept/reject is always based on applying a predicate from the set Φ to its query responses. Then $\mathcal{T}$ can be viewed as an instance of Max-Φ. The above example illustrates a tester which uses the set of ORs of up to 3 literals; thus it can be viewed as an instance of Max-3Sat.

Suppose that $\mathcal{T}$ is a Dictator Test with completeness at least c. Then the Opt of the associated CSP is at least c; indeed, there are n distinct solutions, the dictators, of value at least c. More crucially, suppose further that $\mathcal{T}$ is a Dictator-vs.-quasirandom Test with soundness at most s. This means that any solution f satisfying slightly more than weight s of the constraints must be slightly correlated with a junta on constant number of coordinates; i.e., it must "highlight" a small number of dictators. These two properties of the test, taken together, make it useful as a *gadget* in an NP-hardness-of-approximation

reduction. Specifically, if $\mathcal{T}$ uses predicate set Φ, it can be used to prove hardness for the Max-Φ problem. Indeed, in the study of inapproximability, one has the following "Rule of Thumb":

Rule of Thumb for CSPs   *For the Max-Φ problem, to prove that* MAX Φ (c,s) *is hard, construct a Dictator-vs.-quasirandom Test using Φ, with completeness c and soundness s. We call the pair (c,s) a* dictator-vs.-quasirandom *gap.*

It is natural to ask among all the Dictator Test with completeness c, how small could the soundness s be? .

Definition 2.6.3. *We define the* dictator-vs.-quasirandom gap curve *by*

$$\mathrm{Gap}_{\mathrm{Test}}(c) = \inf\{s : (c,s) \text{ is a dictator-vs.-quasirandom gap}\}.$$

2.6.2   Dictator Test for Learning

The dictator test is also very useful in the learning problems; it is of a somewhat different form: we can only make one query on a Boolean function $f(x_1, .. x_n)$; however, we can assume that f is in some simple function classes that we want to prove hardness results for. Another difference is that the dictator test in learning usually checks whether two functions are "matching dictator" as we will explain further.

For the sake of exposition of the usage of a dictator test, let us sketch a proof for the hardness of HS-MA $(1 - \epsilon, 1/2 + \epsilon)$.

Proposition 2.6.4. *Assuming the UGC, the problem* HS-MA $(1 - \epsilon, 1/2 + \epsilon)$ *is* NP-*hard.*

As is mentioned, the same hardness result (based on P $\neq$ NP) has been established in [53, 68]. However, the following construction is different from (and somewhat simpler than) the other proofs; it helps to illustrate the relationship between hardness of learning and Dictator Test.

Given an instance $\mathcal{L}$ of UNIQUE-GAMES, we will produce a set of labelled examples such that the following holds: if $\mathcal{L}$ is almost satisfiable instance, then there is a halfspaces that agrees with $1 - \epsilon$ fraction of the examples, while if $\mathcal{L}$ is a near unsatisfiable instance then no halfspace has agreement more than $\frac{1}{2} + \epsilon$. Clearly, a reduction of this nature immediately implies Proposition 2.6.4.

Let $\mathcal{L}$ be an instance of UNIQUE-GAMES with an associated graph $G = (U, V, E)$ and a set of labels $[k]$. The examples we generate will have $(|V| + |U|)k$ coordinates, i.e., belong to $\mathbb{R}^{(|U|+|V|)k}$. These coordinates are to be thought of as one block of k coordinates for every vertex $w \in U \cup V$. We will index the coordinates of $x \in \mathbb{R}^{(|U|+|V|)k}$ as $x = (x_w^i)_{w \in U \cup V, i \in [k]}$.

Also for any halfspace function $f : \mathbb{R}^{(|U|+|V|)k}$, we use the notion of f_w for the restriction of f on some vertex $w \in U \cup V$ by setting or the coordinate $x_{w'}^i = 0$ when $w' \neq w$. Similarly, for a particular edge e, we denote f_e for edge $e \in E$ as f's restriction by setting all $x_{w'}^i$ to be 0 for $w' \notin e$.

For every labelling $l : U \cup V \rightarrow [k]$ of the instance, there is a corresponding halfspace

over $\mathbb{R}^{(|V|+|U|)k}$ given by,

$$\text{sgn}(\sum_{u \in U} x_u^{(l(u))} - \sum_{v \in V} x_v^{(l(v))}).$$

The idea is to construct a distribution of examples properly such that if there is a good labelling for the UNIQUE-GAMES, then the above corresponding halfspace has a good agreement rate. On the contrary, if any halfspace with $\frac{1}{2} + \epsilon$ agreement somehow implies a labelling of l satisfying a constant fraction of the edges in $\mathscr{L}$.

Fix an edge $e = (u,v)$. For the sake of exposition, let us assume π^e is the identity permutation for every $i \in [k]$. For each edge e, we require a set of examples $\mathscr{D}_e$ with the following properties:

- All coordinates x_w^i for a vertex $v \notin e$ are fixed to be zero.
- For any label $i \in [k]$, $\text{sgn}(x_u^i - x_v^i)$ has agreement $1 - \epsilon$ with the examples $\mathscr{D}_e$.
- If f has agreement $\frac{1}{2} + \epsilon$ on the set of examples $\mathscr{D}_e$, then there exists a labelling strategy L_f for each $w \in U \cup V$ solely based on f_w such that, L_f satisfies the edge e with non-negligible probability.

As the distribution of $\mathscr{D}_e$ looks at the restriction of f on edge e which can be viewed as a halfspace on $\mathbb{R}^{2k} \to \mathbb{R}$, we can rephrase above requirement as a pure property testing problem. Given a degree halfspace function $f_e : \mathbb{R}^{2k} \to \mathbb{R}$, we need a randomized procedure of generating examples that has the following property:

The procedure must satisfy:

- (Completeness) If $f_e(x) = x_u^i - x_v^i$ then the test accepts with probability $1 - \epsilon$.
- (Soundness) If the test accepts with probability $\frac{1}{2} + \epsilon$, then we can output a coordinate of f_u and a coordinate of f_v such that they match each other with non-negligible probability.

As we can see here, above test not only check whether function f_u, f_v is a dictator. In addition it only accepts when they are dictator with matching coordinate.

We claim that following test will serve the goal:

Matching Dictator Test $\mathscr{T}_1$

Choose ϵ to be $\frac{1}{\log k}$ and δ to be "extremely small"

1. Generate independent ϵ-biased bits $a_1, a_2, \ldots, a_n \in \{0, 1\}$ (i.e., $a_i = 1$ with probability ϵ and 0 with probability $1 - \epsilon$).
2. Generate $2n$ independent unit Gaussian random variables:

$$h_1, h_2 \ldots, h_k, g_1, g_2 \ldots, g_k.$$

3. Generate a random bit $b \in \{-1, 1\}$.
4. Set $r = (a_1 h_1 + g_1, \ldots, a_k h_k + g_k, g_1, \ldots, g_k)$ and $u = (1, 1, 1, \ldots, 1, 0, \ldots, 0) \in \mathbb{R}^{2k}$.
5. Set $y = r + b\delta u$.
6. Accept if $\text{sgn}(f_e(y)) = b$.

Suppose that $f_e(x) = \theta + \sum_{i=1}^{k} w_u^i x_u^i + \sum_{i=1}^{k} w_v^i x_v^i$. Also without loss of generality assume that $\sum_{i=1}^{k} w_v^i = 1$. Then we know $f_e(y) = f_e(r) + b\delta$. Essentially, the test checks below two

inequalities with equal probability

- $f_e(r) \leq \delta$

- $f_e(r) \geq -\delta$

Since at least one of the above two inequality will hold, the passing probability of f_e is $\frac{1}{2} + \frac{1}{2}\mathbf{Pr}(f_e(r) \in (-\delta, \delta))$.

As δ is extremely small, roughly we can think of the passing probability of f_e to be $\frac{1}{2} + \frac{1}{2}\mathbf{Pr}(f(r) = 0)$. On the completeness side, it is easy to check that for $f_e = x_u^i - x_v^i$, $f(r) = a_i h_i$ and $\mathbf{Pr}(f(r) = 0) = 1 - \epsilon$. Overall, it passes the test with probability $1 - \epsilon$.

On the soundness side as $f_e(r) = \sum_i (w_u^i + w_v^i)g_i + \sum w_u^i a_i h_i$, to make $\mathbf{Pr}(f(r) = 0)$ to be non-negligible, we must have $w_u^i + w_v^i = 0$ for each i. Also there must be very "few" nonzero w_u^i as otherwise $\sum w_u^i a_i h_i$ will not vanish. Then a good labelling strategy would be randomly output a coordinate with nonzero weights in f_u and f_v. As there are very few such coordinates, we know with non-negligible probability, they will match.

Generally speaking following is the rule of thumb for proving hardness of learning results

Rule of Thumb for Learning   To prove $\mathscr{F}$-$\mathscr{H}$-MA (c, s) is hard, construct a one query matching Dictator Test such that dictator functions in $\mathscr{F}$ pass with probability at least c while non-dictator functions in $\mathscr{H}$ pass with probability s.

Chapter 3

Mathematical Tools

In this Chapter, we summarize the mathematical tools that is used throughout the thesis.

3.1 Probability Theory

3.1.1 Product Space

The usual way of defining a probability space is a triple: a sample space Ω, a σ-algebra, and a probability measure $\mathscr{P}$. In this thesis as we mostly study the probability space of which Ω is a finite set (with the exception of the Gaussian distribution), we denote a probability space by a pair (Ω, μ) where Ω is the sample space and $\mu : \Omega \to (0,1]$ is the density function.

Definition 3.1.1. *(finite probability space) Let Ω be a finite set of events $\{e_1, \dots e_q\}$. We denote (Ω, μ) to be a* probability space *where $\mu : \Omega \to (0,1]$ is the probability measure on Ω such that $\sum_{i=1}^{q} \mu(e_q) = 1$. The* minimum atom probability *of Ω is defined to be $\min_{i \in [q]} \mu(e_i)$.*

Definition 3.1.2. *(inner product) Given a finite probability space (Ω, μ) and $|\Omega| = q$, we know that the function space $\mathscr{F} = \{f \mid f : \Omega \to \mathbb{R}\}$ is a q-dimensional vector space; we define the inner product induced by the probability measure μ as follows: For any $f, g : \Omega \to \mathbb{R}$,*

$$< f, g > = \mathbf{E}_{e \sim (\Omega, \mu)}[f(e) \cdot g(e)].$$

Definition 3.1.3. *For all $f : \Omega \to \mathbb{R}$, we define its p-norm as*

$$\|f\|_p = (\mathbf{E}_{e \sim (\Omega, \mu)}[|f(e)|^p])^{1/p}.$$

Definition 3.1.4. *(Ensemble) Given a finite probability space (Ω, μ). Suppose that $|\Omega| = q$. We call the collection of functions $(\chi_0, \dots \chi_{q-1})$ an* ensemble *if $\chi_0, \dots \chi_{q-1}$ is an basis for $\mathscr{F}$. Further, we call an ensemble an* orthogonal ensemble *if the ensemble forms an orthogonal basis and χ_0 is the constant 1 function; i.e., $\chi_0(e) = 1$ for any $e \in \Omega$.*

To characterize a finite probability spaces, we can either use (Ω, μ) or an orthogonal ensemble $(\chi_0 = 1, \dots, \chi_{q-1})$ on it.

Next, we introduce the definition of the *product* of probability spaces:

Definition 3.1.5. *(Product Space) For probability spaces $(\Omega_1, \mu_1), (\Omega_2, \mu_2), \dots (\Omega_n, \mu_n)$, we define their* product probability space *$(\Omega, \mu) = \prod_{i=1}^{n}(\Omega_i, \mu_i)$ as follows : the sample space is $\Omega = \prod_{i=1}^{n} \Omega_i = \{(e_1, \dots e_n) \mid i \in [n], e_i \in \Omega_i\}$ and the probability measure μ on any event $(e_1, \dots e_n) \in \prod_{i=1}^{n} \Omega_i$ is defined to be $\prod_{i=1}^{n} \mu_i(e_i)$.*

For simplicity we assume that each Ω_i has the same cardinality q. Also for each function space $\mathscr{F}_i = \{f \mid f : \Omega_i \to \mathbb{R}\}$, we denote its orthogonal ensemble on it as $\{\chi_{i,0} = 1, \chi_{i,1}, \dots, \chi_{i,q-1}\}$. By the fact from basic linear algebra, the function spaces on

$$\mathscr{F} = \{f \mid \prod_{i=1}^{n} \Omega_i^n \to \mathbb{R}\}$$

has an orthogonal basis $\{\chi_\sigma : \prod_{i=1}^{n} \Omega_i \to \mathbb{R} \mid \sigma \in [q]^n\}$ with each χ_σ defined as follows: for $x \in \prod_{i=1}^{n} \Omega_i$,

$$\chi_\sigma(x) = \prod_{i=1}^{n} \chi_{i,\sigma_i}(x_i).$$

46

Therefore any function $f \in \mathcal{F}$ can be written as a linear combination of the basis:

$$f(x) = \sum_{\sigma \in [q]^n} \hat{f}(\sigma) \chi_\sigma(x)$$

We call $\hat{f}(\sigma)$ to be f's *Fourier coefficients* on σ, where $\sigma \in [q]^n$ are also referred as the *multidimensional index.*

As each $\chi_{i,0}$ is the constant 1 function, we can ignore them in the expression of χ_σ; i.e., write $\chi(\sigma)$ as $\prod_{\sigma_i \neq 0} \chi_{i,\sigma_i}(x_i)$. We therefore define the *active* elements for any $\sigma \in [q]^n$ to be

$$S(\sigma) = \{i \mid \sigma_i \neq 0\}.$$

For any $\sigma \in [q]^n$, we define $\deg(\chi_\sigma)$, the degree of the term χ_σ, to be the number of active elements $|S(\sigma)|$.

As any function in $\mathcal{F}$ can be viewed as a multilinear[1] polynomial on functions: $\{\chi_{i,j} \mid i \in [n], 1 \le j \le q\}$. The degree of a function is then defined by the maximum degree among all of its term with nonzero Fourier coefficients.

Definition 3.1.6. *(Degree) For function $f = \sum_{\sigma \in [q]^n} \hat{f}(\sigma) \chi_\sigma(x)$, we define its degree as follows:*

$$\deg(f) = \max\{|\deg(\chi_\sigma) \mid \hat{f}(\sigma) \neq 0, \sigma \in [q]^n.\}$$

Following is a relationship between the Fourier representation and variance of a function:

Fact 3.1.7.

$$\mathbf{Var}(f) = \sum_{|S(\sigma)| \ge 1} \hat{f}(\sigma)^2.$$

.

For any $S \subseteq [n]$, if we take f_S to be $\sum_{S(\sigma)=S} \hat{f}(\sigma)$.and write f as the sum of f_S, we get the Efron-Stein Decomposition.

Theorem 3.1.8. *(Efron-Stein Decomposition [45]) Let $(\Omega_1, \mu_1), ..(\Omega_n, \mu_n)$ be discrete probability spaces. Then for $f : \prod_{i=1}^n (\Omega_i, \mu_i) \to \mathbb{R}$: for $S \subseteq [n]$, if we write f as*

$$f(x) = \sum f_S(x),$$

we call above representation the Efron-Stein Decomposition of f. Such a decomposition has the following properties:

- *$f_S(x)$ depends only on variables in x_S.*
- *For all $S \not\subseteq S'$ and $a_{S'} \in \prod_{i \in S'} \Omega_i$, it holds that $E[f_S(x)|x_{S'} = a_{S'}] = 0$.*

3.1.2 Influence, Noise and Stability

Given a function $f : \prod_{i=1}^n \Omega^i \to \mathbb{R}$ on a probability space $(\Omega, \mu) = \prod(\Omega_i, \mu_i^n)$, we can define its influence on the i-th input as follows:

---

[1]By multilinear, we mean that the power of χ_i in each χ_σ is at most 1.

Definition 3.1.9. *(Influence) The influence of the i-th coordinate is defined to be*

$$\text{Inf}_i = \mathbf{E}_{x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_n}[\mathbf{Var}_{x_i} f(x)].$$

The influence of f on i-th coordinate is the average variance of f over the possible configurations on the other coordinates.

The influence of a function can be represented in terms of the Fourier Coefficients:

Fact 3.1.10.

$$\text{Inf}_i(f) = \sum_{i \in S(\sigma)} \hat{f}(\sigma)^2.$$

We can also generalize above notion to define the low degree influence of a function as follows:

Definition 3.1.11. *For any integer d, $\text{Inf}_i^{\leq d}(f) = \sum_{i \in S(\sigma), |S(\sigma)| \leq d} \hat{f}(\sigma)^2$.*

The sum of all the low degree influence is bounded by d times of the variance.

Fact 3.1.12.

$$\sum_{i=1}^{n} \text{Inf}_i^{\leq d}(f) \leq d \cdot \mathbf{Var}(f).$$

Next we define a important concept called Noise Operator.

Definition 3.1.13. *For a probability space $\prod_{i=1}^{n}(\Omega_i, \mu_i)$ and $0 \leq \rho \leq 1$, we define the* noise operator *T_ρ on functions on $f : \Omega \to \mathbb{R}$ as follows:*

$$T_\rho f(x) = \mathbf{E}[f(x')],$$

where x' has the following distribution: independently each x_i' is set to be x_i with probability ρ and sampled from (Ω_i, μ_i) with probability $1 - \rho$. Also we define the noise stability

$$S_\rho = \mathbf{E}_{x,x'}[f(x)f(x')] = \mathbf{E}_x[f(x)T_\rho f(x)]$$

We have the following facts.

Proposition 3.1.14.

$$T_\rho f = \sum \rho^{|\sigma|} \hat{f}(\sigma).$$

Proposition 3.1.15.

$$S_\rho f = \sum \rho^{|\sigma|} \hat{f}(\sigma)^2.$$

3.2 Advanced Probability Machineries

In the following, We introduce two tools in analyzing functions on product probability spaces.

3.2.1 Invariance Principle

The invariance principle [116] characterizes the asymptotic behaviour of low influence functions over product distribution.

There are multiple versions of the invariance principle and we state the using noisy-influences rather than low-degree influences (for an sketch of the proof, one can look at [126]).

Theorem 3.2.1. *For probability space $(\Omega, \mu) = (\prod_{i=1}^{n} \Omega_i, \mu_i)^n$ where each (Ω_i, μ_i) has an ensemble $\mathscr{X}_i = (\chi_{i,0}, ... \chi_{i,q-1})$. Let $\mathscr{G}_i = (g_{i,0} = 1, ... g_{i,q-1})$ follows the multivariate Gaussian distribution with their covariance matrix specified by the following "matching moments" condition: for any $i \in [n]$ and $j_1, j_2 \in [q]$*

$$\mathbf{E}[\chi_{i,j_1} \chi_{i,j_2}] = \mathbf{E}[g_{i,j_1}, g_{i,j_2}]. \tag{3.1}$$

$\mathscr{G}_1, \mathscr{G}_2, ... \mathscr{G}_n$ are all independent with each other.

Also the minimum atom probability among all (Ω, μ_i) is at least α. Let $f(\mathscr{X}_1, ..., \mathscr{X}_n)$ be a real-valued and assume that $\max_i \mathrm{Inf}_i T_{1-\epsilon} f \leq \tau$. Then for any function $\psi(t) : \mathbb{R} \to \mathbb{R}$ with bounded 3-rd derivative $|\psi'''(t)| \leq B$,

$$\left| \mathbf{E}[\psi(T_{1-\epsilon} f(\mathscr{X}_1, ..., \mathscr{X}_n))] - \mathbf{E}[\psi(T_{1-\epsilon} f(\mathscr{G}_1, ..., \mathscr{G}_n))] \right| \leq o_{\tau, \alpha, \epsilon}(1).$$

Here $\tau, \epsilon, \alpha, B$ are all constant independent of n.

Above invariance principle applies to functions defined on a single product probability space; later in [115] (also see [43]), Mossel generalized above result to vector valued functions and product of functions on correlated probability spaces. To state his results, first let us define the correlation between two probability spaces.

Definition 3.2.2. *Let $(\Omega \times \Theta; \mu)$ be a finite probability space. Define the correlation between Ω and Θ to be:*

$$\rho(\Omega, \Theta; \mu) = \sup\{\mathbf{Cov}[f, g] : f : \Omega \to \mathbb{R}, g : \Theta \to \mathbb{R}, \mathbf{Var}[f] = \mathbf{Var}[g] = 1\}.$$

The conditional operator U_μ associated with μ is a mapping from function space $\{f | f : \Theta \to \mathbb{R}\}$ to $\{g | g : \Omega \to \mathbb{R}\}$ defined as follows: for $f : \Theta \to \mathbb{R}$ and any $x_0 \in \Omega$ and random variable pair $x \in \Omega, y \in \Theta$ drawn from μ, $U_\mu f(x_0) = \mathbf{E}_y[f(y)|x = x_0]$.

We also define following quantity of the Gaussian Distribution.

Definition 3.2.3. *Let $\Phi(x)$ be the CDF function of one dimension Gaussian Distribution. g_1 and g_2 be bivariate Gaussian random variables with mean zero and covariance matrix $\begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$. For $\rho \in [-1, 1]$, we define $\overline{\Gamma}_\rho, \underline{\Gamma}_\rho : [0, 1]^2 \to [0, 1]$ by*

$$\overline{\Gamma}_\rho(\delta_1, \delta_2) = \mathbf{Pr}(g_1 \leq \Phi^{-1}(\delta_1) \wedge g_2 \leq \Phi^{-1}(\delta_2));$$
$$\underline{\Gamma}_\rho(\delta_1, \delta_2) = \mathbf{Pr}(g_1 \leq \Phi^{-1}(\delta_1) \wedge g_2 \geq \Phi^{-1}(1 - \delta_2)).$$

When $\delta_1 = \delta_2 = \delta$, we simplify the above notations by $\overline{\Gamma}_\rho(\delta)$ and $\underline{\Gamma}_\rho(\delta)$.

Following theorem is a generalization of the invariance principle for functions on correlated probability spaces [115]:

Theorem 3.2.4. *Let $\{(\Omega_i \times \Theta_i, \mu)\}_{i=1}^n$ be a collection of correlated probability spaces and assuming that the $\rho(\Omega_i, \Theta_i, \mu_i) \le \rho_0$ for each $i \in [n]$. The probability space $(\Omega \times \Theta, \mu)$ is defined to be $\prod_{i=1}^n (\Omega_i \times \theta_i, \mu_i)$ and functions $f : \Omega \to \mathbb{R}$ and $g : \Theta \to \mathbb{R}$ has the property that that $\mathbf{E}[f] = \delta_1$ and $\mathbf{E}[g] = \delta_2$. (Here the expectation is taken with respect to the marginal distribution of μ_1, μ_2 on Ω and Θ). Assume the minimum atom probability among all $(\Omega_i \times \Theta_i, \mu_i)$ for $i \in [n]$ is at least α. If f and g also satisfy the following influence property:*

$$\max_i \min(\mathrm{Inf}_i T_{1-\epsilon} f, \mathrm{Inf}_i T_{1-\epsilon} g) \le \tau.$$

then we have that

$$\underline{\Gamma}_{\rho_0}(\mu_1, \mu_2) + o_{\tau, \alpha, \epsilon}(1) \le \mathbf{E}[f \cdot g] \le \overline{\Gamma}_{\rho_0}(\delta_1, \delta_2) + o_{\tau, \alpha, \epsilon}(1)$$

3.2.2 Hypercontractivity

Hypercontractivity provides us another tool of analyzing functions defined on product of probability spaces.

Definition 3.2.5. *We say that a real random variable x is (p, q, η)-hypercontractive for $1 \le q \le p < \infty$ and $0 < \eta < 1$ if $\|x\|_p < \infty$, and for all $a \in \mathbb{R}$, $\|a + \eta x\|_p \le \|a + x\|_q$.*

For a discrete distribution, it is known to have the following hypercontractivity:

Theorem 3.2.6. *Let (Ω, μ) be a finite probability space with minimum atom probability α. Then every function $f : \Omega \to \mathbb{R}$ with $\mathbf{E}[f] = 0$ is $(2, p, \eta_p(\alpha))$ hypercontractive with*

$$\eta_p(\alpha) = \sqrt{\frac{A^{1/p} - A^{-1/p}}{A^{1/p'} - A^{-1/p'}}}$$

where $A = \frac{1-\alpha}{\alpha}$ and $1/p + 1/p' = 1$.

As for continuous distributions, following hypercontractivity theorem is known for the Gaussian Distribution:

Theorem 3.2.7. *Let $\mathcal{G}$ be a one-dimensional Gaussian Distribution, the $\mathcal{G}$ is $(2, q, 1/\sqrt{q-1})$-hypercontractive.*

Now we state the hypercontractivity theorem for low degree polynomials on product probability space

Theorem 3.2.8. *If a probability space (Ω, μ) is $(2, p, \eta)$ hypercontractive, then a Degree d polynomial $f : \Omega^n \to \mathbb{R}$ on probability space $((\Omega^n, \mu^n)$ is $(2, p, \eta^d)$ hypercontractive.*

In addition, the following "hypercontractive inequality" [23, 62] is known for functions applied with the noise operator.

Theorem 3.2.9. *Suppose $0 \le \rho \le 1$ and $q \ge 2$ satisfy that $\rho \le 1/\sqrt{(q-1)/(p-1)}$. Then for all $f : \{-1, 1\}^n \to \mathbb{R}$ and assume the distribution is uniformly random on $\{-1, 1\}^n$, then*

$$\|T_\rho f\|_q \le \|f\|_p.$$

For a large domain such as $[q]^n$, the optimal bound was first proved by Diaconis and Saloff-Coste [40]; the following uses their Theorems 3.5.ii and A.1 plus Hölder duality:

Theorem 3.2.10. *Let $q \geq 2$, $f : [q]^n \to \mathbb{R}$, and $0 \leq \epsilon < 1$. Also assume the distribution is uniform distribution on $[q]^n$. Then*

$$\|T_{\sqrt{1-\epsilon}}f\|_2 \leq \|f\|_p, \quad \text{where } p = p(q, \epsilon) = 1 + (1-\epsilon)^{(2-4/q)/\log(q-1)}.$$

Part II

CSPs and SDP

Chapter 4

Approximation Curve for Max Cut

4.1 Introduction

The MAX CUT is a Boolean CSP with the "$\neq$" constraints. It is also equivalent to the following graph problem. Given an undirected graph $G = (V, E)$, the Max Cut problem asks for a partition of the vertices into two sets so as to maximize the number of edges connecting the two sets. It is one of the classic NP-hard problems from Karp's list of 21 [88] and is arguably the simplest NP-hard problem. To cope with its NP-hardness and to understand hard instances, there has been a variety of work on its approximation algorithms. The greedy algorithm (or the random-assignment algorithm) is easily shown to have an approximation ratio of $\frac{1}{2}$ (see [129]). Goemans and Williamson [59] gave a SDP rounding algorithm achieving a .878 approximation ratio.[1] Since the early '90s, there is a large amount of interest in the SDP relaxation, in approximation algorithms, and in hardness of approximation for MAX CUT [3, 4, 15, 30, 36, 37, 47, 48, 50, 59, 73, 86, 99, 102, 107, 142]. In this Chapter, we build on the results in many of these papers and determine an essentially complete picture of the optimal approximation algorithms, SDP gaps, Dictator Tests, and UGC-hardness for MAX CUT.

4.1.1 Definitions

We begin with the basic definitions. We generally work with edge-weighted, undirected graphs $G = (V, E, w)$, where $w : E \to \mathbb{R}^{\geq 0}$ gives the nonnegative edge weights. The issue of self-loops turns out to be a nuisance; our policy will be to disallow them unless otherwise specified. Without loss of generality, we will always assume the edge weights sum to 1; i.e., $\sum_{e \in E} w(e) = 1$. Thus we can think of the weights as giving a probability distribution on edges; we will therefore omit w and think of E as a (symmetric) probability distribution on edges, writing $(u, v) \sim E$ to denote a draw from this distribution.

Definition 4.1.1. *A (proper) cut in G is a partition of the vertices into two parts, $h : V \to \{-1, 1\}$. The* value *of the cut is*

$$\mathrm{val}_G(h) = \Pr_{(u,v) \sim E}[h(u) \neq h(v)] = \mathop{\mathbf{E}}_{(u,v) \sim E}[\tfrac{1}{2} - \tfrac{1}{2}h(u)h(v)].$$

The MAX CUT *problem is the following: Given G, find a proper cut h with as large a value as possible.*

In general, we prefer the second definition of value given above, since it generalizes to *fractional cuts*:

Definition 4.1.2. *A fractional cut in G is a function $h : V \to [-1, 1]$. The* value *of the fractional cut is*

$$\mathrm{val}_G(h) = \mathop{\mathbf{E}}_{(u,v) \sim E}[\tfrac{1}{2} - \tfrac{1}{2}h(u)h(v)].$$

Given a fractional cut h, we can randomly produce a proper cut h' by setting each value $h'(v)$ to be 1 with probability $\frac{1}{2} + \frac{1}{2}h(v)$ and -1 with probability $\frac{1}{2} - \frac{1}{2}h(v)$, independently

---

[1]The SDP relaxation itself was given earlier by Delorme and Poljak [37], who noted it was polynomial-time computable.

across v's. In this way, $\mathbf{E}[h'(v)] = h(v)$. It follows that $\mathbf{E}[\mathrm{val}_G(h')] = \mathrm{val}_G(h)$ (although this uses the fact that G has no self-loops). Hence there always exists a proper cut h' with value at least $\mathrm{val}_h(G)$, and furthermore such a cut can easily be found deterministically from h using the method of conditional expectations. For these reasons, we will henceforth treat the MAX CUT problem as being about finding a *fractional* cut with as large a value as possible, and we will refer to fractional cuts simply as 'cuts'.

Definition 4.1.3. *The* optimum cut value*, or* MAX CUT*, for G is denoted*

$$\mathrm{Opt}(G) = \sup_{h:V \to [-1,1]} \mathrm{val}_G(h).$$

Note that the optimum is always at most 1 and at least $\frac{1}{2}$ (since the fractional cut $h \equiv 0$ is always available).

4.1.2 SDP Gaps of MAX CUT

All of the best approximation guarantees for MAX CUT currently known are achieved by algorithms using the SDP [37, 49, 59, 123]:

Definition 4.1.4. *The* (MAX CUT) SDP value *of a graph G is*

$$\mathrm{Sdp}(G) = \max_{g:V \to B_n} \mathop{\mathbf{E}}_{(u,v) \sim E} [\tfrac{1}{2} - \tfrac{1}{2} g(u) \cdot g(v)], \tag{4.1}$$

where $n = |V|$ and B_n denotes $\{x \in \mathbb{R}^n : \|x\| \le 1\}$. Note that $\mathrm{Sdp}(G) \ge \mathrm{Opt}(G)$, as g can always be taken to map into $[-1,1]$.

We should note that for graphs without self-loops, it is easy to see that the optimal embedding maps all vertices to the boundary of the ball.

Recall the following definition of SDP gap for Max Cut. Note that as there is a trivial way of finding a cut of value above $\frac{1}{2}$, we only consider (c,s)-gap for $c \ge s \ge \frac{1}{2}$.

Definition 4.1.5. *For $\frac{1}{2} \le s \le c \le 1$, we call the pair (c,s) an* SDP gap *if there exists a graph G with $\mathrm{Sdp}(G) \ge c$ and $\mathrm{Opt}(G) \le s$. We define the* SDP gap curve *by*

$$\mathrm{Gap}_{\mathrm{SDP}}(c) = \inf\{s : (c,s) \text{ is an SDP gap}\}.$$

Triangle inequalities. One can also consider strengthening the SDP by adding the 'triangle inequalities': i.e., enforcing

$$g(v_1) \cdot g(v_2) - g(v_2) \cdot g(v_3) - g(v_1) \cdot g(v_3) \ge -1,$$

$$g(v_1) \cdot g(v_2) + g(v_2) \cdot g(v_3) + g(v_1) \cdot g(v_3) \ge -1,$$

for all $v_1, v_2, v_3 \in V$. All of our positive results (rounding algorithms) will hold without the triangle inequalities, and we focus attention in this work almost exclusively on the basic SDP (4.1). However, we will also show that all of our negative results (SDP gaps, algorithmic limitations) hold even *with* the triangle inequalities.

We analogously define the curve $\mathrm{Gap}_{\triangle\mathrm{SDP}}$ for the SDP with the triangle inequalities. Of course, we have $\mathrm{Gap}_{\triangle\mathrm{SDP}}(c) \ge \mathrm{Gap}_{\mathrm{SDP}}(c)$ for all c.

4.1.3 RPR^2 Algorithms

The GW-algorithm's approximation curve is as follows:

$$\mathrm{Apx}_{GW}(c) \geq \begin{cases} \frac{1}{\pi}\arccos(1-2c) & \text{if } c \geq .844, \\ .878c & \text{if } c \leq .844. \end{cases}$$

There has been serval improvements to achieve a better approximation curve (particularly for $c < 0.844$). Generalizing the GW algorithm, Feige and Langberg [48] introduced the 'RPR^2' (Randomized Projection, Randomized Rounding) framework for rounding the solutions of SDP relaxations:

Definition 4.1.6. *An* RPR^2 algorithm *for* MAX CUT *is defined by a* rounding function, $r : \mathbb{R} \to [-1, 1]$. *Given a graph G, the steps of the algorithm are as follows:*

1. *Use SDP to find an optimal embedding $g : V \to S^{n-1}$ for the SDP* (4.1).
2. *Choose a random vector $\boldsymbol{Z} \in \mathbb{R}^n$ according to the n-dimensional Gaussian distribution.*
3. *Output the (fractional) cut $h : V \to [-1, 1]$ defined by $h(v) = r(g(v) \cdot \boldsymbol{Z})$.*

   (Certain implementation details of the RPR^2 method are discussed in Section 4.13.)

All of the known SDP algorithm for Max-Cut fall into the RPR^2 framework. For example, the GW algorithm is RPR^2 with rounding function $r(x) = \mathrm{sgn}(x)$; the random-assignment algorithm is RPR^2 with rounding function $r(x) \equiv 0$. Zwick's algorithm [142] is not obviously RPR^2, but it is shown to be so by Feige and Langberg [48]. In that paper, the authors suggest using 's-linear' rounding functions: i.e., functions of the form $r(t) = t/s$ if $-s \leq t \leq s$, $r(t) = 1$ if $t \geq s$, $r(t) = -1$ if $t \leq -s$. Charikar and Wirth's analysis [30] for $c = \frac{1}{2}$ indeed uses RPR^2 with s-linear rounding functions.

We conclude the discussion of RPR^2 algorithms by mentioning that, given an input graph G, it can be advantageous to try several different rounding functions r. It is well known (as discussed in Section 4.13) that given a collection $\mathscr{R}$ of rounding functions, one can achieve the performance of the best of them with running time slowdown only $O(|\mathscr{R}|\log|\mathscr{R}|)$. Indeed, Feige and Langberg even suggested the idea of trying 'all' possible rounding functions, up to some ϵ-discretization. Whether or not this achieves the performance of the 'optimal' rounding function up to an additive ϵ is a tricky issue which we discuss further in section 4.2.2.

4.1.4 Dictator Tests of "$\neq$"

For MAX CUT one needs a dictator test making only 2 queries and testing $f(x) \neq f(y)$. The rule of thumb is that giving a such a test with 'completeness' c and 'soundness' s may allow one to derive a c vs. s inapproximability result. (We give concrete theorems along these lines later in this section).

Let us briefly recall some of the relevant definitions:

Definition 4.1.7. *A 2-query, $\neq$-based Dictator Test for functions $f : \{-1,1\}^n \to \{-1,1\}$ is a randomized procedure for choosing two strings $x, y \in \{-1,1\}^n$. We think of the test as querying $f(x)$ and $f(y)$, and then* accepting *when $f(x) \neq f(y)$, and* rejecting *otherwise.*

Definition 4.1.8. *The* completeness *of a Dictator test T for n-bit functions is*

$$\text{Completeness}(T) = \min_{i \in [n]} \{\mathbf{Pr}[T \ accepts \ \chi_i]\},$$

where $\chi_i : \{-1,1\}^n \to \{-1,1\}$ is the ith 'Dictator' function, $\chi_i(x) = x_i$.

As for the soundness, we defer the formal explanation to section 4.7; for now, suffice it to say we make a definition along the following lines:

Definition 4.1.9. *(informal) The* soundness *of a Dictator Test T for functions $f : \{-1,1\}^n \to [-1,1]$ is*

$$\text{Soundness}(T) = \max\{\mathbf{Pr}[T \ accepts \ f] : f \ is \ 'quasirandom'\}.$$

In addition to the unspecified notion 'quasirandom', the reader will notice that we have generalized to testing functions whose range is $[-1,1]$ rather than $\{-1,1\}$. The reason for doing this is that all the applications we present require this generalized setting. The distinction is similar to the one between proper and fractional cuts. Again, formal definitions appear in Section 4.7.

Definition 4.1.10. *(informal) We call the pair (c,s) a* dictator-vs.-quasirandom gap *if for all $\eta > 0$, for sufficiently large n there is a dictator-vs.-quasirandom test $T^{(n)}$ for functions $f : \{-1,1\}^n \to [-1,1]$ with $\text{Completeness}(T^{(n)}) \geq c$ and $\text{Soundness}(T^{(n)}) \leq s + \eta$. We define the* dictator-vs.-quasirandom gap curve *by*

$$\text{Gap}_{\text{Test}}(c) = \inf\{s : (c,s) \ is \ a \ dictator\text{-}vs.\text{-}quasirandom \ gap\}.$$

As mentioned, our interest in dictator-vs.-quasirandom tests comes from their application to algorithmic hardness results. We give three such applications here. The first is the original application, implicitly proved in [99]:

Theorem 4.1.11 ([99])**.** *Suppose (c,s) is a dictator-vs.-quasirandom gap, and $\eta > 0$. Then the UGC (UGC) implies that it is NP-hard to distinguish* MAX CUT *instances with value at least $c - \eta$ from instances with value at most $s + \eta$. I.e., assuming the UGC and $\text{P} \neq \text{NP}$ we essentially have $\text{Apx}_A(c) \leq \text{Gap}_{\text{Test}}(c)$ for all efficient algorithms A and all c.*

(The 'essentially' here refers to the fact that we really only have $\text{Apx}_A(c - \eta) \leq \text{Gap}_{\text{Test}}(c)$ for all $\eta > 0$. Ultimately we will show that Gap_{Test} is continuous, so this distinction is irrelevant.)

4.1.5 Motivation and Discussion

In this section we discuss the motivation and merits of deciding the optimal approximability curve of MAX CUT for every values of c.

First, MAX CUT is a fundamental algorithmic problem; indeed, it is arguably the simplest NP optimization problem. For the reasons discussed in section 4.1.1, we feel that

understanding its approximability for the entire range of c is important. We are hardly alone in this regard; for example, in 2001 Feige and Langberg [48] wrote that they were "trying to extend the techniques of [50] in order to prove [that RPR^2 algorithms can match the SDP gap curve for values of $c < .844$]". Besides the algorithmic work on the MAX CUT curve we've already described [30, 48, 59, 142], there has also been a great deal of work recently on the very related problem of the Max-2Lin [1, 2, 8, 17, 77]. For example the Grothendieck/Quadratic Programming results of [1, 2, 30] are nothing more than analysis of the Max-2Lin approximability curve at $\frac{1}{2} + \epsilon$ — with the underlying graph structure fixed to be bipartite, in the Grothendieck case. Further, analyzing the MAX CUT/Max-2Lin approximability curves at $1 - \epsilon$ for *subconstant* ϵ is very strongly related to analyzing Sparsest-Cut approximability.

Further, the fundamental nature of the MAX CUT problem makes our inability to understand its computational complexity all the more galling. Recall that every value of c for which we don't know the largest efficiently achievable value of $\text{Apx}_A(c)$ yields a basic, natural problem not known to be in P and not known to be NP-hard: e.g., "Given a graph with a cut of size 60%, find a cut of size 55%". Without the UGC, it seems we have no idea how to prove sharp inapproximability results, although in this work we did the best we could by ruling out RPR^2 algorithms from achieving $\text{Apx}(c) > S(c)$. Assuming the UGC, though, the present work completely closes the MAX CUT problem. Even if one does not believe the UGC, there are several takeaways: First, we've shown that the UGC cannot be disproved by giving good MAX CUT SDP rounding algorithms, for any value of c. Second, our work gives an improved approximation *algorithm* inspired by UGC/dictator-vs.-quasirandom test considerations.

Finally, we hope that the methods developed— specifically, the use of Hermite analysis, von Neumann's Minimax Theorem, Borell's rearrangement inequality [24], and the Karush-Kuhn-Tucker conditions — can be used to make progress on understanding SDP gaps and approximability of other fundamental problems. Specifically, we believe our methods should be useful for attacking Max-2Sat and other 2-CSPs (some indication of this is given already in the recent work of Austrin [13, 14]), 3-CSPs, and perhaps even for determining the Grothendieck constant [63].

4.1.6   Statement of Main Results

Our first result, from which the remaining results derive, is a complete determination of the SDP gap curve. We introduce an explicit function $S : [\frac{1}{2}, 1] \to [\frac{1}{2}, 1]$, and show that $\text{Gap}_{\text{SDP}}(c) = S(c)$ for all c. In particular, the proof of the lower bound, $\text{Gap}_{\text{SDP}}(c) \geq S(c)$, is achieved via a poly(n)-time RPR^2 algorithm. Thus we have an efficient algorithm for MAX CUT which has optimal SDP-approximation curve. The fact that an RPR^2 algorithm achieves the SDP gap confirms a conjecture suggested by Feige and Langberg [48].

Next, we show how to transform the SDP results into dictator-vs.-quasirandom testing results. Specifically, we are able to show that the dictator-vs.-quasirandom gap curve is identical to the SDP gap curve; i.e, $\text{Gap}_{\text{Test}}(c) = S(c)$ for all $c \in [\frac{1}{2}, 1]$. This result gives us

optimal dictator-vs.-quasirandom tests. In addition:

- The SDP gap curve with triangle inequalities, $\text{Gap}_{\triangle\text{SDP}}$, is also identical to the curve S.

- If A is any RPR^2 algorithm then $\text{Apx}_A(c) \leq S(c)$ for all c, even assuming both of the following: (i) A uses the SDP with triangle inequalities; (ii) A is not required to choose $\boldsymbol{Z}$ to be a random n-dimensional Gaussian, but rather is allowed to deterministically select the best $\boldsymbol{Z}$ satisfying $\|\boldsymbol{Z}\| = \Theta(\sqrt{n})$. (Contrast this with the fact that in graphs exhibiting the c vs. $S(c)$ SDP gap, our RPR^2 algorithm actually finds an essentially optimal cut.)

- If A is *any* polynomial-time algorithm then $\text{Apx}_A(c) \leq S(c)$ for all c, assuming $\text{P} \neq \text{NP}$ and the UGC.

4.1.7 The Critical Curve, S

At this point the reader might wish to know the identity of this critical curve $S(c)$. Unfortunately, there is no 'nice' formula for it. Rather, it is defined as follows:

$$S(c) \quad = \quad \inf_{\substack{(1,\rho_0)\text{-distributions } P \\ \text{with mean } 1-2c}} \quad \sup_{\substack{r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \text{val}_{\mathcal{G}_P}(r). \tag{4.2}$$

Not all of the expressions above have even been defined yet — in particular '$(1,\rho_0)$-distribution' (a certain simple kind of probability distribution on $[-1,1]$) and '$\mathcal{G}_P$' (a certain infinite graph). Further, on the face of it this definition does not look very 'explicit', especially since the inf and sup are both over infinite sets. Nevertheless, in section 4.5 we prove the following:

Theorem 4.1.12. *There is an algorithm that, on input $c \in [\frac{1}{2}, 1]$ and $\epsilon > 0$, runs in time* poly$(1/\epsilon)$ *and computes $S(c)$ to within $\pm\epsilon$.*

We believe this justifies our claim that S is 'explicitly given'. A brief discussion of this point appears in section 4.6.1.

In fact, as we will describe in the next section, significant portions of $S(c)$ can be described or estimated more simply. For $c \geq .844$, $S(c)$ agrees with the Goemans-Williamson SDP-approximation curve, $\frac{1}{\pi}\arccos(1-2c)$. For $c = \frac{1}{2} + \epsilon$, $S(c) \approx \frac{1}{2} + \frac{1}{2}\cdot\epsilon/\ln(1/\epsilon)$ up to lower-order terms (this is proved in Section 4.14, tightening the asymptotics of [30, 102]). A plot of $S(c)$ versus c appears in Section 4.15.

4.1.8 Prior Work

Surveying the entirety of the previous work on approximation algorithms, SDP gaps, and hardness results for MAX CUT would take several pages, so we restrict ourselves to briefly summarizing the best results known prior to this work.

SDP and Dictator Testing gaps. Combining prior work of many authors yields the following:

1. For $c \geq .844$: $\text{Gap}_{\text{SDP}}(c) = \text{Gap}_{\triangle\text{SDP}}(c) = \text{Gap}_{\text{Test}}(c) = \frac{1}{\pi}\arccos(1-2c)$.

2. For $c = \frac{1}{2}+\epsilon$: $\text{Gap}_{\text{SDP}}(c)$, $\text{Gap}_{\triangle\text{SDP}}(c)$, and $\text{Gap}_{\text{Test}}(c)$ all have asymptotics $\frac{1}{2}+\Theta(\epsilon/\ln(1/\epsilon))$.

As can be seen, this already pins down substantial portions of these curves fairly well. In the next section we will argue the merits of pinning them down precisely.

The lower bound $\text{Gap}_{\text{SDP}}(c) \geq \frac{1}{\pi}\arccos(1-2c)$ for $c \geq .844$ is, as mentioned, due to Goemans and Williamson [59], using RPR^2 with the rounding function sgn. The matching upper bound is due to Feige and Schechtman [50], using infinite graphs with vertex set S^{n-1} and edge set connecting all vectors with inner product at most $1-2c$. The lower bound $\text{Gap}_{\text{SDP}}(c) \geq \frac{1}{2} + \Omega(\epsilon/\ln(1/\epsilon))$ is due to Charikar and Wirth [30], using RPR^2 with s-linear rounding functions, as suggested by Feige and Langberg [48]. The upper bound $\text{Gap}_{\text{SDP}}(c) \leq \frac{1}{2} + O(\epsilon/\ln(1/\epsilon))$ is due to Khot and O'Donnell [102], using mixtures of correlated Gaussian graphs (described in section 4.2.2). As mentioned, we tighten the asymptotics of the previous two results in Section 4.14. Finally, Feige and Langberg showed some additional numerical lower bounds for $\text{Gap}_{\text{SDP}}(c)$, via RPR^2 with s-linear rounding functions; e.g., $\text{Gap}_{\text{SDP}}(.6) \geq .5477$.

The upper bound $\text{Gap}_{\text{Test}}(c) \leq \frac{1}{\pi}\arccos(1-2c)$ actually holds for all $c \in [\frac{1}{2}, 1]$; this was conjectured by Khot, Kindler, Mossel, and O'Donnell [99] and proved by Mossel, O'Donnell, and Oleszkiewicz [116]. The 'noise sensitivity' test from [99] involves choosing $x \in \{-1,1\}^n$ uniformly at random and choosing y by flipping each coordinate of x with probability c. (As we will discuss in section 4.10, this construction is quite similar to one introduced by Karloff [86] and analyzed further in [3, 4].) The upper bound $\text{Gap}_{\text{Test}}(\frac{1}{2}+\epsilon) \leq \frac{1}{2} + O(\epsilon/\ln(1/\epsilon))$ was proved by Khot and O'Donnell [102], by mixing together two tests of the type in [99]. The remaining parts of the above statements implicitly follow from Khot and Vishnoi [107]. Interestingly, although proving lower bounds for $\text{Gap}_{\text{Test}}(c)$ is a very natural problem from the point of view of Property Testing, it doesn't seem to have been explicitly been considered in the literature. Indeed, using the Khot-Vishnoi result is a very circuitous way to prove Dictator Testing lower bounds. We discuss this point further in section 4.9.

Algorithmic hardness. Early results on algorithmic hardness involved showing upper bounds on the approximation curve of specific algorithms. In particular, work of Karloff [86], Alon and Sudakov [3], and Alon, Sudakov, and Zwick [4] showed that for the GW algorithm, $\text{Apx}_{GW}(c) \leq \frac{1}{\pi}\arccos(1-2c)$, where $\text{Apx}_{GW}(c)$ denotes the *expected* performance, over $\mathbf{Z}$, of the GW algorithm. Further, this result holds even if one adds all 'valid' constraints to the SDP. As we describe in section 4.10, these results can be seen as very weak forms of dictator-vs.-quasirandom tests. Feige and Schechtman [50] extended these results to the case where the algorithm can pick *any* halfspace cut (although only under the triangle inequalities, not any valid constraints). Assuming the UGC, [99]'s Theorem 4.1.11 implies NP-hardness of achieving approximation curve exceeding $\text{Gap}_{\text{Test}}(c)$. The best unconditional NP-hardness result is much weaker: Håstad [76] together with Trevisan, Sorkin, Sudan, and Williamson [138] showed that achieving $\text{Apx}(\frac{17}{21}) > \frac{16}{21}$ is NP-hard; it is easy to translate this into hardness of $\text{Apx}(\frac{1}{2}+\epsilon) > \frac{1}{2} + \frac{11}{13}\epsilon$ for $\epsilon \leq \frac{13}{42}$ and hardness

of $\text{Apx}(1-\epsilon) > 1 - \frac{5}{4}\epsilon$ for $\epsilon \leq \frac{4}{21}$.

4.1.9 Comparison with Raghavendra's Result

In an independent powerful work obtained by Raghavendra [125], he established a equivalent relationship (with $o(1)$ slack) between $\text{Gap}_{\text{SDP}}(c)$ and the optimal approximation $\text{Alg}^{\text{optimal}}(c)$ as weell as $\text{Gap}_{\text{Test}}(c)$ for almost every CSP with bounded arity. In addition, he also gave an algorithm of calculating the Gap_{SDP} with running time $\exp(\exp(\Omega(1/\epsilon)))$ and an optimal SDP rounding algorithm (assuming UGC) with running time $\text{poly}(n) \cdot \exp(\exp(\Omega(1/\epsilon)))$ (also see [127]). Compared with [125], one main advantage of our work on the problem of MAX CUT is that we have a much better running time on SDP rounding and SDP gap calculation. This allows us to explicitly determine the actually value of the SDP gap as well as the optimal approximation curve for the Max Cut problem. In addition, our work has a concrete construction of the worst SDP gap instance is: it is certain $(1, \rho_0)$ Gassuian Mixture graph, which will be defined later.

4.2 Proof Overview

In this section we describe the ideas and intuition underlying the determination of Gap_{SDP}. By the end of the section we will also have defined all the terms necessary for the definition (4.2) of the curve $S(c)$.

4.2.1 Embedded Graphs

The first idea is to slightly shift the way one looks at SDP gaps for MAX CUT. Usually one thinks of first finding a graph G, then showing $\text{Sdp}(G)$ is large and $\text{Opt}(G)$ is small. But suppose one determines that $\text{Sdp}(G)$ is large for some graph G; then one *may as well identify G with its optimal SDP embedding on the sphere*.

Definition 4.2.1. *An (n-dimensional) embedded graph G is one whose vertex set V is a subset of S^{n-1}. For embedded graphs, we explicitly* allow *self-loops.[2] The ρ-distribution of the embedded graph, denoted $P = P(G)$, is the discrete probability distribution on $[-1, 1]$ given by the distribution of $u \cdot v$ when $(u, v) \sim E$. We define the* spread *of G (which we also call the spread of P) to be*

$$\text{Spread}(G) = \text{Spread}(P) = \mathop{\mathbf{E}}_{\rho \sim P}[\tfrac{1}{2} - \tfrac{1}{2}\rho] \in [0, 1].$$

Thinking about embedded graphs leads to some important observations. The first is that we can *symmetrize* any SDP gap instance. Specifically, let G be an embedded graph with $\text{Spread}(G) = c$ and $\text{Opt}(G) \leq s$. Suppose $\mathcal{O}$ is any rotation of space; then it is clear that the rotated embedded graph $\mathcal{O}G$ also satisfies $\text{Spread}(\mathcal{O}G) = c$ and $\text{Opt}(\mathcal{O}G) \leq s$, and is thus an equally good gap instance. Further, if one takes a mixture $H = \lambda G + (1 - \lambda)G'$ of any two embedded graphs G and G' with $\text{Spread}(G) = \text{Spread}(G') = c$ and $\text{Opt}(G), \text{Opt}(G') \leq s$,

---

[2]Although we disallow self-loops in MAX CUT inputs, we allow them in embedded graphs. One reason for this is that there is no guarantee that every optimal SDP embedding $g : V \to S^{n-1}$ is injective.

then Spread(H) is again c, and also Opt(H) $\leq s$ by a simple averaging argument. Hence we can average an SDP gap instance G over *all* rotations of space, and preserve the gap. When we do this we get an 'infinite embedded graph' whose vertex set is all of S^{n-1} and whose edge distribution is 'symmetric', in the sense that the density on the pair (u,v) depends only on the inner product $u \cdot v$. In fact, the 'ρ-distribution' of the symmetrized graph is precisely the original ρ-distribution $P(G)$.

Definition 4.2.2. *Let P denote any discrete probability distribution on $[-1,1]$. We define the d-dimensional symmetric embedded graph $\mathscr{S}_P^{(d)}$ to be the embedded graph with vertex set S^{d-1} and edge distribution over $S^{d-1} \times S^{d-1}$ given by drawing a random pair of unit vectors with inner product ρ, where ρ itself is drawn from P.*

Thus we have reduced the search for graphs with large SDP gap to the search for ρ-distributions P such that Spread(P) $= c$ (i.e., the mean of P is $1-2c$) but Opt($\mathscr{S}_P^{(d)}$) is small. Indeed, Feige and Schechtman's SDP gap instance [50] is precisely of this form; roughly speaking, they take P to be the distribution with all of its mass concentrated on $1-2c$.

Unfortunately, analyzing Opt($\mathscr{S}_P^{(d)}$) is not so easy; we will come back to the problem later. For now let us move to the algorithmic side of things. We have seen that we can reduce the problem of finding large SDP gaps to studying symmetric embedded graphs. Can we similarly reduce the problem of finding large cuts in arbitrary graphs to studying symmetric embedded graphs? The observation here is that, in some sense, this is just what the RPR^2 algorithm is doing. Consider the steps of the algorithm from Definition 4.1.6. RPR^2 algorithms do not use the fact that the SDP solution they operate on is optimal; hence we can mentally dispense with Step 1 (SDP) and view RPR^2 algorithms as simply taking an embedded graph G as input and trying to find a large cut in it. Next, recalling that the d-dimensional Gaussian distribution is spherically symmetric, we see that the RPR^2 algorithm can, at a rough level, be thought of as: (i) implicitly constructing the symmetrized version of G; and then, (ii) outputting the 'one-dimensional' fractional cut r. We will make this idea more precise in the next section. For now, we note that if RPR^2 algorithms are to achieve the SDP gap, it must in some sense be the case that optimal cuts in symmetric embedded graphs $\mathscr{S}_P^{(d)}$ are 'one-dimensional'. The key to our determination of Gap$_{\text{SDP}}(c)$ is showing that this statement is sufficiently true.

4.2.2 Gaussian Mixture Graphs

By now our analysis is heavily dependent on understanding Opt($\mathscr{S}_P^{(d)}$), where P is a distribution with mean $1-2c$. I.e., we want to determine

$$\sup_{h:S^{d-1}\to[-1,1]} \quad \mathbf{E}_{\rho\sim P} \quad \mathbf{E}_{\substack{(u,v)\sim S^{d-1}\times S^{d-1} \\ \text{with } \langle u,v\rangle = \rho}} \quad [\tfrac{1}{2} - \tfrac{1}{2}h(u)\cdot h(v)].$$

This is somewhat complicated by the fact the distribution on vertices — i.e., the uniform distribution on the surface of the sphere — is not a product distribution, and depends in a nontrivial way on the dimension d. It is possible to at once avoid this difficulty *and* hew much more closely to the RPR^2 framework by replacing the uniform distribution on S^{d-1} by the d-dimensional *Gaussian* distribution.

Definition 4.2.3. *Let P denote any discrete probability distribution on $[-1, 1]$. We define the d-dimensional Gaussian mixture graph $\mathscr{G}_P^{(d)}$ to be the probability measure on $\mathbb{R}^d \times \mathbb{R}^d$ given by drawing a pair of 'ρ-correlated d-dimensional Gaussians', where ρ itself is drawn from P. In the case $d = 1$, we simply write $\mathscr{G}_P$. By ρ-correlated d-dimensional Gaussians we mean a pair $(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x}$ is a standard d-dimensional Gaussian and $\boldsymbol{y} \sim \rho \boldsymbol{x} + \sqrt{1 - \rho^2}\boldsymbol{Z}$, with $\boldsymbol{Z}$ being another d-dimensional Gaussian independent of $\boldsymbol{x}$. Note that this distribution is symmetric in $\boldsymbol{x}$ and $\boldsymbol{y}$.*

Gaussian mixture graphs, with P concentrated on 1 and $-\frac{1}{2}$, were introduced in [102] to show SDP gaps for c near $\frac{1}{2}$.

Regarding the effect of switching from $\mathscr{S}_P^{(d)}$ to $\mathscr{G}_P^{(d)}$, recall that the Gaussian distribution in a high dimension d is very similar to the uniform distribution on the sphere of radius $\sqrt{d}$. Using this fact, it is not too hard to show that when $\text{Spread}(P) = c$ we have $\text{Sdp}(\mathscr{G}_P^{(d)}) \geq c - o_d(1)$, via the embedding $x \mapsto x/\|x\|$. Thus we can equally well search for SDP gaps based on Gaussian mixture graphs. As for algorithms, the RPR^2 framework now has a very simple interpretation: Given an embedded graph G with ρ-distribution P, the RPR^2 algorithm implicitly converts it to $\mathscr{G}_P$ and cuts it with the rounding function r. More specifically, the expected value of the cut produced by RPR^2 on graph G is:

$$
\begin{aligned}
\text{Alg}_{\text{RPR}^2}(G) \ &= \ \mathop{\mathbf{E}}_{\boldsymbol{Z}}\left[\mathop{\mathbf{E}}_{(u,v)\sim E}[\tfrac{1}{2} - \tfrac{1}{2}r(u \cdot \boldsymbol{Z})r(v \cdot \boldsymbol{Z})] \right] \\
&= \ \mathop{\mathbf{E}}_{\substack{\rho \sim P(G)}} \mathop{\mathbf{E}}_{\substack{(x,y)\ \rho\text{-corr'd} \\ \text{1-dim Gaussians}}}[\tfrac{1}{2} - \tfrac{1}{2}r(x)r(y)] \ = \ \text{val}_{\mathscr{G}_P^{(1)}}(r). \quad (4.3)
\end{aligned}
$$

The reader can now see that given G, an RPR^2 algorithm should strive to take r to be the optimal cut $r : \mathbb{R} \to [-1, 1]$ for $\mathscr{G}_P$ (i.e., $\mathscr{G}_P^{(1)}$). This leads us to two questions:

1. Can we *algorithmically* determine an r which gives a near-optimal cut for $\mathscr{G}_P$?

2. Whether or not we can, would this be enough to match the SDP gap? In other words, is it true that for all ρ-distributions P with spread $c \in [\frac{1}{2}, 1]$,

$$
\text{Opt}(\mathscr{G}_P) \ \geq \ \inf_{P' \text{ with mean } 1 - 2c} \text{Opt}(\mathscr{G}_{P'}^{(d)})? \quad (4.4)
$$

Here the left-hand side represents what we hope to achieve algorithmically with RPR^2, and the right-hand side represents the upper-bound on $\text{Gap}_{\text{SDP}}(c)$ we can achieve using Gaussian mixture graphs.

Question 2 above is the heart of the matter; we describe its affirmative answer in the next section. For now, let us discuss Question 1. Although analytically we don't know the optimal cut for $\mathscr{G}_P$, there is a feeling that one could algorithmically find an r coming within ϵ of the optimum by using the Feige-Langberg idea of trying 'all' possible r, suitably discretized. Indeed, Feige and Langberg wrote that if one only considers 'well-behaved' rounding functions r (suggesting piecewise differentiable functions with bounded derivatives) then one can construct a collection of $2^{\text{poly}(1/\epsilon)}$ many discretized rounding functions such that one of them achieves a cut in $\mathscr{G}_P$ that is within ϵ of that achieved by the best

well-behaved rounding function.

Unfortunately, there is no guarantee that the optimal cut for $\mathscr{G}_P$ is is 'well-behaved'. Even if it were guaranteed to be piecewise differentiable, we have no way of proving that its derivatives don't depend on 'n'; i.e., the number of points in P's support. Thus we do not know of any way of efficiently (in n) discretizing the search space for the optimal rounding function of a given $\mathscr{G}_P$. But luckily, in the next section we will see that for the 'worst' P, there is a relatively well-behaved optimal cut r; specifically, there is an *increasing* optimal cut. The fact that increasing functions are $O(1/\epsilon)$-Lipschitz except on a set of measure ϵ means it will be sufficient to discretize the set of rounding functions r in a way depending only on ϵ and not on n. Indeed, our actual algorithm for finding cuts of size at least $S(c) - \epsilon$ in graphs G with $\mathrm{Sdp}(G) \geq c$ is:

Algorithm 4.2.4. *Perform the RPR^2 algorithm, trying out all $2^{\tilde{O}(1/\epsilon^2)}$ possible 'ϵ-discretized' rounding functions r.*

The definition of 'ϵ-discretized' is given in section 4.4. A discussion of the running time, $\mathrm{poly}(|V|) \cdot 2^{\tilde{O}(1/\epsilon^2)}$, appears in section 4.6.2.

4.2.3 Hermite Analysis, Minimax, and Borell's Gaussian Rearrangement

We now come to the main conceptual part of the determination of $\mathrm{Gap}_{\mathrm{SDP}}$, namely proving (4.4). Suppose we could show that for every P', there was an optimal cut f for $\mathscr{G}_{P'}^{(d)}$ that was 'one-dimensional' — i.e., of the form $f(\boldsymbol{x}) = r(\boldsymbol{u} \cdot \boldsymbol{x})$, where $r : \mathbb{R} \to [-1, 1]$ and $\boldsymbol{u}$ is any unit vector. It's easy to see that the value of f in $\mathscr{G}_{P'}^{(d)}$ is just $\mathrm{val}_{\mathscr{G}_{P'}}(r)$; hence we would show $\mathrm{Opt}(\mathscr{G}_{P'}^{(d)}) = \mathrm{Opt}(\mathscr{G}_{P'})$, proving (4.4). Unfortunately, we do not know whether this is the case. What we *will* show, though, is that when P' is the 'worst' distribution, $\mathscr{G}_{P'}^{(d)}$ has an optimal one-dimensional (and increasing, as promised) cut.

To start, we take advantage of our switch to Gaussian graphs; this allows us to express the value of cuts $f : \mathbb{R}^d \to [-1, 1]$ using 'Hermite analysis' (akin to Fourier analysis over $\{-1, 1\}^n$). Specifically, given a cut f one has

$$\mathrm{val}_{\mathscr{G}_P^{(d)}}(f) = \tfrac{1}{2} - \tfrac{1}{2} \mathop{\mathbf{E}}_{\rho \sim P} \left[\sum_{S \in \mathbb{N}^d} \widehat{f}(S)^2 \rho^{|S|} \right], \tag{4.5}$$

where each $\widehat{f}(S) \in \mathbb{R}$ is a 'Hermite coefficient', and $|S|$ denotes $\sum_{i=1}^{d} S_i$. Using this formula one can easily show that any optimal cut f may as well be odd; i.e., satisfy $f(-\boldsymbol{x}) = -f(\boldsymbol{x})$. Further, when f is odd, the sum in (4.5) can be restricted to only be over S's such that $|S|$ is odd.

We now make the following observation: For fixed odd f, the expression $\mathbb{S}_f(\rho) := \sum_{|S| \text{ odd}} \widehat{f}(S)^2 \rho^{|S|}$ is a polynomial in ρ (power series, actually) with nonnegative coefficients and only odd powers. This means that it is convex for $\rho \geq 0$ and concave for $\rho \leq 0$. Now suppose we keep f fixed but vary the ρ-distribution P, subject only to it having mean $1 - 2c$.

66

Using formula (4.5), one sees that we can make $\mathrm{val}_{\mathscr{G}_P^{(d)}}(f)$ as low as the value of the convex lower envelope of $\frac{1}{2} - \frac{1}{2}\mathbb{S}_f(\rho)$ at $1 - 2c$. Further, by the convexity/concavity described, one achieves this by concentrating all of P's probability mass on at most two points: some negative number ρ_0, and possibly also 1.

Definition 4.2.5. *We call a discrete probability distribution P on $[-1, 1]$ a $(1, \rho_0)$-distribution if P puts positive probability on some $-1 \leq \rho_0 \leq 0$, nonnegative probability on 1, and zero probability on all other values in $[-1, 1]$.*

These considerations suggest that the Gaussian mixture graphs with lowest MAX CUT are those based on $(1, \rho_0)$-distributions. This doesn't constitute a proof, though, because we fixed the cut and the graph in the wrong order: we are supposed to fix the distribution P first and then choose the optimal cut. Ultimately, though, we prove that $(1, \rho_0)$-distributions *are* the worst case for Gaussian mixture graphs by using the von Neumann Minimax Theorem: we can reverse the order of fixing the distribution and the cut if we allow the 'cut Player' to choose a distribution on cuts. Fortunately, the convex combination of $\mathbb{S}_f(\rho)$ polynomials has the same convexity/concavity properties as a single one, so the previous argument goes through. Unfortunately, one also has to overcome some rather severe discretization/compactness complications to use the von Neumann Theorem in this infinitary setting.

At this point we essentially have that the Gaussian mixture graphs with smallest MAX CUT are those based on $(1, \rho_0)$-distributions. Finally, we are able to deduce that in such graphs there are optimal, one-dimensional, increasing cuts through the use of Borell's rearrangement inequality for Gaussian space [24]. Borell's theorem implies that for $\rho \in [0, 1]$, the quantity $\mathbb{S}_f(\rho)$ can only increase if one 'rearranges' f's values into an increasing, one-dimensional function. If $G = \mathscr{G}_P^{(d)}$ is a Gaussian mixture graph with P a $(1, \rho_0)$-distribution, then formula (4.5) tells us that $\mathrm{val}_G(f)$ is (up to an additive $\frac{1}{2}$) a negative linear combination of $\mathbb{S}_f(1)$ and $\mathbb{S}_f(\rho_0)$. It turns out that $\mathbb{S}_f(1)$ is just $\mathbf{E}[f^2]$, which doesn't change under rearrangement, and when f is odd $\mathbb{S}_f(\rho_0) = -\mathbb{S}_f(-\rho_0)$; hence Borell implies that this quantity decreases under rearrangement. This proves that indeed there is a one-dimensional and increasing optimal cut.

Thus we establish that (4.4) holds and that the right-hand side in that inequality is precisely $S(c)$.

4.2.4 Organizations of the Remaining Proof

Above is just a high level overview of the proof. The missing part is organized as follows: The construction of optimal dictator-vs.-quasirandom tests from Gaussian mixture graphs mimics the proof of the Majority Is Stablest theorem using the 'Invariance Principle' from [116]; the poly($1/\epsilon$)-time algorithm for computing $S(c)$ within ϵ, promised in Theorem 4.1.12, involves combining the Karush-Kuhn-Tucker conditions with Borell's theorem; and, the remaining work involves careful discretization arguments.

4.3 $\text{Gap}_{\text{SDP}}(c) \leq S(c)$: **Hermite Analysis and Borell's Rearrangement**

In this section we prove $\text{Gap}_{\text{SDP}}(c) \leq S(c)$; i.e., we show that for each $c \in [\frac{1}{2}, 1]$ and $\eta > 0$, there exists a graph G exhibiting a large SDP gap: $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq S(c) + \eta$. We remind the reader here of the definition of $S(c)$:

$$S(c) \quad = \inf_{\substack{(1, \rho_0)\text{-distributions } P \\ \text{with mean } 1 - 2c}} \quad \sup_{\substack{r: \mathbb{R} \to [-1, 1] \\ \text{increasing, odd}}} \quad \text{val}_{\mathscr{G}_P}(r).$$

4.3.1 SDP Gaps via Gaussian Mixture graphs

As described in sections 4.2.2 and 4.2.3, the graphs we use to exhibit SDP gaps will be high-dimensional Gaussian mixture graphs based on $(1, \rho_0)$-distributions. Since these are infinite graphs, we will need to extend a number of our basic definitions, including '$\text{Sdp}(G)$' and '$\text{Opt}(G)$'. The reader may object that these will not proper SDP gap examples because the graphs are infinite and also have self-loops (one might even object that the graphs are weighted). However in Section 4.12 we show that these issues can be circumvented:

Proposition 4.3.1. *Suppose $G = \mathscr{G}_P^{(d)}$ is a Gaussian mixture graph with $\text{Sdp}(G) \geq c$ and $\text{Opt}(G) \leq s$. Then for any $\epsilon > 0$, there is a finite, self-loopless, unweighted graph G' (with $n = (1/\epsilon)^{O(d)}$ vertices) with $\text{Sdp}(G') \geq c - \epsilon$ and $\text{Opt}(G') \leq s + \epsilon$.*

The proof of this proposition essentially only uses straightforward, already-known ideas [8, 50, 102]. The reader should also note that arbitrarily small losses in c are also immaterial, since we can show (essentially a priori) that $\text{Gap}_{\text{SDP}}(c)$ is continuous:

Proposition 4.3.2. *The function Gap_{SDP} is continuous on $[\frac{1}{2}, 1]$, and strictly increasing from $\frac{1}{2}$ to 1.*

The proof of this proposition is in Section 4.11.

Extending the basic MAX CUT definitions to infinite graphs is quite straightforward; see [102]. Here we will just treat the special case of Gaussian mixture graphs, which require a little extra care due to the fact that they can have 'self-loops'. To begin, we define cuts and value as before: A (fractional) cut for $\mathscr{G}_P^{(d)}$ is any measurable function $f : \mathbb{R}^d \to [-1, 1]$, and

$$\text{val}_{\mathscr{G}_P^{(d)}}(f) = \mathop{\mathbf{E}}_{\rho \sim P} \quad \mathop{\mathbf{E}}_{\substack{(\boldsymbol{x}, \boldsymbol{y}) \ \rho\text{-corr'd} \\ d\text{-dim. Gaussians}}} \quad [\tfrac{1}{2} - \tfrac{1}{2} f(\boldsymbol{x}) f(\boldsymbol{y})].$$

Since we allow 'self-loops' (i.e., P's with probability mass on 1), one should note that we *can't* necessarily find 'proper' cuts with value at least that of fractional cuts. We define $\text{Opt}(\mathscr{G}_P^{(d)})$ to be the supremum of the value over all *fractional* cuts.

Second, we define $\text{Sdp}(\mathscr{G}_P^{(d)})$ essentially as in the SDP (4.1):

$$\text{Sdp}(\mathscr{G}_P^{(d)}) = \sup_{g: \mathbb{R}^d \to B_d} \mathop{\mathbf{E}}_{(u,v)} [\tfrac{1}{2} - \tfrac{1}{2} g(u) \cdot g(v)].$$

Some comments on this definition: Again, because of self-loops, it is not necessarily true that the optimal embedding g maps into the surface of the ball S^{d-1}. As it happens, though, we are only concerned with proving lower bounds on $\text{Sdp}(\mathscr{G}_P^{(d)})$, and the embeddings we will use happen to map into S^{d-1} anyway. Second, the most natural definition of $\text{Sdp}(G)$ for an 'infinite graph' G would allow embeddings into B_m and have an additional sup over $m \in \mathbb{N}$. But again, we will end up only considering embeddings $\mathbb{R}^d \to S^{d-1}$ for $\mathscr{G}_P^{(d)}$, so we choose to make the above simpler definition.

Having made these definitions, the goal of this section is to prove the following two theorems:

Theorem 4.3.3. *Let $G = \mathscr{G}_P^{(d)}$ be a d-dimensional Gaussian mixture graph, and let $c = \text{Spread}(P) = \mathbf{E}_{\rho \sim P}[\frac{1}{2} - \frac{1}{2}\rho]$. Then $\text{Sdp}(G) \geq c - O(\sqrt{\log d / d})$, via the embedding $g : \mathbb{R}^d \to S^{d-1}$ mapping x to $x/\|x\|$.*[3]

Theorem 4.3.4. *Let $G = \mathscr{G}_P^{(d)}$ be a d-dimensional Gaussian mixture graph for which P is a $(1, \rho_0)$-distribution. Then the optimal fractional cut for G is achieved by an increasing, odd, 'one-dimensional' cut; i.e., a function $s : \mathbb{R}^d \to [-1, 1]$ of the form $s(x) = r(x_1)$, where $r : \mathbb{R} \to [-1, 1]$ is increasing and odd.*

Theorem 4.3.3 is just a calculation; the heart of the matter is Theorem 4.3.4.

Before proving these theorems, let us see how together they imply $\text{Gap}_{\text{SDP}}(c) \leq S(c)$. Let P be a $(1, \rho_0)$-distribution achieving the inf in the definition of $S(c)$ to within ϵ. Now consider $G = \mathscr{G}_P^{(d)}$. By Theorem 4.3.3, $\text{Sdp}(G) \geq c - O(\sqrt{\log d / d})$. On the other hand, Theorem 4.3.4 implies that

$$\text{Opt}(G) \leq \sup_{\substack{s : \mathbb{R}^d \to [-1,1] \\ \text{one-dimensional, increasing, odd}}} \text{val}_G(s).$$

But when s is one-dimensional, $s(x) = r(x_1)$, it's immediate from the definitions that $\text{val}_G(s) = \text{val}_{\mathscr{G}_P^{(1)}}(r)$. Thus we have $\text{Opt}(G) \leq S(c) + \epsilon$.

Having determined this Gaussian mixture graph G with $\text{Sdp}(G) \geq c - O(\sqrt{\log d / d})$ and $\text{Opt}(G) \leq S(c) + \epsilon$, we are essentially done. Using Proposition 4.3.1 we can convert G to a finite, self-loopless graph G' with $\text{Sdp}(G') \geq c - O(\sqrt{\log d / d})$ and $\text{Opt}(G) \leq S(c) + 2\epsilon$; since $\epsilon > 0$ is arbitrary this proves that $\text{Gap}_{\text{SDP}}(c - O(\sqrt{\log d / d})) \leq S(c)$. Now by the continuity of Gap_{SDP} (Proposition 4.3.2), we conclude that $\text{Gap}_{\text{SDP}}(c) \leq S(c)$.

4.3.2 Proof of Theorem 4.3.3

Theorem 4.3.3 *Let $G = \mathscr{G}_P^{(d)}$ be a d-dimensional Gaussian mixture graph, and let $c = \text{Spread}(P) = \mathbf{E}_{\rho \sim P}[\frac{1}{2} - \frac{1}{2}\rho]$. Then $\text{Sdp}(G) \geq c - O(\sqrt{\log d / d})$, via the embedding $g : \mathbb{R}^d \to S^{d-1}$ mapping x to $x/\|x\|$.*

---

[3]$g(0)$ can be set arbitrarily.

Proof. As stated, let $g(x) = x/\|x\|$, which maps $\mathbb{R}^d$ onto S^{d-1}. (The value of $g(0)$ may be set arbitrarily since the probability that one of $\mathscr{G}_P^{(d)}$'s 'edges' involves 0 is 0.) We need to show:

$$\mathop{\mathbf{E}}_{\rho \sim P} \mathop{\mathbf{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y}) \ \rho\text{-corr'd} \\ d\text{-dim, Gaussians}}} \left[\frac{1}{2} - \frac{1}{2} \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} \cdot \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}\right] \geq \mathop{\mathbf{E}}_{\rho \sim P}[\tfrac{1}{2} - \tfrac{1}{2}\rho] - O(\sqrt{\log d / d}).$$

Clearly it suffices to prove the following:

$$\text{for all } \rho \in [-1, 1], \quad \mathop{\mathbf{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y}) \ \rho\text{-corr'd} \\ d\text{-dim, Gaussians}}} \left[\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} \cdot \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}\right] \leq \rho + O(\sqrt{\log d / d}). \tag{4.6}$$

This can be considered a standard probability result. Inside the expectation, in the numerator, we have

$$\boldsymbol{x} \cdot \boldsymbol{y} = \sum_{i=1}^n x_i y_i,$$

and the summands $x_i y_i$ are i.i.d. real-valued random variables. The expectation of $x_i y_i$ is ρ, and the variance and third absolute moment are bounded by absolute constants. Thus the Berry-Esseen theorem implies that $x \cdot y$ will be in the range $\rho d \pm O(\sqrt{d \log d})$ except with probability at most $O(1/\sqrt{d})$. In the denominator, it is well-known (and a similar argument shows) that $\|\boldsymbol{x}\|$ and $\|\boldsymbol{y}\|$ will each be in the range $\sqrt{d} \pm O(\sqrt{\log d})$ except with probability at most $O(1/\sqrt{d})$. Hence except with probability at most $O(1/\sqrt{d})$ we have that

$$\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} \cdot \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|} \leq \frac{\rho d + O(\sqrt{d \log d})}{(\sqrt{d} - O(\sqrt{\log d}))(\sqrt{d} - O(\sqrt{\log d}))} \leq \rho + O(\sqrt{\log d / d}).$$

Since $\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|} \cdot \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}$ is bounded above by 1 always, we gain at most $O(1/\sqrt{d})$ in the exceptional cases, and conclude that (4.6) indeed holds. $\qquad\square$

4.3.3 Proof of Theorem 4.3.4

Before proceeding with the proof of Theorem 4.3.4 we record here the basic facts from 'Hermite analysis' we will use throughout this work.

The space of functions $L^2(\mathbb{R}^d)$ under the Gaussian distribution has a countable orthonormal basis given by products of normalized Hermite polynomials. These products are indexed by vectors $S \in \mathbb{N}^d$; we use the notation $|S|$ for $\sum_{i=1}^d S_i$, which is also the degree of the product polynomial H_S. We can express any such function f via its 'Hermite expansion',

$$f(x) = \sum_{S \in \mathbb{N}^d} \widehat{f}(S) H_S(x),$$

with convergence in L^2-norm. We make frequent use of the following definition:

Definition 4.3.5. *Given $f \in L^2(\mathbb{R}^d)$ and $\rho \in [-1, 1]$, the* noise stability *of f at ρ is*

$$\mathbb{S}_\rho(f) = \mathop{\mathbf{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y}) \ \rho\text{-corr'd} \\ d\text{-dim. Gaussians}}} [f(\boldsymbol{x}) f(\boldsymbol{y})].$$

70

(Note that we reversed the notational position of ρ and g in section 4.2.3 for clarity of exposition.) The following basic facts about Hermite expansions are well known; it is essentially the Fourier Analysis in Chapter 2 with (Ω, μ) taken to be the Gaussian Distribution (though the dimensionality is infinity now See, e.g., [102] and the references therein.

Proposition 4.3.6.

1. $\mathbb{S}_\rho(f) = \sum_{S \in \mathbb{N}^d} \rho^{|S|} \widehat{f}(S)^2$.
2. $\mathbb{S}_1(f) = \sum_{S \in N^d} \widehat{f}(S)^2 = \mathbf{E}[f^2]$.
3. *If f is an odd function (i.e., $f(-x) = -f(x)$), then $\widehat{f}(S) = 0$ unless $|S|$ is odd.*
4. *If f is an odd function then $\mathbb{S}_{-\rho}(f) = -\mathbb{S}_\rho(f)$.*

We also immediately deduce the following fact:

Proposition 4.3.7. *Assume f is an odd function. Then as a function of ρ, $\mathbb{S}_\rho(f)$ is a power series with nonnegative coefficients, odd powers of ρ only, and radius of convergence at least 1. In particular it is an odd function of ρ, strictly increasing on $[-1,1]$, 0 at 0, concave on $[-1,0]$, and convex on $[0,1]$.*

We now proceed with the proof:

Theorem 4.3.4 *Let $G = \mathscr{G}_P^{(d)}$ be a d-dimensional Gaussian mixture graph for which P is a $(1, \rho_0)$-distribution. Then the optimal fractional cut for G is achieved by an increasing, odd, 'one-dimensional' cut; i.e., a function $s : \mathbb{R}^d \to [-1,1]$ of the form $s(x) = r(x_1)$, where $r : \mathbb{R} \to [-1,1]$ is increasing and odd.*

Proof. Suppose P has weight p on the point $-1 \le \rho_0 \le 0$ and weight $1-p$ on the point 1. Let (f_i) be a sequence of measurable fractional cuts, $f_i : \mathbb{R}^d \to [-1,1]$, for which $\mathrm{val}_G(f_i) \nearrow \mathrm{Opt}(G)$. We have

$$\mathrm{val}_G(f_i) = \mathop{\mathbf{E}}_{\rho \sim P} \mathop{\mathbf{E}}_{\substack{(\boldsymbol{x}, \boldsymbol{y}) \, \rho\text{-corr'd} \\ d\text{-dim. Gaussians}}} [\tfrac{1}{2} - \tfrac{1}{2} f_i(\boldsymbol{x}) f_i(\boldsymbol{y})] = \tfrac{1}{2} - \tfrac{1}{2} \mathop{\mathbf{E}}_{\rho \sim P} [\mathbb{S}_\rho(f_i)],$$

and hence

$$1 - 2\mathrm{val}_G(f_i) = (1-p)\mathbb{S}_1(f_i) + p\mathbb{S}_{\rho_0}(f_i). \tag{4.7}$$

Consider now replacing f_i by f_i^{odd}, the function $\mathbb{R}^d \to [-1,1]$ given by $f_i^{\mathrm{odd}}(x) = (f_i(x) - f_i(-x))/2$. It is well known that $\widehat{f_i^{\mathrm{odd}}}(S)$ equals $\widehat{f_i}(S)$ for odd $|S|$ and is 0 for even $|S|$. Thus when we make this replacement, $\mathbb{S}_1(f_i) = \sum_S \widehat{f_i}(S)^2$ only decreases, and similarly $\mathbb{S}_{\rho_0}(f_i) = \sum_S \widehat{f_i}(S)^2 \rho_0^{|S|}$ only decreases (using the fact that $\rho_0 \le 0$). Thus (4.7) only decreases, and hence $\mathrm{val}_G(f_i)$ can only increase. Thus we may assume each f_i is odd.

Given this assumption and using Proposition 4.3.6.4,

$$1 - 2\mathrm{val}_G(f_i) \quad = \quad (4.7) \quad = \quad (1-p)\mathbf{E}[f_i^2] - p\mathbb{S}_{-\rho_0}(f_i). \tag{4.8}$$

71

We now appeal to the Gaussian rearrangement inequality of Borell [24], which implies that for any function $f_i \in L^2(\mathbb{R}^d)$ and any nonnegative ρ,

$$\mathbb{S}_\rho(f_i) \le \mathbb{S}_\rho(f_i^*);$$

here f_i^* is the Gaussian rearrangement of f_i, an increasing, one-dimensional function.[4] Suppose then we replace each f_i by f_i^*. Since it holds that $\mathbf{E}[(f_i^*)^2] = \mathbf{E}[f_i^2]$, the first term in (4.8) does not change. But $-\rho_0$ is nonnegative, so we can use Borell's result to conclude that the second term $\mathbb{S}_{-\rho_0}(f_i)$ only increases. Hence (4.8) only decreases under Gaussian rearrangement and thus $\mathrm{val}_G(f_i)$ only increases. Thus we may replace all of the f_i's by their Gaussian rearrangements. Note that an odd function, when rearranged, is still odd.

We now have a sequence of one-dimensional, odd, increasing functions $r_i : \mathbb{R} \to [-1, 1]$, with $\mathrm{val}_G(r_i) \nearrow \mathrm{Opt}(G)$ (we abuse notation here slightly instead of writing $\mathrm{val}_G(s_i)$ where $s_i : \mathbb{R}^d \to [-1, 1]$ is defined by $s_i(x) = r(x_1)$). It is well known that using a Helly-type proof we can pass to a subsequence that converges a.e. to an increasing, one-dimensional function r, which must also be odd. Dominated convergence then implies that $\mathrm{val}_G(r) = \mathrm{Opt}(G)$. $\qquad\square$

4.4 $\mathrm{Gap}_{\mathrm{SDP}}(c) \ge S(c)$: Discretized RPR^2 and Minimax

In this section we show that $\mathrm{Gap}_{\mathrm{SDP}}(c) \ge S(c)$. As described in section 4.2.2, the idea will be to randomly find cuts in a given embedded graph by trying the RPR^2 algorithm with 'all' increasing, odd rounding functions. Of course, we actually only try 'all' such functions up to some discretization. Specifically:

Definition 4.4.1. *Given $\epsilon > 0$, let $\mathscr{I}_\epsilon$ denote the partition of $\mathbb{R} \setminus \{0\}$ into intervals,*

$$\mathscr{I}_\epsilon = \{\pm(-\infty, -B], \pm(-B, -B + \epsilon^2], \pm(-B + \epsilon^2, -B + 2\epsilon^2], \ldots, \pm(-2\epsilon^2, \epsilon^2], \pm(-\epsilon^2, \epsilon^2)\},$$

where $B = B(\epsilon)$ is the smallest integer multiple of ϵ^2 exceeding $\sqrt{2\ln(1/\epsilon)}$. We say that a function $r : \mathbb{R} \to [-1, 1]$ is ϵ-discretized if the following hold:

- *r is identically -1 on $(-\infty, -B]$, 0 at 0, and identically 1 on $[B, \infty)$.*
- *r's values on the finite intervals in $\mathscr{I}_\epsilon$ are from the set $\epsilon\mathbb{Z} \cap (-1, 1)$.*

Note that the number of different ϵ-discretized r's is $2^{\tilde{O}(1/\epsilon^2)}$.

The main theorem we prove in this section is the following:

Theorem 4.4.2. *There is a universal constant[5] $K < \infty$ such that for all $c \in [\frac{1}{2}, 1]$,*

$$\inf_{\substack{\text{discrete dists } P \text{ on } [-1,1] \\ \text{with mean } 1-2c}} \quad \max_{\substack{\epsilon\text{-discretized } r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r) \tag{4.9}$$

---

[4]Borell only proves this for f_i Lipschitz and nonnegative, but both conditions are inessential; the first can be removed by standard approximation arguments and the second simply by adding a sufficiently large constant. Alternatively, one can use the alternate proof of Borell's theorem due to Beckner [18].

[5]In future results in this section, different K's may have different values; however they never depend on c or ϵ.

is within $\pm K\epsilon$ of

$$S(c) \quad = \quad \inf_{\substack{(1,\rho_0)\text{-}distributions\ P \\ with\ mean\ 1-2c}} \quad \sup_{\substack{r:\mathbb{R}\to[-1,1] \\ increasing,\ odd}} \quad \mathrm{val}_{\mathscr{G}_P}(r).$$

Aside from discretization issues, the main idea here is using Hermite analysis and the von Neumann Minimax Theorem to show that 'worst' ρ-distribution is a $(1,\rho_0)$-distribution. Incidentally, the discretization issues are not just necessary because we want a finitary algorithm; in fact, discretization is also necessary for the employ of the Minimax Theorem (which also requires a finitary setting, or at least some kind of continuity and compactness).

Let us explain how we can use Theorem 4.4.2 algorithmically:

Theorem 4.4.3. *Let G be any (discrete) embedded graph with spread c. If we run Algorithm 4.2.4 on G, trying RPR^2 on G with all possible increasing, odd ϵ-discretized rounding functions r, then at least one will achieve, in expectation, a cut of value at least $S(c) - O(\epsilon)$. In particular, there exists a cut in G with value at least $S(c)$.*

Proof. Given any r, the observation (4.3) from Section 4.2.2 implies that $\mathrm{Alg}_{RPR^2}(G) = \mathrm{val}_{\mathscr{G}_P}(r)$. Thus the suggested algorithm achieves at least (4.9), which by Theorem 4.4.2 is at least $S(c) - K\epsilon$. As for the last statement in the theorem, we've in particular shown that there exists some cut $f_\epsilon : V \to [-1,1]$ with value at least $S(c) - K\epsilon$. Taking $\epsilon \to 0$ we can get a sequence of cuts f_i with $\limsup \mathrm{val}_G(f_i) \geq S(c)$. But since each cut is just a point in the compact, finite-dimensional cube $[-1,1]^{|V|}$ and since $\mathrm{val}_G(\cdot)$ is continuous, we can extract a limiting cut f with value at least $S(c)$. $\square$

Corollary 4.4.4. *For each $c \in [\frac{1}{2},1]$ it holds that $\mathrm{Gap}_{SDP}(c) \geq S(c)$. Indeed, there is an algorithm which, given any graph G with $\mathrm{Sdp}(G) \geq c$ and any $\epsilon > 0$, runs in time $\mathrm{poly}(|V|) \cdot 2^{\tilde{O}(1/\epsilon^2)}$ and with high probability outputs a proper cut in G with value at least $S(c) - \epsilon$.*

Proof. Given G, we can solve the semidefinite program and find an isomorphic embedded graph G' with spread at least c. It is quite easy to decrease the spread of an embedded graph arbitrarily; for example, map each $x \in S^{n-1}$ to $(tx, \sqrt{1-t^2}) \in S^n$ for a $t \in [0,1]$ of one's choosing. Thus we may assume that G' has spread exactly c. Now the algorithm from Theorem 4.4.3 (which has the dominating running time stated) is used to obtain a cut with value at least $S(c) - O(\epsilon)$. As $\epsilon > 0$ can be arbitrarily small, this establishes $\mathrm{Gap}_{SDP}(c) \geq S(c)$. $\square$

Some minor algorithmic details are discussed more carefully in Section 4.13. One we need to mention explicitly is that our algorithm cannot solve the SDP exactly. Instead, we can use it to find an isomorphic graph with spread exactly $c - \epsilon^2$. Then the algorithm will find a cut with value at least $S(c - \epsilon^2) - O(\epsilon)$. Since we now know $S = \mathrm{Gap}_{SDP}$, we can inspect the proof of Proposition 4.3.2 and conclude that $S(c - \epsilon^2) \geq S(c) - O(\epsilon^2)$ if c is bounded away from 1, and we can use the fact that $\mathrm{Gap}_{SDP}(1 - \delta) = 1 - \arccos(-1 + 2\delta)/\pi = 1 - \Theta(\sqrt{\delta})$ (from Goemans-Williamson) to conclude that $S(c - \epsilon^2) \geq S(c) - O(\epsilon)$ if c is close to 1. $\square$

We discuss the issue of the running time's dependence of ϵ in section 4.6.2.

Combining Corollary 4.4.4 with the results of section 4.3 completes the proof that

$$\text{Gap}_{\text{SDP}}(c) = S(c).$$

The remainder of this section is devoted to proving Theorem 4.4.2. The proof will proceed by transforming (4.9) into $S(c)$ in several steps. Each step will modify the range of either the inf or sup, while changing the overall value by at most $K\epsilon$.

4.4.1 Discretizing Distributions

The first step involves showing we can discretize the *distributions P* appearing in (4.9). This will facilitate our application of the Minimax Theorem.

Definition 4.4.5. *Let $c \in [\frac{1}{2}, 1]$ be given and fixed. We say that a discrete distribution P on $[-1,1]$ is η-discretized if its support is contained in $\eta\mathbb{Z} \cup \{-1,1\}$.*

Lemma 4.4.6. *There is a universal constant $K < \infty$ such that for each $c \in [\frac{1}{2},1]$,*

$$(4.9) \quad = \inf_{\substack{\text{discrete dists } P \text{ on } [-1,1] \\ \text{with mean } 1-2c}} \quad \max_{\substack{\epsilon\text{-discretized } r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \text{val}_{\mathscr{G}_P}(r)$$

is within $\pm K\epsilon$ of

$$\inf_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1-2c}} \quad \max_{\substack{\epsilon\text{-discretized } r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \text{val}_{\mathscr{G}_P}(r). \qquad (4.10)$$

Proof. In fact, (4.10) is clearly at least (4.9), since the inf is over a smaller set. To show the difference is at most $O(\epsilon)$ it suffices to show that every discrete distribution P on $[-1,1]$ with mean $1-2c$ can be converted into an ϵ^7-discretized distribution P' with mean $1-2c$ such that

$$\left| \text{val}_{\mathscr{G}_P}(r) - \text{val}_{\mathscr{G}_{P'}}(r) \right| \le O(\epsilon)$$

$$\Leftrightarrow \quad \left| \underset{\rho \sim P}{\mathbf{E}} \underset{\substack{(x,y) \ \rho\text{-corr'd} \\ \text{Gaussians}}}{\mathbf{E}} [r(x)r(y)] - \underset{\rho \sim P'}{\mathbf{E}} \underset{\substack{(x,y) \ \rho\text{-corr'd} \\ \text{Gaussians}}}{\mathbf{E}} [r(x)r(y)] \right| \le O(\epsilon) \qquad (4.11)$$

holds for for every ϵ-discretized, increasing, odd r.

The conversion of P to P' proceeds as follows. For each atom ρ_i of P, choose $\rho_i' \le \rho_i''$ to be the two values in $\epsilon^7\mathbb{Z} \cup \{-1,1\}$ which straddle ρ_i as closely as possible. Write also $\rho_i = \lambda_i \rho_i' + (1-\lambda_i)\rho_i''$, $\lambda_i \in [0,1]$. We form P' be replacing each atom ρ_i with probability mass p_i in P with the pair of atoms ρ_i', ρ_i'' with masses $p_i\lambda_i$, $p_i(1-\lambda_i)$, respectively. We have that P' is indeed an ϵ^7-discretized distribution with the same mean as P, namely $1-2c$.

Note that $|\rho_i' - \rho_i|, |\rho_i'' - \rho_i| \le \epsilon^7$ always. It's easy now to see that (4.11) will follow if we can show

$$\left| \underset{\substack{(x,y) \ \rho_i'\text{-corr'd} \\ \text{Gaussians}}}{\mathbf{E}} [r(x)r(y)] - \underset{\substack{(x,y) \ \rho_i\text{-corr'd} \\ \text{Gaussians}}}{\mathbf{E}} [r(x)r(y)] \right| \le O(\epsilon) \qquad (4.12)$$

holds for all ϵ-discretized increasing odd r, using only $|\rho_i' - \rho_i| \leq \epsilon^7$. Now the left side of (4.12) is equal to $|\mathbb{S}_{\rho_i'}(r) - \mathbb{S}_{\rho_i}(r)|$, and r here is odd. Thus by the increasing/concavity/convexity properties of $\mathbb{S}_\rho(r)$ given in Proposition 4.3.7, we immediately see that the largest possible of $|\mathbb{S}_{\rho_i'}(r) - \mathbb{S}_{\rho_i}(r)|$ value would occur when $\rho_i' = 1$ and $\rho_i = 1 - \epsilon^7$ (or equivalently, $\rho_i' = -1$, $\rho_i = -1 + \epsilon^7$). Thus the proof of (4.12) and hence the theorem follows Claim 4.4.7 below. $\qquad\square$

Claim 4.4.7. *For every fixed ϵ-discretized, increasing, odd r,*

$$\left| \underset{\substack{(x,y) \text{ 1-corr'd} \\ \text{Gaussians}}}{\mathbf{E}} [r(x)r(y)] \quad - \quad \underset{\substack{(x,y) \text{ } (1-\epsilon^7)\text{-corr'd} \\ \text{Gaussians}}}{\mathbf{E}} [r(x)r(y)] \right| \leq O(\epsilon).$$

Proof. Write $\eta = \epsilon^7$. Since 1-correlated Gaussians are identical, we are comparing

$$\underset{\substack{(x,y) \text{ } (1-\eta)\text{-corr'd} \\ \text{Gaussians}}}{\mathbf{E}} [r(x)r(y)]$$

with $\mathbf{E}[r(x)^2]$. Using the fact that r is ϵ-discretized, it suffices to show that when (x,y) is a pair of $(1-\eta)$-correlated Gaussians, the probability that x and y land in different intervals from $\mathscr{I}_\epsilon$ (recall Definition 4.4.1) is at most $O(\epsilon)$. We will first give up on the half-infinite intervals in $\mathscr{I}_\epsilon$; using the fact that x and y are both individually distributed as Gaussians, the probability that either of them ends up at least $B \geq \sqrt{2\ln(1/\epsilon)}$ in absolute value is at most $O(\epsilon)$ anyway. Also, the probability that either lands on 0 is 0. It remains to consider the intervals of the form $I = [t, t + \epsilon^2)$, where $0 \leq t < B$ (the case of negative intervals will be the same). The probability density function for x is nearly constant over the interval I; in particular, the ratio between its values at t and $t + \epsilon^2$ is $\exp(\epsilon^2 t + \epsilon^4/2)$, which is close to 1 (since $t < B = O(\sqrt{\log(1/\epsilon)})$). Even just using that it is at most 2, we conclude that conditioned on x falling into I, the probability that x falls into $[t + 2\epsilon^3, t + \epsilon^2 - \epsilon^3]$ is at least $1 - O(3\epsilon^3/\epsilon^2) = 1 - O(\epsilon)$.

By losing $O(\epsilon)$ probability, we will assume this happens. In this case, y is distributed as $(1-\eta)x + \sqrt{1 - (1-\eta)^2}N(0,1)$, where $N(0,1)$ is a standard normal. Note that $(1-\eta)x = x - \eta x \geq x - \eta B \geq x - \epsilon^3$, since $\eta x \leq \epsilon^7 B \ll \epsilon^3$. Hence we have $(1-\eta)x \in [t + \epsilon^3, t + \epsilon^2 - \epsilon^3]$. Given this, the conditional probability that y won't also fall into I is at most the probability that $\sqrt{1 - (1-\eta)^2}N(0,1)$ will exceed ϵ^3 in absolute value. But the standard deviation of this normal is $O(\sqrt{\eta}) = O(\epsilon^{3.5})$, so the probability it will exceed ϵ^3 in absolute value is exponentially small in ϵ, certainly smaller than $O(\epsilon)$. Thus we've shown that except with probability at most $O(\epsilon)$, x and y will fall into the same interval from $\mathscr{I}_\epsilon$, and this completes the proof of the claim. $\qquad\square$

4.4.2 Minimax

The next step in the proof of Theorem 4.4.2 is to reinterpret the space of ϵ^7-discretized distributions P with mean $1 - 2c$:

Fact 4.4.8. *Any ϵ^7-discretized distribution P with mean $1 - 2c$ can be expressed as a convex combination of 2-point ϵ^7-discretized distributions each with mean $1 - 2c$ (and vice versa, clearly).*

Here, by a '2-point distribution' we mean one whose support is on at most two points (i.e., either one or two points).

Proof. This fact can be considered standard. One proof sketch is the following: Given any ϵ^7-discretized P with mean $1-2c$, pick any two points which straddle $1-2c$ and on which P has positive probability mass (the two points may coincide in case P has mass on $1-2c$). Such a pair must exist because P has mean $1-2c$. Take the mean-$(1-2c)$ probability distribution over this pair and 'remove it from P' (i.e., subtract and rescale) to the greatest extent possible. This will preserve the mean of P being $1-2c$, and it will also cause P to have support on (at least) one fewer point. Repeat this process until P is empty; the pairs extracted give the required combination of 2-point distributions. $\square$

The next step is to reverse the inf/min and max in (4.10) using the von Neumann Minimax theorem.

Lemma 4.4.9.

$$(4.10) \quad = \quad \min_{\substack{\epsilon^7\text{-}discretized\ dists\ P \\ with\ mean\ 1-2c}} \quad \max_{\substack{\epsilon\text{-}discretized\ r:\mathbb{R}\to[-1,1] \\ increasing,\ odd}} \quad \mathrm{val}_{\mathscr{G}_P}(r) \tag{4.13}$$

$$= \quad \max_{\substack{probability\ distributions\ R\ over \\ \epsilon\text{-}discretized,\ increasing \\ odd\ r:\mathbb{R}\to[-1,1]}} \quad \min_{\substack{2\text{-}point\ \epsilon^7\text{-}discretized\ dists\ P \\ with\ mean\ 1-2c}} \quad \mathop{\mathbf{E}}_{r\sim R} [\mathrm{val}_{\mathscr{G}_P}(r)]. \tag{4.14}$$

Proof. Note that (4.10), which has an inf, is not precisely the same as (4.13), which has a min. We will show that (4.10) equals (4.14) using the Minimax theorem. Since a corollary of the Minimax theorem is that the inf's and sup's involved are achieved, this will imply that (4.10) is equal to (4.13) and that we can write min and max everywhere.

Consider a zero-sum game between a 'Distribution Player' and a 'Function Player'. Acting simultaneously, the Distribution Player chooses a 2-point ϵ^7-discretized probability distribution P with mean $1-2c$, and the Function Player chooses an increasing, odd, ϵ-discretized $r : \mathbb{R} \to [-1,1]$. The payoff is $\mathrm{val}_{\mathscr{G}_P}(r)$ to the Function Player from the Distribution Player.

Note that both players choose from a finite set of strategies; for the Distribution Player, this uses the fact that for any pair of discretized points, there is at most one distribution with mean $1-2c$ supported on this pair. Therefore we may apply the von Neumann Minimax theorem. We conclude that the game has some value, which is achieved in both of the following scenarios: (a) the Function Player goes first and gets to choose a mixed strategy, and then the Distribution Player goes second and gets to choose a pure strategy; and, (b) the Distribution Player goes first and gets to choose a mixed strategy, and the Function Player goes second and gets to choose a pure strategy. The value in (a) is clearly (4.14). As for the value in (b), we claim it equals (4.13). This follows from Fact 4.4.8, along with the fact that if we identify a P with a convex combination of 2-point distributions Q, then for

any r,

$$\mathop{\mathbf{E}}_{Q\sim P}[\mathrm{val}_{\mathscr{G}_Q}(r)] \quad = \quad \mathop{\mathbf{E}}_{Q\sim P}\ \mathop{\mathbf{E}}_{\rho\sim Q}\ \mathop{\mathbf{E}}_{\substack{(x,y)\ \rho\text{-corr'd}\\ \text{Gaussians}}}\ [\tfrac{1}{2} - \tfrac{1}{2}r(x)r(y)]$$

$$= \quad \mathop{\mathbf{E}}_{\rho\sim P}\ \mathop{\mathbf{E}}_{\substack{(x,y)\ \rho\text{-corr'd}\\ \text{Gaussians}}}\ [\tfrac{1}{2} - \tfrac{1}{2}r(x)r(y)] \quad = \quad \mathrm{val}_{\mathscr{G}_P}(r).$$

Hence (4.13) equals (4.14) and the proof is complete. □

4.4.3   More Minimax; Convexity and Concavity

In the next step, we use the special properties of $\mathbb{S}_\rho(r)$ for odd r given in Proposition 4.3.7, along with further Minimax-based reasoning, to deduce that the 'Distribution Player' essentially may as well use a $(1, \rho_0)$-distribution. This idea was discussed in section 4.2.3.

Definition 4.4.10. *We say an ϵ^7-discretized distribution P is* almost-$(1, \rho_0)$ *if it is the mixture of two $(1, \rho_0)$-distributions for which the two ρ_0 values are neighboring (or equal) discretized values.*

Lemma 4.4.11.

$$(4.13) = \min_{\substack{\epsilon^7\text{-discretized dists }P\\ \text{with mean }1-2c}}\ \max_{\substack{\epsilon\text{-discretized }r:\mathbb{R}\to[-1,1]\\ \text{increasing, odd}}}\ \mathrm{val}_{\mathscr{G}_P}(r)$$

$$= \min_{\substack{\epsilon^7\text{-discretized almost-}(1,\rho_0)\text{-dists }P\\ \text{with mean }1-2c}}\ \max_{\substack{\epsilon\text{-discretized }r:\mathbb{R}\to[-1,1]\\ \text{increasing, odd}}}\ \mathrm{val}_{\mathscr{G}_P}(r). \quad (4.15)$$

Proof. Let P^* denote an ϵ^7-discretized distribution with mean $1-2c$ achieving the min in (4.13); i.e., an optimal mixed strategy for the Distribution Player. Let R^* denote a distribution over ϵ-discretized, increasing, odd r achieving the max in (4.14); i.e., an optimal mixed strategy for the Function Player. The Minimax Theorem further implies that P^* is an optimal strategy for the Distribution Player given that the Function Player uses R^*. I.e., P^* is a minimizing choice for P in the following:

$$\min_{\substack{\epsilon^7\text{-discretized dists }P\\ \text{with mean }1-2c}}\ \mathop{\mathbf{E}}_{r\sim R^*}[\mathrm{val}_{\mathscr{G}_P}(r)].$$

Now

$$\mathop{\mathbf{E}}_{r\sim R^*}[\mathrm{val}_{\mathscr{G}_P}(r)] \quad = \quad \mathop{\mathbf{E}}_{r\sim R^*}\ \mathop{\mathbf{E}}_{\rho\sim P}\ \mathop{\mathbf{E}}_{\substack{(x,y)\ \rho\text{-corr'd}\\ \text{Gaussians}}}\ [\tfrac{1}{2} - \tfrac{1}{2}r(x)r(y)] \quad = \quad \tfrac{1}{2} - \tfrac{1}{2}\mathop{\mathbf{E}}_{\rho\sim P}\mathop{\mathbf{E}}_{r\sim R^*}[\mathbb{S}_\rho(r)],$$

and so it follows that P^* is a maximizing choice for P in the following:

$$\max_{\substack{\epsilon^7\text{-discretized dists }P\\ \text{with mean }1-2c}}\ \mathop{\mathbf{E}}_{\rho\sim P}\mathop{\mathbf{E}}_{r\sim R^*}[\mathbb{S}_\rho(r)].$$

Suppose we fix a particular odd r. We now have the special properties of $\mathbb{S}_\rho(r)$ as a function of ρ given in Proposition 4.3.7. We also claim that the convexity and concavity

of this function are essentially *strict*; i.e., $\mathbb{S}_\rho(r)$ is not linear on any open interval. For otherwise, by analyticity, $\frac{d^2}{d\rho^2}\mathbb{S}_\rho(r)$ would have to be 0 everywhere on $[-1,1]$, implying that r is equal (in the L^2 sense) to a linear function. But an ϵ-discretized function cannot be linear, since it is constantly -1 on $(-\infty, -B]$ and constantly 1 on $[B, \infty)$.

Next, note that all of the properties mentioned in Proposition 4.3.7 are maintained under finite convex combinations, in particular because first and second derivatives are linear. Hence if we define

$$q(\rho) = \mathop{\mathbf{E}}_{r \sim R^*}[\mathbb{S}_\rho(r)],$$

we conclude that $q(\rho)$ is also an odd function of ρ, strictly increasing on $[-1,1]$, 0 at 0, concave on $[-1,0]$, convex on $[0,1]$, and not linear on any open interval. An illustration of what q may look like is given in Figure 1.
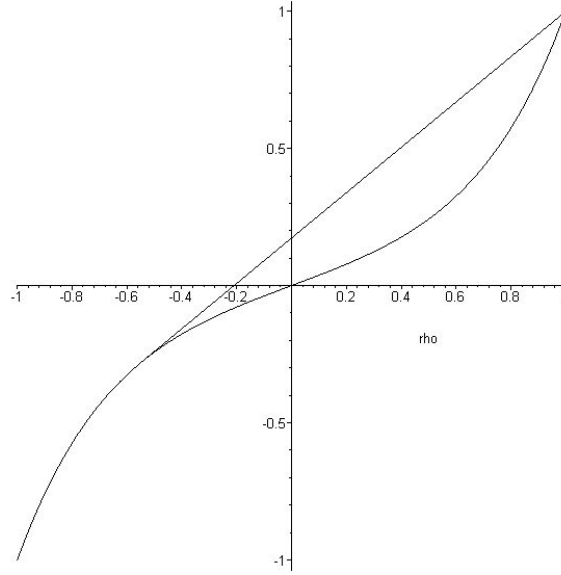
Figure 4.1: Illustrative $q(\rho)$, with least concave upper bound $\overline{q}(\rho)$.

Recall now that P^* is a maximizing choice for P in

$$\max_{\substack{\epsilon^7\text{-discretized dists } P \\ \text{with mean } 1-2c}} \mathop{\mathbf{E}}_{\rho \sim P}[q(\rho)].$$

To complete the proof, we will show that this forces P^* to be almost-$(1, \rho_0)$. Suppose we first disregard the constraint of being ϵ^7-discretized. Then it is easy to see that the maximum value in the above is equal to $\overline{q}(1-2c)$, where $\overline{q}$ denotes the least concave upper bound of the function q. We have that $\overline{q}$ equals q on some interval $[-1, \rho_0]$, where $\rho_0 < 0$, and is a straight line joining $q(\rho_0)$ and $q(1)$ on $[\rho_0, 1]$. Further, in this case there would be a unique maximizing P^*: either the 1-point distribution concentrated on $1-2c$, if $1-2c \leq \rho_0$,

or the $(1, \rho_0)$-distribution with mean $1 - 2c$, if $1 - 2c \geq \rho_0$.

Now we reintroduce the constraint that P^* must be ϵ^7-discretized. Let $\widetilde{q}$ denote the piecewise linear function which interpolates q's values on the discretized points $\epsilon^7 \mathbb{Z}$. We now have that the maximum value of $\mathbf{E}_{\rho \sim P}[q(\rho)]$ is equal to $\overline{\widetilde{q}}(1 - 2c)$, where again $\overline{\widetilde{q}}$ is the least concave upper bound of $\widetilde{q}$. The function $\widetilde{q}$ is still odd, strictly increasing, concave on $[-1, 0]$, and convex on $[0, 1]$; hence again the function $\overline{\widetilde{q}}$ equals $\widetilde{q}$ on some interval $[-1, \rho_0]$, where $\rho_0 < 0$, and is a straight line joining $q(\rho_0)$ and $q(1)$ on $[\rho_0, 1]$. The only difference now is that the point ρ_0 is not necessarily unique; there may be two consecutive possibilities, if the 'secant' at one of the possible ρ_0's is parallel to one of the line segments touching $q(\rho_0)$. (Note that there cannot be more than two possible ρ_0's, since otherwise the graph of q would have three distinct collinear points on $[-1, 0]$ and would thus be linear on some open interval.) We conclude that any maximizing P^* must have all of its support among 1 and the (at most) two discretized values that straddle ρ_0; i.e., P^* must be almost-$(1, \rho_0)$. $\qquad \square$

Finally, we can convert almost-$(1, \rho_0)$-distributions to $(1, \rho_0)$-distributions:

Lemma 4.4.12. *There is a universal constant $K < \infty$ such that for each $c \in [\frac{1}{2}, 1]$,*

$$(4.15) \quad = \quad \min_{\substack{\epsilon^7\text{-discretized almost-}(1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \quad \max_{\substack{\epsilon\text{-discretized } r : \mathbb{R} \to [-1, 1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r)$$

is within $\pm K\epsilon$ of

$$\min_{\substack{\epsilon^7\text{-discretized } (1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \quad \max_{\substack{\epsilon\text{-discretized } r : \mathbb{R} \to [-1, 1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r). \qquad (4.16)$$

Proof. We sketch the proof, which uses the same ideas used in the proof of Lemma 4.4.6. We need to show that any almost-$(1, \rho_0)$-distribution P with mean $1 - 2c$ can be converted into a $(1, \rho_0)$-distribution P' with mean $1 - 2c$ in a such a way that $\mathrm{val}(r)$ changes by at most $O(\epsilon)$ for every ϵ-discretized, increasing, odd r. If P is already a $(1, \rho_0)$-distribution then we are done. Otherwise, it has support on two neighboring discretized values, say $\rho_0' < \rho_0''$. Since the mean of P is $1 - 2c$ we must have $\rho_0' < 1 - 2c$. We now form P' by pushing the weight λ that P gave to ρ_0'' onto ρ_0'. This changes the mean by $\lambda(\rho_0'' - \rho_0') \leq \epsilon^7$, but we can compensate for this by shifting a small amount of weight (at most $2\epsilon^7$) onto the support point 1. One bounds the change in $\mathrm{val}(r)$ caused by these shifts by $O(\epsilon) + O(\epsilon^7)$ via $|\rho_0' - \rho_0''| \leq \epsilon^7$ and Claim 4.4.7. $\qquad \square$

4.4.4 Undiscretizing

We have now reached (4.16), which is very close to $S(c)$; the only difference is that we have discretized distributions and functions. We now 'undiscretize':

Lemma 4.4.13. *There is a universal constant $K < \infty$ such that for each $c \in [\frac{1}{2}, 1]$,*

$$(4.16) \quad = \quad \min_{\substack{\epsilon^7\text{-discretized } (1, \rho_0)\text{-dists } P \\ \text{with mean } 1 - 2c}} \quad \max_{\substack{\epsilon\text{-discretized } r : \mathbb{R} \to [-1, 1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r).$$

79

is within $\pm K\epsilon$ of

$$\inf_{\substack{(1,\rho_0)\text{-}distributions\ P \\ with\ mean\ 1-2c}} \quad \sup_{\substack{r:\mathbb{R}\to[-1,1] \\ increasing,\ odd}} \quad \mathrm{val}_{\mathcal{G}_P}(r) \quad = \quad S(c). \tag{4.17}$$

Proof. It is straightforward to see that the ideas from Lemma 4.4.6 can be used to replace the min in (4.16) with the inf from (4.17), changing the value of (4.16) by at most $O(\epsilon)$. Thus we concentrate on discretizing the functions. To that end, fix any $(1,\rho_0)$-distribution P (in fact, our argument will hold for *any* distribution on $[-1,1]$). We will show that for any increasing, odd $r:\mathbb{R}\to[-1,1]$, there is an ϵ-discretized, increasing, odd $r':\mathbb{R}\to[-1,1]$ with $|\mathrm{val}_{\mathcal{G}_P}(r)-\mathrm{val}_{\mathcal{G}_P}(r')|\le O(\epsilon)$. This will complete the proof.

So let r be given. Define the increasing, odd, ϵ-discretized function $r':\mathbb{R}\to[-1,1]$ as follows: On each finite interval I in $\mathscr{I}_\epsilon$, we will take r' to be identically equal to the value of r on the midpoint of I,[6] rounded to the nearest integer multiple of ϵ (or ± 1, if one of these is closer). As necessary, we will also take r' to be identically -1 on $(-\infty,-B]$ and identically 1 on $[B,\infty)$. We now argue that $\mathrm{val}_{\mathcal{G}_P}(r')$ is within $\pm O(\epsilon)$ of $\mathrm{val}_{\mathcal{G}_P}(r)$.

The idea is that $|r-r'|\le\epsilon$ except on a set of small Gaussian measure. We will give up on the two half-infinite intervals and include them in the exceptional set. As for the finite intervals in $\mathscr{I}_\epsilon$, since r is increasing and bounded in $[-1,1]$, for at most $1/\epsilon$ of these intervals can r increase by more than ϵ. On the intervals where it increases by less than ϵ, we indeed have $|r-r'|\le\epsilon$. Hence $|r-r'|$ fails on at most $1/\epsilon$ intervals of width ϵ^2, plus perhaps the two half-infinite intervals $\pm(-\infty,B]$. Note that the total Gaussian measure of these intervals is at most $O(\epsilon)$. It is thus easy to see that

$$\mathrm{val}_{\mathcal{G}_P}(r) = \mathop{\mathbf{E}}_{\rho\sim P} \mathop{\mathbf{E}}_{\substack{(x,y)\ \rho\text{-corr'd} \\ \text{Gaussians}}} [\tfrac{1}{2}-\tfrac{1}{2}r(x)r(y)]$$

is within $\pm O(\epsilon)$ of $\mathrm{val}_{\mathcal{G}_P}(r')$: The probability that either x or y falls into the 'bad' intervals is at most $2\cdot O(\epsilon)$, since x and y are each individually distributed as standard Gaussians. In this case, the difference in values is at most 1. Otherwise, we have that $|r(x)-r'(x)|,|r(y)-r'(y)|\le\epsilon$, and then the difference in values is at most $O(\epsilon)$. $\qquad\square$

Combining all of the Lemmas 4.4.6, 4.4.9, 4.4.11, 4.4.12, 4.4.13, we have proved Theorem 4.4.2.

We end with the following observation:

Corollary 4.4.14. *Each* sup *in the definition of $S(c)$, as well as the* inf*, is achieved. Hence*

$$S(c) \quad = \quad \min_{\substack{(1,\rho_0)\text{-}distributions\ P \\ with\ mean\ 1-2c}} \quad \max_{\substack{r:\mathbb{R}\to[-1,1] \\ increasing,\ odd}} \quad \mathrm{val}_{\mathcal{G}_P}(r).$$

---

[6]Since we are working in $L^2(\mathbb{R})$, technically here we mean the value of any increasing representative of r's equivalence class.

Proof. (Sketch.) The fact that the sup is achieved for each P is proved in Theorem 4.3.4. The fact that the inf is achieved can be deduced by taking a converging subsequence of ρ_0's, and using the discretization Lemmas 4.4.6 and 4.4.13 to show that the max's for close values of ρ_0 are close. $\qquad\square$

4.5 Estimating $S(c)$ Efficiently

This section is devoted to the proof of Theorem 4.1.12:

Theorem 4.1.12 *There is an algorithm that, on input $c \in [\frac{1}{2}, 1]$ and $\epsilon > 0$, runs in time* $\mathrm{poly}(1/\epsilon)$ *and computes $S(c)$ to within $\pm\epsilon$.*

As Lemma 4.4.13 shows, $S(c)$ is within $\pm O(\epsilon)$ of

$$(4.16) \quad = \min_{\substack{\epsilon^7\text{-discretized }(1,\rho_0)\text{-dists P} \\ \text{with mean } 1-2c}} \quad \max_{\substack{\epsilon\text{-discretized }r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r).$$

Since we can enumerate all $\mathrm{poly}(1/\epsilon)$ many ϵ^7-discretized $(1, \rho_0)$-distributions, it is clearly sufficient to show we can efficiently estimate

$$\max_{\substack{\epsilon\text{-discretized }r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r) \tag{4.18}$$

for any $(1, \rho_0)$-distribution P. In fact, for technical reasons, we will show how to estimate a slightly different quantity. Specifically, instead of using the rounding function discretization described in Definition 4.4.1, we will use a different one:

Definition 4.5.1. *Let $\epsilon > 0$ be such that $1/\epsilon^2$ is an odd integer. We define $\mathscr{J}_\epsilon$ to be the partition of $\mathbb{R}$ into $1/\epsilon^2$ intervals of equal Gaussian measure ϵ^2.[7] We say that a function $r : \mathbb{R} \to [-1, 1]$ is ϵ^2-equidiscretized if r is constant on each of the intervals in $\mathscr{J}_\epsilon$.*

We will show how to estimate

$$\sup_{\substack{\epsilon^2\text{-equidiscretized }r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r) \tag{4.19}$$

to within $\pm O(\epsilon)$ in time $\mathrm{poly}(1/\epsilon)$, whenever P is a $(1, \rho_0)$-distribution. Although this quantity is not directly comparable to (4.18), nevertheless with only minor modifications to the proof of Lemma 4.4.13 one can show that $S(c)$ is also within $\pm O(\epsilon)$ of

$$\min_{\substack{\epsilon^7\text{-discretized }(1,\rho_0)\text{-dists P} \\ \text{with mean } 1-2c}} \quad \sup_{\substack{\epsilon^2\text{-equidiscretized }r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \quad \mathrm{val}_{\mathscr{G}_P}(r).$$

(To see this, first note that the function discretization step hardly changes. Second, the proof of Lemma 4.4.6 goes through with ϵ^2-equidiscretized functions as well because the

---

[7]Which partition points are included in which intervals is immaterial.

intervals in $\mathcal{J}_\epsilon$ are only wider than the intervals in $\mathcal{I}_\epsilon$.) Thus efficient estimation of (4.19) for $(1, \rho_0)$-distributions is sufficient to establish Theorem 4.1.12.

The reason for our redefinition of discretization is the following: it allows us to drop the conditions 'increasing, odd' from the optimization problem (4.19). Specifically:

Proposition 4.5.2. *Let P be a $(1, \rho_0)$-distribution and consider the following optimization problem:*

$$\sup_{\epsilon^2\text{-equidiscretized } r:\mathbb{R}\to[-1,1]} \text{val}_{\mathcal{G}_P}(r). \tag{4.20}$$

There exists an optimal solution r^* achieving the sup *which is both increasing and odd.*

Proof. The proof is essentially identical to that of Theorem 4.3.4; the key point is that performing Gaussian rearrangement on an ϵ^2-equidiscretized function yields another ϵ^2-equidiscretized function. □

We now consider (4.20). Suppose P has weight $1 - p$ on the point 1 and weight p on the point ρ_0; of course, $p = 2c/(1 - \rho_0)$. Let us index the intervals in $\mathcal{J}_\epsilon$ from left to right as $I_{-m}, \ldots, I_m$, where $m = (1/\epsilon^2 - 1)/2$. We identify an ϵ^2-equidiscretized function r with the length-$(2m+1)$ vector giving its value on each interval; we will write r_j for the entry corresponding to I_j, $-m \leq j \leq m$. Finally, we write W_ρ for the $(2m+1) \times (2m+1)$ matrix whose (j, k) entry equals the probability that a ρ-correlated pair of Gaussians (x, y) will satisfy $x \in I_j, y \in I_k$. Now

$$\text{val}_{\mathcal{G}_P}(r) = \tfrac{1}{2} - \tfrac{1}{2}\left((1-p) \sum_{-m \leq j,k \leq m} W_1(j,k) r_j r_k + p \sum_{-m \leq j,k \leq m} W_{\rho_0}(j,k) r_j r_k \right),$$

and hence the optimization problem (4.20) is equivalent to the problem

$$\text{minimize} \quad r^\top((1-p)W_1 + pW_{\rho_0})r,$$

$$\text{subject to } -1 \leq r_j \leq 1 \quad \text{for all } -m \leq j \leq m.$$

We now consider the Karush-Kuhn-Tucker conditions for this quadratic program and conclude that any optimal solution r must satisfy

$$\sum_{-m \leq k \leq m} ((1-p)W_1(j,k) + pW_{\rho_0}(j,k))r_k = 0, \quad \text{for all } j \text{ such that } -1 < r_j < 1. \tag{4.21}$$

These necessary conditions for the optimality of a rounding function was already determined by Feige and Langberg [48].

The key observation that lets us make efficient use of the conditions is that we know from Proposition 4.5.2 that there is an optimal *increasing* odd r^*. In particular, there is some $0 \leq m_0 \leq m$ such that

$$\begin{aligned} r_j^* &= -1, \quad \text{for all } j < -m_0, \\ r_j^* &= 1, \quad \text{for all } j > m_0, \\ -1 < r_j^* &< 1, \quad \text{for all } -m_0 \leq j \leq m_0. \end{aligned} \tag{4.22}$$

Thus algorithmically, we can try all possible values for m_0, incurring only an $O(1/\epsilon^2)$ factor slowdown. For each choice, we assume an r^* satisfying the conditions (4.22), and we solve (4.21) for the remaining unknown values; i.e., we solve the square system

$$\sum_{-m_0 \le k \le m_0} ((1-p)W_1(j,k) + pW_{\rho_0}(j,k))r_k = b_j \quad \text{for all } -m_0 \le j \le m_0, \qquad (4.23)$$

where $b_j = \sum_{k<-m_0}((1-p)W_1(j,k) + pW_{\rho_0}(j,k)) - \sum_{k>m_0}((1-p)W_1(j,k) + pW_{\rho_0}(j,k))$. We are guaranteed that there exists an optimal, feasible solution r^* satisfying (4.23) for at least one value of m_0.

4.5.1 Evading Singularity

The above discussion suggests a poly($1/\epsilon$) time algorithm for computing (4.19) exactly. There are two problems we need to circumvent, however. The first problem is that, algorithmically, we cannot compute the values $W_\rho(j,k)$ — or even the endpoints of the intervals in $\mathcal{J}_\epsilon$ — exactly. The more challenging problem is that the square system (4.23) may be singular, in which case it may produce infinitely solutions that would need to be tried. As we will see, once we take care of the latter problem, the former will follow.

Let us write the square system (4.23) more compactly as

$$((1-p)M_{1,m_0} + pM_{\rho_0,m_0})s = b, \qquad (4.24)$$

where M_{ρ,m_0} represents the square submatrix of W_ρ corresponding to indices $-m_0 \ldots m_0$, and s represents the truncation of the vector r to these indices. We may assume here that $m_0 \ge 1$, since there is nothing to solve for if $m_0 = 0$ (note that r_0^* must be 0 by oddness). Write $M_{\rho_0,m_0,p} = (1-p)M_{1,m_0} + pM_{\rho_0,m_0}$.

We are concerned about the possibility that $\det(M_{\rho_0,m_0,p}) = 0$. More generally, we are concerned if the condition number $\kappa(M_{\rho_0,m_0,p})$ is very large, since in this case our inability to calculate the M_{ρ,m_0} matrices precisely would lead to very inaccurate solutions to (4.24). Since the matrix $M_{\rho_0,m_0,p}$ is symmetric, its condition number is

$$\kappa(M_{\rho_0,m_0,p}) = |\lambda_{\max}(M_{\rho_0,m_0,p})|/|\lambda_{\min}(M_{\rho_0,m_0,p})|,$$

where $\lambda_{\max}$ and $\lambda_{\min}$ denote largest and smallest eigenvalues in absolute value. Since each M_{ρ,m_0} is a submatrix of the stochastic matrix W_ρ, its maximum eigenvalue is at most 1; hence we need only worry about the smallest eigenvalue of $M_{\rho_0,m_0,p}$. Since M_{1,m_0} is a multiple of the identity matrix, it can be simultaneously diagonalized with M_{ρ_0,m_0}, and hence the eigenvalues of $M_{\rho_0,m_0,p}$ are precisely

$$\{(1-p) + p\lambda_{\rho_0,m_0}(j)\}_{-m_0 \le j \le m_0},$$

where the $\lambda_{\rho_0,m_0}(j)$'s are the eigenvalues of M_{ρ_0,m_0}. It is easy to see that for any particular $\lambda_{\rho_0,m_0}(j)$, the set of p's for which $(1-p) + p\lambda_{\rho_0,m_0}(j)$ is in the range $(-\delta,\delta)$ is an interval of width at most 2δ. Hence we deduce the following:

Proposition 4.5.3. *For each ρ_0, the set*

$$B_{\rho_0} := \bigcup_{1 \leq m_0 \leq m} \{p : \kappa(M_{\rho_0, m_0, p}) > 1/\delta\}$$

is a collection of at most $m \cdot (2m + 1) = O(1/\epsilon^4)$ intervals of width at most 2δ each.

Our trick now will be to give up on these 'bad' p's; or rather, the 'bad' c-values with which they are associated. Recalling the relationship $p = 2c/(1 - \rho_0) \Leftrightarrow c = (1 - \rho_0)p/2$, we have that

$$C := \bigcup_{\epsilon^7\text{-discretized}\rho_0} \{(1 - \rho_0)p/2 : p \in B_{\rho_0}\}$$

is a collection of at most $O(1/\epsilon^{11})$ intervals of width at most 2δ each. And, whenever $c \notin C$, we are assured that the square system (4.24) has a matrix with condition number at most $1/\delta$.

We now set $\delta = \epsilon^{15}$ and use the following algorithm for estimating $S(c)$. Given c, we try to estimate $S(c')$ for all values $c' = c + t\epsilon^{14}$, for t an integer with $|t| \leq 1/\epsilon^{12}$. If we manage to succeed for some c', then the resulting estimate for $S(c')$ will also be a $\pm O(\epsilon)$ estimate for $S(c)$, since $|c' - c| \leq \epsilon^2$ (and see the proof of Corollary 4.4.4 regarding the continuity of S). There are at most $O(1/\epsilon^{11})$ 'bad' intervals comprising C, and each has width at most 2δ. Since $2\delta \ll \epsilon^{14}$, each such interval contains at most one possible c'; but, there are $2/\epsilon^{12} + 1 \gg O(1/\epsilon^{11})$ possible c', and hence at least one choice must fall outside C. Hence we will succeed for at least one c'.

4.6 On $S(c)$ and Running Times

4.6.1 On $S(c)$

As we have shown, $S(c)$ can be computed to within $\pm\epsilon$ in time poly$(1/\epsilon)$; we believe this result justifies our claim that $S(c)$ is 'explicit'. A reasonable way to understand the notion of 'explicitness' would be with respect to the 'bit model' of Braverman and Cook [25]; in that setting, our poly$(1/\epsilon)$ time algorithm would correspond to a fairly liberal notion of 'explicit', with a polylog$(1/\epsilon)$ time algorithm corresponding to a fairly demanding notion of 'explicit'. The latter notion is the level of explicitness one has for, e.g., '$\frac{1}{\pi}\arccos(1 - 2c)$'. On the other hand, some less explicit-looking bounds have been given for related problems; for example, Haagerup's bound [72] for the complex Grothendieck constant is $8/\pi(k_0 + 1)$, where k_0 is the unique solution of the equation

$$\frac{\pi(k + 1)}{8k} = \int_0^{\pi/2} \frac{\cos^2 t}{\sqrt{1 - k^2 \sin^2 t}} \, dt$$

in the interval $[0, 1]$. This value can surely be computed to within $\pm\epsilon$ in time poly$(1/\epsilon)$; it may well also be computable in time polylog$(1/\epsilon)$ but this is, at least, not immediately obvious.

We in fact used the algorithm behind Theorem 4.1.12 to approximate $S(c)$ for the values .505, .510, .515, ..., .840 (with the values $S(.5) = .5$ and $S(c) = \arccos(1 - 2c)/\pi$ for $c \geq$ .844 being already known). The values we found are given in the table in Section 4.15. We were not completely formal about the approximation process and thus the results in Section 4.15 should not be considered rigorous. In particular, the approximations of the matrices W_ρ were done numerically in Matlab; also, the problem of singularity discussed in section 4.5.1 did not seem to arise and so we disregarded it. We can also report that the best rounding functions r arising in the algorithm were very close to being s-linear, in all cases; they became only slightly rounded near $\pm s$ (convex near $-s$, concave near s).

4.6.2 On the Running Time of the Rounding Algorithm

As shown in Corollary 4.4.4, our MAX CUT rounding algorithm is efficient (polynomial) in terms of its dependence on n, the number of vertices; indeed, the running time is dominated by the time for SDP. To get a cut that is provably within ϵ of $S(\mathrm{Opt}(G))$, however, our algorithm's dependence on ϵ is exponential, $2^{\tilde{O}(1/\epsilon^2)}$. As we will discuss in Section 4.13, all known RPR^2 algorithms have at least some ϵ dependence as well. This dependence is at least poly$(1/\epsilon)$, from converting expectation results to high probability results; in some papers, it is exponential (as in the derandomized Goemans-Williamson algorithm from [46]).

In practice, we feel this issue is not very important. As mentioned in the previous section, we observed that using RPR^2 with s-linear rounding functions (as Feige and Langberg suggested) seems nearly optimal. In particular, it seems to achieve cuts that are within about 10^{-4} of $S(c)$, across all values of c. Further, one can precompute a table of which value of 's' to use for 'each' possible value of c (suitably discretized) — and the algorithm knows what c is after solving the SDP. Thus in practice one can achieve within 10^{-4} of $S(c)$ with *no* real running time overhead. If error smaller than 10^{-4} is desired, it seems one can perform a local search for a better rounding function, starting from the appropriate s-linear function and modifying it slightly near $\pm s$.

Finally, given our poly$(1/\epsilon)$ time algorithm for approximating $S(c)$ to within $\pm \epsilon$, we believe that our rounding algorithm should also be able to have this improved dependence. Since this is not the main focus of our work, we will only briefly describe the technicalities that would need to be overcome. Given an embedded graph G with ρ-distribution P, the idea would *not* be to try to solve the Karush-Kuhn-Tucker conditions for $\mathscr{G}_P$ — since in general we have no promise that the optimal rounding function for $\mathscr{G}_P$ is increasing, we wouldn't be able to effectively try all possibilities for where it is ± 1. Instead, one might simply try to use all of the rounding functions constructed in the determination of $S(c)$. This seems as though it should work: the proof of Theorem 4.4.2 using the Minimax Theorem seems to imply that a convex combination of the optimal rounding functions for $(1, \rho_0)$-distributions will achieve at least $S(c)$ for $\mathscr{G}_P$.

Unfortunately, several technical problems crop up. First, the Minimax proof only implies that 'nearly' $(1, \rho_0)$-distributions are the worst case, and it is unclear if we can effectively enumerate these, since the weight to distribute to the three points is not com-

pletely determined by c. Second, even if we circumvent this problem, the Minimax theorem only implies that some convex combination of *all* the optimal rounding functions for $(1, \rho_0)$-distribution will be good for $\mathcal{G}_P$; however, our algorithm for computing $S(c)$ only finds the *increasing* ones. This problem too might be circumventable if one could prove *strict* increase in Borell's rearrangement inequality assuming the function is not already monotone. Such an 'equality condition' result is probably true, but is currently unknown. Finally, even if both of these issues were fixed, we still have the problem that the Karush-Kuhn-Tucker conditions might be a singular system and thus have multiple (and possibly very many) solutions, all of which theoretically might need to be combined by the 'Function Player'.

4.7 Ddictator-vs.-quasirandom Tests

In this section we discuss Dictator Tests and give the definitions necessary for our 'dictator-vs.-quasirandom' tests. The subsequent two sections are devoted to the proof that $\mathrm{Gap}_{\mathrm{Test}}(c) = S(c)$.

We begin with an essential observation: 2-query Dictator Tests are nothing more than embedded graphs (see Definition 4.2.1), with the vertex set being further restricted to lie within the discrete cube. To make the connection clearer, we treat the discrete cube as lying on the unit sphere:

Definition 4.7.1. *We write* $\{-1, 1\}^n = \{-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}\}^n$ *for the discrete cube, since it is convenient to have* $\{-1, 1\}^n \subseteq S^{n-1}$.

Definition 4.1.7 defines a 2-query, $\neq$-based Long Code test to be a probability distribution on pairs $(x, y) \in \{-1, 1\}^n \times \{-1, 1\}^n$. Since we think of the Long Code test as testing $f(x) \neq f(y)$ and since $\neq$ is symmetric, there is no loss in generality if we insist that the probability distribution be symmetric in x and y. But such a symmetric distribution on $\{-1, 1\}^n \times \{-1, 1\}^n$ is identical to a weighted undirected graph G on $\{-1, 1\}^n$, with self-loops allowed. Note that this is an embedded graph, with the additional property that the vertex set is (a subset of) $\{-1, 1\}^n$. Further, if $f : \{-1, 1\}^n \to \{-1, 1\}$ is the function being tested, then $\frac{1}{2} - \frac{1}{2} f(x) f(y)$ is 1 if $f(x) \neq f(y)$ and 0 if $f(x) = f(y)$. Hence the probability that f passes the test is just $\mathrm{val}_G(f)$. Extending this definition to functions $f : \{-1, 1\}^n \to [-1, 1]$, we have the following:

Definition 4.7.2. *A* dictator-vs.-quasirandom test *for n-bit functions* $f : \{-1, 1\}^n \to [-1, 1]$ *is an embedded graph T whose vertex set is* $\{-1, 1\}^n$. *The* value *of the test on f is* $\mathrm{val}_T(f)$, *and this is sometimes referred to as the* probability that T passes/accepts f.

Our notion of the 'completeness' of a dictator-vs.-quasirandom test is essentially as in Definition 4.1.8: the least probability with which one of the Dictators passes:

Definition 4.7.3. *The ith Dictator function* $\chi_i : \{-1, 1\}^n \to \{-1, 1\}$ *is defined by* $\chi_i(x) = \sqrt{n} \cdot x_i$.

Definition 4.7.4. *The* completeness *of an n-bit dictator-vs.-quasirandom test T is*

$$\mathrm{Completeness}(T) = \min_{i \in [n]} \{\mathrm{val}_T(\chi_i)\}$$

The average of the probabilities with which Dictators pass a test T is precisely its spread:

Proposition 4.7.5. *Given an n-bit dictator-vs.-quasirandom test $T = (\{-1,1\}^n, E)$, we have*

$$\mathrm{Spread}(T) = \underset{i \in [n]}{\mathrm{avg}}\{\mathrm{val}_T(\chi_i)\}.$$

Hence $\mathrm{Spread}(T) \geq \mathrm{Completeness}(T)$.

Proof.

$$\mathrm{Spread}(T) = \underset{(x,y) \sim E}{\mathbf{E}}[\tfrac{1}{2} - \tfrac{1}{2}x \cdot y] = \underset{(x,y) \sim E}{\mathbf{E}}[\tfrac{1}{2} - \tfrac{1}{2}\textstyle\sum_{i=1}^n x_i y_i] = \tfrac{1}{n}\textstyle\sum_{i=1}^n \underset{(x,y) \sim E}{\mathbf{E}}[\tfrac{1}{2} - \tfrac{1}{2}n x_i y_i] = \underset{i \in [n]}{\mathrm{avg}}\{\mathrm{val}_T(\chi_i)\}.$$

□

As discussed in section 4.1.4 we use a weakened soundness notion for dictator-vs.-quasirandom tests; specifically, these tests only need to reject functions that are sufficiently 'quasirandom'. This soundness condition allows us to get large completeness/soundness gaps despite using only 2 queries. The notion of being 'quasirandom' is, for all intents and purposes, the same as the notion of having small 'low-degree influences' introduced in [99] and used in previous papers on UNIQUE-GAMES-hardness. We will make a very slightly different definition because we feel it is more natural. To make this definition we need to recall the basics of Fourier analysis of Boolean functions.

Analogous to the Hermite analysis described in section 4.3.3, the space of functions $L^2(\{-1,1\}^n)$ under the uniform distribution has a complete orthonormal basis given by the monomials $(\chi_S)_{S \subseteq [n]}$:

$$\chi_S(x) = \prod_{i \in S}(\sqrt{n} \cdot x_i).$$

One can uniquely express any function $f : \{-1,1\}^n \to \mathbb{R}$ via its Fourier expansion,

$$f = \sum_{S \subseteq [n]} \widehat{f}(S)\chi_S.$$

We now introduce quasirandom functions:

Definition 4.7.6. *For $0 \leq \epsilon, \delta \leq 1$, we say a function $f : \{-1,1\}^n \to [-1,1]$ is (ϵ,δ)-quasirandom if for each $i \in [n]$,*

$$\mathrm{Inf}_i^{(1-1-\delta)}(f) \leq \epsilon,$$

where we define the $(1-\delta)$-attenuated influence of i on f to be

$$\mathrm{Inf}_i^{(1-1-\delta)}(f) = \sum_{\substack{S \subseteq [n] \\ i \in S}} (1-\delta)^{|S|-1}\widehat{f}(S)^2.$$

Note that this definition becomes stricter when ϵ or δ decreases; we think of functions as being 'more quasirandom' when δ and (especially) ϵ are small. As an example, Dictator functions χ_i are the antithesis of being quasirandom; in particular, if $\epsilon < 1$ then

χ_i is not (ϵ, δ)-quasirandom even for $\delta = 1$.[8] On the other hand, the Majority function is extremely quasirandom; specifically, $(O(\frac{1}{\sqrt{n}}), 0)$-quasirandom. We have chosen the name quasirandom based on the 'Invariance Principle' from [116], which essentially states that if $f : \{-1, 1\}^n \to [-1, 1]$ is very quasirandom, then the distribution of

$$\sum_{S \subseteq [n]} \hat{f}(S) \prod_{i \in S} X_i$$

is nearly unchanged whether one takes the X_i's to be independent ± 1 bits or independent $N(0, 1)$ Gaussians.

Having defined quasirandom functions, we give the soundness notion for our tests:

Definition 4.7.7. *The (ϵ, δ)-soundness of a dictator-vs.-quasirandom test T for functions $f : \{-1, 1\}^n \to [-1, 1]$ is*

$$\text{Soundness}_{\epsilon, \delta}(T) = \max\{\text{val}_T(f) : f \text{ is } (\epsilon, \delta)\text{-quasirandom}\}.$$

Given this definition, the most natural Property Testing question to ask is how far apart completeness and soundness can be for dictator-vs.-quasirandom tests:

Definition 4.7.8. *We call the pair (c, s) a* dictator-vs.-quasirandom test (ϵ, δ)-gap *if for all sufficiently large n, there is a dictator-vs.-quasirandom test $T^{(n)}$ for functions $f : \{-1, 1\}^n \to [-1, 1]$ with $\text{Completeness}(T^{(n)}) \geq c$ and $\text{Soundness}_{\epsilon, \delta}(T^{(n)}) \leq s$. We call the pair (c, s) simply a* dictator-vs.-quasirandom test gap *if $\forall \eta > 0, \exists \epsilon, \delta > 0$ such that $(c, s + \eta)$ is a dictator-vs.-quasirandom test (ϵ, δ)-gap.*

Definition 4.7.9. *The* dictator-vs.-quasirandom gap curve *is the function* $\text{Gap}_{\text{Test}} : [\frac{1}{2}, 1] \to [\frac{1}{2}, 1]$ *defined by*

$$\text{Gap}_{\text{Test}}(c) = \min\{s : (c, s) \text{ is a dictator-vs.-quasirandom test gap}\}.$$

(It is immediate from the definitions that this min is achieved; i.e., we needn't write inf.)

In the next section we will show that $\text{Gap}_{\text{Test}}(c) \leq S(c)$; substituting this into these theorems yields our results from section 4.1.6; the subsequent section will be devoted to the inequality $\text{Gap}_{\text{Test}}(c) \geq S(c)$, whose proof completes the result $\text{Gap}_{\text{Test}}(c) = S(c)$. Although the inequality $\text{Gap}_{\text{Test}}(c) \geq \text{Gap}_{\text{SDP}}(c)$ was already implicitly proved in [107], we will give an alternate direct proof which clarifies the connection between SDP rounding algorithms and dictator-vs.-quasirandom testing. Finally, in the last section we will connect dictator-vs.-quasirandom tests with the SDP-hardness constructions in [3, 4, 86].

4.8 $\text{Gap}_{\text{Test}}(c) \leq S(c)$: **Invariance Principle**

To upper-bound $\text{Gap}_{\text{Test}}(c)$, we need to determine dictator-vs.-quasirandom tests with completeness at least c for which all quasirandom functions pass with small probability. Studying just how small this soundness can be is very similar to searching for the largest

---

[8] We take $0^0 = 1$ in the definition.

possible SDP gap, discussed in section 4.2. For example, given a particular test T on $\{-1,1\}^n$ with $\mathrm{Completeness}(T) \geq c$ and $\mathrm{Soundness}_{\epsilon,\delta}(T) \leq s$, one can symmetrize it with respect to all $2^n n!$ symmetries of $\{-1,1\}^n$, forming T'. Then one still has $\mathrm{Completeness}(T') \geq c$ and $\mathrm{Soundness}_{\epsilon,\delta}(T') \leq s$, and furthermore T' has the property that the probability of choosing a pair (x,y) depends only on its Hamming distance; i.e., only on $\langle x,y\rangle$. Just as we switched from $\mathscr{S}_P^{(d)}$ (which insisted on $\langle x,y\rangle$ being precisely ρ) to the analytically-easier $\mathscr{G}_P^{(d)}$, it is natural to switch to the version of symmetrized tests with independence across coordinates:

Definition 4.8.1. *We define the* noise sensitivity mixture test $\mathcal{T}_P^{(n)}$ *on* $\{-1,1\}^n$ *by analogy with Gaussian mixture graphs. In particular we define* (x,y) *to be ρ-correlated n-bit strings if x is drawn uniformly from $\{-1,1\}^n$ and y is formed by taking $y_i = x_i$ with probability $\frac{1}{2} + \frac{1}{2}\rho$ and $y_i = -x_i$ with probability $\frac{1}{2} - \frac{1}{2}\rho$, independently across i.*

We remark that a ρ-correlated pair (x,y) has $\langle x,y\rangle$ tightly concentrated around ρ, and that further:

Fact 4.8.2. $\mathrm{Completeness}(\mathcal{T}_P^{(n)}) = \mathrm{Spread}(P) = \mathbf{E}_{\rho\sim P}[\frac{1}{2} - \frac{1}{2}\rho]$.

Also, given $f : \{-1,1\}^n \to \mathbb{R}$ we use the notation

$$\mathbb{S}_\rho(f) = \mathop{\mathbf{E}}_{\substack{(x,y)\ \rho\text{-corr'd} \\ n\text{-bit strings}}} [f(x)f(y)].$$

The reader is warned that we use the notation $\mathbb{S}_\rho(f)$ for both $f : \{-1,1\}^n \to \mathbb{R}$ and $f \in L^2(\mathbb{R}^n)$ with the Gaussian distribution. For more on noise sensitivity tests, see [99].

Having decided that the best dictator-vs.-quasirandom gaps will occur essentially with noise sensitivity mixture tests, the ideas from section 4.2.3 again apply. The Hermite and Fourier formulas for noise stability are the same and we again conclude that the optimal mixture should come from a $(1,\rho_0)$-distribution. This provides an explanation for why such tests were useful in [102].

Finally, to upper-bound the value of quasirandom functions on noise sensitivity $(1,\rho_0)$-mixture tests, we use the Invariance Principle of [116] (which is also stated in Section 3.2.1 without explicitly giving the error bound) to reduce to the analysis of the MAX CUT in Gaussian mixture graphs. Then Theorem 4.3.4 can be used to get an upper bound of $S(c)$. More precisely, we prove the following theorem:

Theorem 4.8.3. *Let P be any $(1,\rho_0)$-distribution and let T denote the dictator-vs.-quasirandom test $\mathcal{T}_P^{(n)}$. Then for any $\tau > 0$,*

$$\mathrm{Soundness}_{\tau,\Omega(1/\log(1/\tau))}(T) \leq \sup_{\substack{r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \mathrm{val}_{\mathscr{G}_P}(r) + O(\log(1/\tau)^{-1/8}).$$

Before proving Theorem 4.8.3, let us see how it implies the desired result:

Corollary 4.8.4. $\mathrm{Gap}_{\mathrm{Test}}(c) \leq S(c)$.

Proof. Let P be the $(1,\rho_0)$-distribution with mean $1-2c$ achieving the minimum in the definition of $S(c)$ (or rather, in Corollary 4.4.14). Writing $T = \mathcal{T}_P^{(n)}$, we have $\mathrm{Completeness}(T) =$

c by Fact 4.8.2. Now by definition,

$$\sup_{\substack{r:\mathbb{R}\to[-1,1]\\ \text{increasing, odd}}} \mathrm{val}_{\mathscr{G}_P}(r)$$

is precisely $S(c)$. Hence Theorem 4.8.3 implies that the (ϵ,δ)-soundness of T can be made at most $S(c)$ plus an arbitrarily small amount, by taking ϵ and δ sufficiently small. This establishes $\mathrm{Gap}_{\mathrm{Test}}(c) \le S(c)$. $\qquad\square$

4.8.1 Proof of Theorem 4.8.3

The proof is an extension of the proof of the Majority Is Stablest theorem from [116]. Let P, T, and τ be as in the statement of the theorem, and let $f : \{-1,1\}^n \to [-1,1]$ be a $(\tau, \Omega(1/\log(1/\tau)))$-quasirandom function. We need to show that

$$\mathrm{val}_T(f) = \mathop{\mathbf{E}}_{\rho\sim P} \mathop{\mathbf{E}}_{\substack{(x,y)\ \rho\text{-corr'd}\\ n\text{-bit strings}}} [\tfrac{1}{2} - \tfrac{1}{2}f(x)f(y)] = \tfrac{1}{2} - \tfrac{1}{2}\mathop{\mathbf{E}}_{\rho\sim P}\big[\mathbb{S}_\rho(f)\big],$$

is, up to an additive $O(\log(1/\tau)^{-1/8})$, at most

$$\sup_{\substack{r:\mathbb{R}\to[-1,1]\\ \text{increasing, odd}}} \mathrm{val}_{\mathscr{G}_P}(r) \quad = \quad \sup_{\substack{r:\mathbb{R}\to[-1,1]\\ \text{increasing, odd}}} \left(\tfrac{1}{2} - \tfrac{1}{2}\mathop{\mathbf{E}}_{\rho\sim P}\big[\mathbb{S}_\rho(r)\big]\right).$$

Equivalently, we must show

$$\mathop{\mathbf{E}}_{\rho\sim P}\big[\mathbb{S}_\rho(f)\big] \quad\ge\quad \inf_{\substack{r:\mathbb{R}\to[-1,1]\\ \text{increasing, odd}}} \mathop{\mathbf{E}}_{\rho\sim P}\big[\mathbb{S}_\rho(r)\big] \ -O(\log(1/\tau)^{-1/8}). \tag{4.25}$$

Let us write p for the weight of P on ρ_0. Then the left side of (4.25) is

$$(1-p)\mathbf{E}[f^2] + p\,\mathbb{S}_{\rho_0}(f).$$

As in the proof of Theorem 4.3.4, this quantity can only decrease if we replace f by f^{odd}, in which case it becomes

$$(1-p)\mathbf{E}[f^2] - p\,\mathbb{S}_{-\rho_0}(f), \tag{4.26}$$

analogous to (4.8). (Note that a similar formula will arise on the right side of (4.25), since the r's are odd.) Since f^{odd} has the same Fourier expansion as f except with the even-degree terms dropped, we have that $\mathrm{Inf}_i^{(1-\delta)}(f^{\mathrm{odd}}) \le \mathrm{Inf}_i^{(1-\delta)}(f)$, and hence $f = f^{\mathrm{odd}}$ is still (ϵ,δ)-quasirandom.

We now set $\gamma = O\left(\frac{\log\log(1/\tau)}{\log(1/\tau)}\right)$ and distinguish the two cases $\rho_0 \le -1+3\gamma$ and $\rho_0 > -1+3\gamma$:

90

Case 1: $\rho_0 \le -1 + 3\gamma$. In this case we use $\mathbb{S}_{-\rho_0}(f) \le \mathbb{S}_1(f) = \mathbf{E}[f^2]$ to deduce that (4.26) is at least $1 - 2p$. On the other hand, by taking $r = \mathrm{sgn}$ (which is increasing and odd), we conclude that the term on the right side of (4.25) satisfies

$$\inf_{\substack{r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \mathbf{E}_{\rho\sim P}\left[\mathbb{S}_\rho(r)\right] \le (1-p)\mathbf{E}[\mathrm{sgn}^2] - p\mathbb{S}_{-\rho_0}(\mathrm{sgn}) = (1-p) - p(1 - \Theta(\sqrt{\gamma})) = 1 - 2p + \Theta(\sqrt{\gamma}),$$

where we used the estimate $\mathbb{S}_{1-\delta}(\mathrm{sgn}) = 1 - \Theta(\sqrt{\delta})$. Since $\Theta(\sqrt{\gamma}) \ll O(\log(1/\tau)^{-1/8})$, the proof of (4.25) in this case is complete.

Case 2: $\rho_0 > -1 + 3\gamma$. In this case we follow the arguments from [116]'s proof of the Majority Is Stablest theorem. Write $\rho = -\rho_0 < 1 - 3\gamma$, and express $\rho = \rho' \cdot (1-\gamma)^2$. We let $g \in L^2(\mathbb{R}^n)$ be the multilinear polynomial

$$g(x_1, \ldots, x_n) = \sum_{S \subseteq n} (1-\gamma)^{|S|} \widehat{f}(S) \prod_{i \in S} x_i,$$

and we let $\widetilde{g} : \mathbb{R}^n \to [-1, 1]$ be the function defined by

$$\widetilde{g}(\boldsymbol{x}) = \begin{cases} g(\boldsymbol{x}) & \text{if } |g(\boldsymbol{x})| \le 1, \\ \mathrm{sgn}(g(\boldsymbol{x})) & \text{else.} \end{cases}$$

We note that f being odd implies that both g and $\widetilde{g}$ are odd. Since

$$\mathbf{E}[f^2] = \sum_{S \subseteq [n]} \widehat{f}(S)^2 = \sum_{S \in \mathbb{N}^n} \widehat{g}(S)^2 = \mathbf{E}[g^2] \ge \mathbf{E}[\widetilde{g}^2],$$

we have

$$(4.26) \ge (1-p)\mathbf{E}[\widetilde{g}^2] - p\mathbb{S}_\rho(f).$$

Further, using the fact that f is $(\tau, \Omega(1/\log(1/\tau)))$-quasirandom, the Invariance Principle-based arguments in [116] imply that

$$|\mathbb{S}_\rho(f) - \mathbb{S}_{\rho'}(\widetilde{g})| \le \tau^{\Omega(\gamma)}.$$

Hence we have

$$(4.26) \quad \ge (1-p)\mathbf{E}[\widetilde{g}^2] - p\mathbb{S}_{\rho'}(\widetilde{g}) - \tau^{\Omega(\gamma)} = (1-p)\mathbf{E}[\widetilde{g}^2] + p\mathbb{S}_{-\rho'}(\widetilde{g}) - \tau^{\Omega(\gamma)} = \left(1 - 2\mathrm{val}_{\mathscr{G}_{P'}^{(n)}}(\widetilde{g})\right) - \tau^{\Omega(\gamma)},$$

where the first equality uses the fact that $\widetilde{g}$ is odd and where P' the probability distribution that puts weight $1 - p$ on 1 and weight p on $-\rho'$. But P' is a '$(1, \rho_0)$-distribution', and hence Theorem 4.3.4 implies that

$$\mathrm{val}_{\mathscr{G}_{P'}^{(n)}}(\widetilde{g}) \le \sup_{\substack{r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \mathrm{val}_{\mathscr{G}_{P'}}(r).$$

Thus we have

$$(4.26) \quad \ge \inf_{\substack{r:\mathbb{R}\to[-1,1] \\ \text{increasing, odd}}} \mathbf{E}_{\rho\sim P'}\left[\mathbb{S}_\rho(r)\right] - \tau^{\Omega(\gamma)}.$$

By taking the constant in the definition of γ large enough we get $\tau^{\Omega(\gamma)} \ll O(\log(1/\tau)^{-1/8})$. Thus to complete the proof of (4.25), we only need to relate the inf with P to the inf with P', using the fact that $|(-\rho') - \rho_0| \le O(\gamma)$. This can be done by using the discretization Lemmas 4.4.6 and 4.4.13; the resulting error term is at most $O(\gamma^{1/7}) \le O(\log(1/\tau)^{-1/8})$, as required.

4.9 $\mathrm{Gap}_{\mathrm{Test}}(c) \geq S(c)$: RPR^2 Algorithms Imply Testing Lower Bounds

In this section we discuss 'lower bounds' for the dictator-vs.-quasirandom testing problem; i.e., proofs that any test T with $\mathrm{Completeness}(T) = c$ cannot have $\mathrm{Soundness}_{\epsilon,\delta}(T)$ which is too small. As mentioned earlier, Khot and Vishnoi's result can be used to get such lower bounds: it gives a long translation of a (c, s) dictator-vs.-quasirandom test gap into a $(c - \eta, s + \eta)$ SDP gap (with triangle inequality, even), for arbitrarily small η. This means that an SDP-rounding guarantee can be used to rule out the existence of strong dictator-vs.-quasirandom tests. A similar idea arises from the earlier Theorem 4.1.11, which shows that a (c, s) dictator-vs.-quasirandom test gap can be translated into into a $c - \eta$ vs. $s + \eta$ UGC-hardness result for MAX CUT. Since one feels it is unlikely that the UGC would be disproved via an elaborate reduction to MAX CUT followed by a too-strong SDP-rounding algorithm, Theorem 4.1.11 also suggests that SDP-rounding algorithms should be able to prove dictator-vs.-quasirandom testing lower bounds.

In this section we show explicitly and directly that RPR^2 algorithms give rise to dictator-vs.-quasirandom testing lower bounds. More specifically, the following theorem implies (and indeed is slightly stronger than) the result $\mathrm{Gap}_{\mathrm{Test}}(c) \geq S(c)$:

Theorem 4.9.1. *Let $\epsilon > 0$ be given. Then for all $n \geq O(1/\epsilon^7)$, if T is any dictator-vs.-quasirandom test for functions $f : \{-1, 1\}^n \to [-1, 1]$ satisfying $\mathrm{Completeness}(T) \geq c$, then $\mathrm{Soundness}_{\epsilon,0}(T) \geq S(c) - \epsilon$.*

Proof. Let T be a such a test. As described in section 4.7, T can be thought of as an embedded graph on the vertex set $\{-1, 1\}^n \subseteq S^{n-1}$. Write P for the ρ-distribution of T, and recall from Proposition 4.7.5 that $\mathrm{Spread}(P) \geq \mathrm{Completeness}(T) \geq c$.

Imagine we now run our RPR^2 Algorithm 4.2.4) on T, with the discretization parameter set to $\epsilon' := \epsilon/K$. By Theorem 4.4.3, it will at some point hit upon an ϵ'-discretized, increasing, odd rounding function $r^* : \mathbb{R} \to [-1, 1]$ which satisfies

$$\mathrm{Alg}_{\mathrm{RPR}^2}(T) = \mathrm{val}_{\mathscr{G}_P}(r^*) \geq S(\mathrm{Spread}(P)) - O(\epsilon') \geq S(c) - \epsilon/2, \qquad (4.27)$$

assuming K is a sufficiently large constant. (Here we also used that S is increasing.) Recall that when we run the RPR^2 algorithm with r^*, it chooses a random n-dimensional Gaussian $\mathbf{Z}$ and outputs the fractional cut $f_{\mathbf{Z}} : \{-1, 1\}^n \to [-1, 1]$ defined by

$$f_{\mathbf{Z}}(x) = r^*(x \cdot \mathbf{Z}).$$

Thus (4.27) is equivalent to

$$\mathbf{E}_{\mathbf{Z}}[\mathrm{val}_T(f_{\mathbf{Z}})] \geq S(c) - \epsilon/2.$$

Our goal is now to show the intuitively plausible claim that that $f_{\mathbf{Z}}$ is very likely to be a quasirandom Boolean function:

Claim 4.9.2. *With probability at least $1 - O(1/n)$ over over the choice of $\mathbf{Z}$, the function $f_{\mathbf{Z}}$ is $(O(\sqrt{\ln n/n})/\epsilon'^2, 0)$-quasirandom.*

With our choice of $n \geq O(1/\epsilon^7)$, this claim implies that with probability at least $1 - \epsilon/2$ the function $f_{\mathbf{Z}}$ is $(\epsilon, 0)$-quasirandom. This in turn completes the proof of the theorem, since it implies

$$\mathop{\mathbf{E}}_{\mathbf{Z}}[\mathrm{val}_T(f_{\mathbf{Z}}) \mid f_{\mathbf{Z}} \text{ is } (\epsilon, 0)\text{-quasirandom}] \geq S(c) - \epsilon/2 - \epsilon/2.$$

Thus there must exist an $(\epsilon, 0)$-quasirandom $f : \{-1, 1\}^n \to [-1, 1]$ with $\mathrm{val}_T(f) \geq S(c) - \epsilon$, and we conclude that $\mathrm{Soundness}_{\epsilon, 0}(T) \geq S(c) - \epsilon$ as needed.

Proof. (of Claim 4.9.2.) Given $\mathbf{Z}$, let us write $f = f_{\mathbf{Z}}$ for notational simplicity. Let us also write $\gamma = O(\sqrt{\ln n/n})/\epsilon'^2$. We need to show that with probability at least $1 - O(1/n)$,

$$\gamma \geq \mathrm{Inf}_i^{(0)}(f) = \mathrm{Inf}_i(f) = \mathop{\mathbf{E}}_{x \in \{-1,1\}^n} \left[\left(\frac{(f(x^{(i=1)}) - f(x^{(i=-1)}))}{2} \right)^2 \right] \qquad \text{for all } 1 \leq i \leq n. \qquad (4.28)$$

Here we have used the notation $x^{(i=b)}$ for the string x with the ith coordinate set to $b/\sqrt{n}$, along with the well-known alternate definition of Boolean influences (see [99]). In fact, we will show that (4.28) holds whenever both of the following hold:

$$|Z_i| \leq 2\sqrt{\ln n} \qquad \text{for all } 1 \leq i \leq n; \qquad (4.29)$$

$$\tfrac{1}{2}n \leq \|\mathbf{Z}\|_2^2 \leq \tfrac{3}{2}n. \qquad (4.30)$$

Since $|Z_i| \leq 2\sqrt{\ln n}$ for each i except with probability at most $O(1/n^2)$, we get that (4.29) holds except with probability $O(1/n)$. It's also well known (and the proof is sketched in the proof of Theorem 4.3.3) that (4.30) holds except with exponentially small probability in n. Thus both (4.29) and (4.30) hold except with probability at most $O(1/n)$, as necessary.

Let us henceforth fix $\mathbf{Z} = Z$ satisfying (4.29) and (4.30). We wish to prove now that (4.28) holds. We will show that it holds for $i = n$, and the fact that it holds for $1 \leq i < n$ will follow by an identical argument. So we must prove that

$$\begin{aligned} \gamma \;\geq\; & \mathop{\mathbf{E}}_{x \in \{-1,1\}^n} \left[\left(\frac{(f(x^{(n=1)}) - f(x^{(n=-1)}))}{2} \right)^2 \right] \\ =\; & \frac{1}{4} \mathop{\mathbf{E}}_{x \in \{-1,1\}^{n-1}} \left[\left(r^* \left(\sum_{i=1}^{n-1} Z_i x_i + \frac{Z_i}{\sqrt{n}} \right) - r^* \left(\sum_{i=1}^{n-1} Z_i x_i - \frac{Z_i}{\sqrt{n}} \right) \right)^2 \right]. \end{aligned}$$

Using the fact that r is ϵ'-discretized, we can even show the following stronger result:

$$\mathop{\mathbf{Pr}}_{x \in \{-1,1\}^{n-1}} \left[\sum_{i=1}^{n-1} Z_i x_i \pm \frac{Z_i}{\sqrt{n}} \text{ fall into different intervals from } \mathscr{I}_{\epsilon'} \right] \leq \gamma. \qquad (4.31)$$

Let σ^2 denote $\sum_{i=1}^{n-1} Z_i^2/n$, which by (4.29) and (4.30) satisfies $\tfrac{1}{3} \leq \sigma^2 \leq \tfrac{3}{2}$. Now the random variable $\sum_{i=1}^{n-1} Z_i x_i$ has distribution close to that of a mean-zero Gaussian with

93

variance σ^2; more specifically, using the Berry-Esseen Theorem we have that for every interval I,

$$\left| \mathbf{Pr}\left[\sum_{i=1}^{n-1} Z_i x_i \in I \right] - \mathbf{Pr}[N(0,\sigma^2) \in I] \right| \le O\left(\frac{\max_i |Z_i|}{\sigma \sqrt{n}} \right) = O(\sqrt{\log n/n}). \qquad (4.32)$$

The analysis is now very similar to the analysis in Claim 4.4.7. Given any interval $J \in \mathscr{I}_{\epsilon'}$, let J' denote the subinterval gotten by moving the boundary points inwards by $3\sqrt{\ln n/n}$. The analysis from Claim 4.4.7 implies that a standard Gaussian will fall into one of the J' intervals except with probability $O(\sqrt{\ln n/n}/\epsilon'^2)$, and only the constant in the $O(\cdot)$ changes if we consider instead a Gaussian with variance $\sigma^2 \in [\frac{1}{3}, \frac{3}{2}]$. Hence the same is true of the random variable $\sum_{i=1}^{n-1} Z_i x_i$, using (4.32). But whenever this random variable falls into some J', we get that $\sum_{i=1}^{n-1} Z_i x_i \pm \frac{Z_i}{\sqrt{n}}$ are both in the associated J, since $|Z_i| \le 2\sqrt{\ln n}$. Since we took $\gamma = O(\sqrt{\ln n/n})/\epsilon'^2$, we have that (4.31) indeed holds, as needed. $\qquad \square$

$$\text{(Theorem 4.9.1)} \qquad \square$$

4.10 Hardness Results for RPR^2 Algorithms

In this section we revisit the constructions of Karloff [86], Alon and Sudakov [3], and Alon, Sudakov, and Zwick [4]. The purpose of these constructions is to demonstrate that the *analysis* of the Goemans-Williamson approximation guarantee is tight (and likewise for the Zwick [142] approximation guarantee, in the case of [4]). For now we discuss [3, 86], returning to [4] at the end of the section.

The works [3, 86] consider the graph T on $\{-1,1\}^n$ in which a pair of vertices (x,y) is connected if and only if the vertices' inner product is exactly $1-2c$; here c is any rational parameter in $(\frac{1}{2},1)$.[9] The authors show (for infinitely many n) that the identity map is an optimal SDP embedding, and hence $\mathrm{Opt}(T) = \mathrm{Sdp}(T) = c$. On the other hand, since every edge in the embedded graph connects vectors with inner product $1-2c$, the expected value of the cut output by the GW algorithm (RPR^2 with the rounding function sgn) is only $\arccos(1-2c)/\pi$. Thus (in expectation, at least) the GW approximation curve satisfies $\mathrm{Apx}_{GW}(c) \le \arccos(1-2c)/\pi$.

As the reader can clearly see, this construction can be viewed as a dictator-vs.-quasirandom test with completeness c. Indeed, the noise sensitivity test of [99] is almost identical to it; the only difference is that the noise sensitivity test picks edges with *expected* inner product $1-2c$ rather than precise inner product $1-2c$. The 'soundness' result used in [3, 86] is that the average value among 'random halfspace functions' $\mathrm{sgn}(x \cdot \mathbf{Z})$ is at most $\arccos(1-2c)/\pi$. As we saw in section 4.9, these random halfspace functions are almost surely quasirandom.

---

[9] The earlier work of [86] was slightly more complicated as it only included vertices with Hamming weight exactly $n/2$.

The result from [3, 86] has some additional strengths and weaknesses. One strength is that the SDP embedding used has all of its unit vectors on the discrete cube $\{-1,1\}^n$; hence these points satisfy the triangle inequalities, and indeed satisfy all 'valid' inequalities (see [86]). Thus $\text{Apx}_{GW}(c)$ is still at most $\arccos(1-2c)/\pi$ even if the SDP with triangle inequalities is used. A weakness of the original result was that it only stated that the *expected* value of the cut GW produces is at most $\arccos(1-2c)/\pi$; it said nothing, e.g., about what happens if the GW algorithm is run several times and the best resulting cut is selected. For the noise sensitivity version of the test, a result in [99] shows that GW achieves at most $\arccos(1-2c)/\pi + o(1)$ with high probability. However, Feige and Schechtman [50] showed an even better result:

Theorem 4.10.1 ([50]). *For any rational $c \in (\frac{1}{2}, 1)$ and any $\eta > 0$, there are optimally embedded graphs G, with arbitrarily large numbers of vertices, satisfying:*

- $\text{Opt}(G) = \text{Sdp}(G) = c$;
- *the vectors in G satisfy the triangle inequalities;*
- *every halfspace cut has value at most $\arccos(1-2c)/\pi + \eta$.*

The conclusion from this result is that running the RPR^2 algorithm A with the rounding function sgn cannot achieve $\text{Apx}_A(c) > \arccos(1-2c)/\pi$, even if: (i) A uses the SDP with triangle inequalities; and, (ii) A is not required to choose $\mathbf{Z}$ at random but is allowed to use the best possible $\mathbf{Z}$ of length $\sqrt{n}$. (When $r = \text{sgn}$, the length of $\mathbf{Z}$ is irrelevant and may as well be fixed.)

Feige and Schechtman prove Theorem 4.10.1 (non-constructively) as follows: They begin with the embedded graph T on $\{-1,1\}^n$ constructed in [3, 86]. They then essentially take G to consist of m disjoint copies of T, each embedded in a random n-dimensional subspace of $\mathbb{R}^d$. If $d \gg n^2 \log m$, then the triangle inequalities hold in G with high probability; on the other hand, if d is not too large then it can be shown that every halfspace cut of G has value at most $\arccos(1-2c)/\pi + \eta$.

We now prove a generalization of Theorem 4.10.1. We would like to emphasize that our proof follows Feige and Schechtman's extremely closely.

Theorem 4.10.2. *Suppose (c,s) is a dictator-vs.-quasirandom gap, and $\eta > 0$. Fix any RPR^2 rounding function r which is piecewise constant.*[10] *Then there are embedded graphs G in S^{d-1}, with arbitrarily large numbers of vertices, satisfying:*

- $\text{Opt}(G) \geq c$;
- *the vectors in G satisfy the triangle inequalities;*
- *every fractional cut $f_\mathbf{Z}$ of the form $f_\mathbf{Z}(u) = r(u \cdot \mathbf{Z})$ satisfies $\text{val}_G(f_\mathbf{Z}) \leq s + \eta$, as long as $\|\mathbf{Z}\|_2 = \Theta(\sqrt{d})$.*

Proof. Select $\epsilon, \delta > 0$ and a family $(T^{(n)})$ of dictator-vs.-quasirandom tests, with $T^{(n)}$ operating on $\{-1,1\}^n$, such that $\text{Completeness}(T^{(n)}) \geq c$ and $\text{Soundness}_{\epsilon,\delta}(T_n) \leq s + \eta/3$, for all sufficiently large n. We would also like to assume that each $T^{(n)}$ is regular, meaning that each $x \in \{-1,1\}^n$ participates in the test with the same probability. We can ensure this by symmetrizing each $T^{(n)}$ with respect to the $2^n n!$ symmetries of $\{-1,1\}^n$, as discussed in

---

[10]As all functions implemented on a discrete computer must be.

section 4.8. (Alternatively, the dictator-vs.-quasirandom tests we will actually use, constructed in section 4.8, are already regular.)

As in [50], we take G to be m equally weighted disjoint copies of $T^{(n)}$, embedded on the unit d-dimensional sphere S^{d-1} with independent random orientations. Since Completeness$(T^{(n)}) \geq c$, certainly Opt$(G) = $ Opt$(T^{(n)}) \geq c$. Also, as shown in [50], if $d \gg n^2 \log m$ then the vectors in G satisfy the triangle inequalities with high probability; this uses the fact that the vectors in $T^{(n)}$ satisfy the triangle inequalities. It remains to analyze val$_G(f_{\mathbf{Z}})$ for all possible fractional cuts $f_{\mathbf{Z}}(u) := r(u \cdot \mathbf{Z})$ where $\|\mathbf{Z}\|_2 = \Theta(\sqrt{d})$. For concreteness, assume that this means $(1/c)\sqrt{d} \leq \|\mathbf{Z}\|_2 \leq c\sqrt{d}$ for some $c > 0$.

Let us consider the piecewise constant function r. Choose a small enough $\gamma > 0$ so that the set
$$\mathscr{B} := \bigcup\{[t - \gamma, t + \gamma] : t \text{ is a point of discontinuity for } r\}$$
has total measure at most $\epsilon\eta/O(\sqrt{c})$. Following [50], we now take a γ-net $\mathscr{N}$ for the set $\{\mathbf{Z} 1/c)\sqrt{d} \leq \|\mathbf{Z}\|_2 \leq c\sqrt{d}\}$; this can have cardinality $O(c\sqrt{d}/\gamma)^d$. We show that, with high probability over the orientations of G, both of the following hold for all $v \in \mathscr{N}$:

1. val$_G(f_v) \leq s + 2\eta/3$;
2. the fraction of vertices u of G for which $u \cdot v \in \mathscr{B}$ is at most $\eta/6$.

Having shown this, it follows that val$_G(f_{\mathbf{Z}}) \leq s + \eta$ for *all* $(1/c)\sqrt{d} \leq \|\mathbf{Z}\|_2 \leq c\sqrt{d}$. To see this for a given $\mathbf{Z}$, take v to be the closest net point. Then for every $u \in G$ we have $|u \cdot \mathbf{Z} - u \cdot v| \leq \|u\| \cdot \|\mathbf{Z} - v\| \leq \gamma$. It follows that $f_{\mathbf{Z}}(u) = f_v(u)$ except possibly when $u \cdot v \in \mathscr{B}$. But this occurs only for at most an $\eta/6$ fraction of vertices in G, and hence at most an $\eta/6$ fraction of edge weight, by regularity. It follows that $|\text{val}_G(f_{\mathbf{Z}}) - \text{val}_G(f_v)| \leq 2\eta/6$, and hence val$_G(f_{\mathbf{Z}}) \leq s + \eta$, as required.

It remains to prove that items (1) and (2) above indeed hold with high probability. Fix any $v \in \mathscr{N}$ and let $T_1, \ldots, T_m$ denote the randomly oriented copies of $T^{(n)}$ making up G. In analyzing some T_i vis-a-vis v, we imagine instead that the orientation of T_i is fixed and v is chosen randomly from the surface of the sphere of radius $\|v\|_2$. In this framework, let $\mathbf{Y}$ denote the projection of the random v onto the n-dimensional subspace containing T_i. Now the projection of a random vector from the surface of a sphere onto a lower-dimensional subspace yields a distribution which is close to Gaussian. In particular, since we are already assuming $d \gg n^2 \log m \geq O(n^2)$, the results in [39] imply that the variation distance between $\mathbf{Y}$ and the n-dimensional Gaussian distribution with coordinate variances equal to $\|v\|_2/\sqrt{d} \in [1/c, c]$ is at most $O(n/d) = O(1/n)$. If $\mathbf{Y}$ were truly drawn from that Gaussian distribution, then we would have the following (cf. the proof of Claim 4.9.2):

- the expected fraction of vertices u of T_i for which $u \cdot \mathbf{Y} \in \mathscr{B}$ is at most $O(\sqrt{c}|\mathscr{B}|)$;
- $|Y_i| \leq O(\sqrt{c \ln n})$ for all $1 \leq i \leq n$;
- $\frac{1}{2c}n \leq \|\mathbf{Y}\|_2^2 \leq \frac{3c}{2}n$.

Similar to the proof of Claim 4.9.2, the last two of these imply that f_v is a $(\epsilon, 0)$-quasirandom cut for T_i, as long as $O(\sqrt{c}\ln n/n) \leq \gamma$ and $O(\sqrt{c}|\mathscr{B}|) \leq \epsilon$. The latter holds by design; the

former holds so long as we take $n \geq \text{poly}(c/\gamma)$. But when $f_{\boldsymbol{v}}$ is a $(\epsilon, 0)$-quasirandom cut for T_i, we have $\text{val}_{T_i}(f_{\boldsymbol{v}}) \leq s + \eta/3$. Note also that $O(\sqrt{c}|\mathscr{B}|) \leq \eta/24$ by design.

Overall, we conclude that for each i independently we have $\text{val}_{T_i}(f_{\boldsymbol{v}}) \leq s + \eta/3$, except with probability at most $O(1/n)$ over the choice of orientations. If we ensure that $n \geq O(1/\eta)$, we conclude that the expected value of $\text{val}_{T_i}(f_{\boldsymbol{v}})$ is at most $s + \eta/2$. Similarly, we can conclude that the expected fraction of vertices $\boldsymbol{u}$ of T_i for which $\boldsymbol{u} \cdot \boldsymbol{v} \in \mathscr{B}$ is at most $\eta/12$. Since $\text{val}_G(f_{\boldsymbol{v}}) = \text{avg}_{i \in [m]} \text{val}_{T_i}(f_{\boldsymbol{v}})$, a Chernoff bound implies that item (1) above holds except with probability at most $\exp(-O(\eta^2 m))$. Similarly, item (2) above holds except with probability at most $\exp(-O(\eta^2 m))$. If we take $m \gg d \log d$ then this probability will be much smaller than $O(c\sqrt{d}/\gamma)^{-d}$ (treating c, γ, and η as constants), and so we get that both items (1) and (2) hold with high probability for all net points simultaneously, by a union bound.

As in [50], the overall constraints we have on m and d are that $n^2 \log m \ll d \ll m/\log m$, and this can clearly be realized. $\qquad\square$

We end this section by discussing the issue of self-loops and the construction of Alon, Sudakov, and Zwick [4]. If we use Theorem 4.10.2 with the noise sensitivity $(1, \rho_0)$-mixture tests constructed in section 4.8, we get a hard instance for RPR^2, but one that might be considered slightly unsatisfactory: this is because the embedded graph G constructed has self-loops. However one can't simply dismiss embedded graphs with self-loops, because optimally embedded graphs *can* have self-loops. In fact, Alon, Sudakov, and Zwick's construction is the following: for each $(1, \rho_0)$-mixture distribution, they construct a self-loopless graph for which the optimal SDP embedding is essentially the noise sensitivity $(1, \rho_0)$-mixture test. More precisely, it is the version in which vertices are connected if their inner product is *exactly* ρ_0 or 1. The technique of [4] involves taking the $(1, \rho_0)$-mixture test and replacing the self-loops by cliques, similar to the self-loop removal technique discussed in Section 4.12.

4.11 $\text{Gap}_{\text{SDP}}(c)$ **is Continuous**

In this Section we prove Proposition 4.3.2. The fact that $\text{Gap}_{\text{SDP}}(c)$ is increasing on $[\frac{1}{2}, 1]$ is immediate from the definition (since if $c' > c$, the inf for c' is over a subset of the inf for c). We mainly focus on the proof that $\text{Gap}_{\text{SDP}}(c)$ is continuous on $(\frac{1}{2}, 1)$; this requires only a simple trick — the use of the *isolated edge*. The proof of continuity at 1 requires appealing to Goemans-Williamson, and the continuity at $\frac{1}{2}$ is trivial. Finally, the proof that $\text{Gap}_{\text{SDP}}(c)$ is *strictly* increasing requires an *isolated clique* trick, plus an appeal to a result of Zwick [142].

Definition 4.11.1. *Given a graph G and a parameter $0 \leq \epsilon \leq 1$, we define the graph $G \sqcup \text{edge}_\epsilon$ to be the graph in which G's edge-weights are scaled by a factor of $1 - \epsilon$, and then two new vertices are added, with an edge between them of weight ϵ.*

The following is easy to verify:

Proposition 4.11.2. $\text{Sdp}(G \sqcup \text{edge}_\epsilon) = (1-\epsilon)\text{Sdp}(G) + \epsilon$ *and* $\text{Opt}(G \sqcup \text{edge}_\epsilon) = (1-\epsilon)\text{Opt}(G) + \epsilon$.

We now prove:

Proposition 4.11.3. $\mathrm{Gap}_{\mathrm{SDP}}(c)$ *is continuous on* $(\frac{1}{2}, 1)$.

Proof. We first prove right-continuity on $(\frac{1}{2}, 1)$. Suppose $c \in (\frac{1}{2}, 1)$, and let $s = \mathrm{Gap}_{\mathrm{SDP}}(c)$. Given any sufficiently small $\epsilon > 0$, assume $c < c' < c + (1-c)\epsilon/2 < 1$. By the definition of $\mathrm{Gap}_{\mathrm{SDP}}(c) = s$ we can find some graph G with $\mathrm{Sdp}(G) \geq c$ and $\mathrm{Opt}(G) \leq s + \epsilon/2$. Let $\widetilde{G} = G \sqcup \mathrm{edge}_{\epsilon/2}$. Then we have $\mathrm{Sdp}(\widetilde{G}) \geq (1 - \epsilon/2)c + \epsilon/2 = c + (1-c)\epsilon/2 > c'$, and further, $\mathrm{Opt}(\widetilde{G}) \leq (1 - \epsilon/2)(s + \epsilon/2) + \epsilon/2 \leq s + \epsilon$. This proves $\mathrm{Gap}_{\mathrm{SDP}}(c') \leq s + \epsilon$. Since $\mathrm{Gap}_{\mathrm{SDP}}$ is increasing, we have proven right-continuity at c.

The proof of left-continuity on $(\frac{1}{2}, 1)$ is similar. Suppose $c \in (\frac{1}{2}, 1)$, and let $s = \mathrm{Gap}_{\mathrm{SDP}}(c)$. Given any sufficiently small $\epsilon > 0$, assume $\frac{1}{2} < c - 2\epsilon(1-c) < c' < c$. For any graph G with $\mathrm{Sdp}(G) \geq c'$, let $\widetilde{G} = G \sqcup \mathrm{edge}_{2\epsilon}$. We have $\mathrm{Sdp}(\widetilde{G}) \geq (1 - 2\epsilon)c' + 2\epsilon = c' + 2\epsilon(1 - c') \geq c' + 2\epsilon(1 - c) \geq c$ and also $\mathrm{Opt}(\widetilde{G}) = (1 - 2\epsilon)\mathrm{Opt}(G) + 2\epsilon$. By the definition of $\mathrm{Gap}_{\mathrm{SDP}}(c) = s$, it holds that $\mathrm{Opt}(\widetilde{G}) \geq s$. Hence $(1 - 2\epsilon)\mathrm{Opt}(G) + 2\epsilon \geq s$ which implies $\mathrm{Opt}(G) \geq s - (1 - \mathrm{Opt}(G))2\epsilon \geq s - \epsilon$. This proves $\mathrm{Gap}_{\mathrm{SDP}}(c') \geq s - \epsilon$. Since $\mathrm{Gap}_{\mathrm{SDP}}$ is increasing, we have proven left-continuity at c. $\qquad\square$

We next check continuity at the endpoints, $c = \frac{1}{2}, 1$. It's easy to see that if $\mathrm{Sdp}(G) = 1$ then G must be bipartite and so $\mathrm{Opt}(G) = 1$. Hence $\mathrm{Gap}_{\mathrm{SDP}}(1) = 1$. Next, by taking the sequence of complete graphs K_m (each with total edge-weight 1), which satisfy $\mathrm{Opt}(K_m) \leq \frac{1}{2} + \frac{1}{m} \to \frac{1}{2}$ as $m \to \infty$, we see that $\mathrm{Gap}_{\mathrm{SDP}}(\frac{1}{2}) = \frac{1}{2}$. Thus to check continuity at the endpoints we need to show that $\lim_{c \to (1/2)^+} \mathrm{Gap}_{\mathrm{SDP}}(c) = \frac{1}{2}$ and $\lim_{c \to 1^-} \mathrm{Gap}_{\mathrm{SDP}}(c) = 1$.

The first of these follows simply because $\mathrm{Gap}_{\mathrm{SDP}}(c)$ is sandwiched between $\frac{1}{2}$ and c for all c. For the second of these, suppose G is any graph with $\mathrm{Sdp}(G) \geq 1 - \epsilon$. The analysis of Goemans and Williamson [59] implies that one can find a cut in G with value at least $1 - O(\sqrt{\epsilon})$. Thus $\mathrm{Gap}_{\mathrm{SDP}}(1 - \epsilon) \geq 1 - O(\sqrt{\epsilon})$, and so $\lim_{c \to 1^-} \mathrm{Gap}_{\mathrm{SDP}}(c) = 1$ as claimed.

Finally, we check that $\mathrm{Gap}_{\mathrm{SDP}}(c)$ is strictly increasing. For this we introduce isolated cliques:

Definition 4.11.4. *Given a graph G and two parameters $m \in \mathbb{N}$ and $0 \leq \epsilon \leq 1$, we define the graph $G \sqcup K_{m,\epsilon}$ to be the graph in which G's edge-weights are scaled by a factor of $1 - \epsilon$, and then an isolated m-clique is added, whose total edge-weight is ϵ.*

Using the fact that $\mathrm{Opt}(K_{m,1}) \leq \frac{1}{2} + \frac{1}{m}$, one can check:

Proposition 4.11.5. $\mathrm{Sdp}(G \sqcup K_{m,\epsilon}) \geq (1 - \epsilon)\mathrm{Sdp}(G) + \epsilon/2$ *and* $\mathrm{Opt}(G \sqcup K_{m,\epsilon}) \leq (1 - \epsilon)\mathrm{Opt}(G) + (\frac{1}{2} + \frac{1}{m})\epsilon$.

We now have:

Proposition 4.11.6. $\mathrm{Gap}_{\mathrm{SDP}}(c)$ *is strictly increasing on* $[\frac{1}{2}, 1]$.

Proof. It's enough to check this on $(\frac{1}{2}, 1)$. So suppose $\frac{1}{2} < c < c' < 1$, and write $s' = \mathrm{Gap}_{\mathrm{SDP}}(c')$. Zwick [142] was the first to show that $c' > \frac{1}{2}$ implies $s' > \frac{1}{2}$; Charikar and Wirth [30] specifically proved that $\mathrm{Sdp}(G) \geq \frac{1}{2} + \gamma$ implies $\mathrm{Opt}(G) \geq \frac{1}{2} + \Omega(\gamma/\log(1/\gamma))$. Thus we have $s' > \frac{1}{2}$. Write $\epsilon = (c' - c)/c'$. Select m large enough that $s' - \frac{1}{2} - \frac{1}{m}$ is still strictly positive. Finally, take $\delta > 0$ so that $\delta < (s' - \frac{1}{2} - \frac{1}{m})\epsilon$.

98

By definition of $\text{Gap}_{\text{SDP}}(c') = s'$, we can find a graph G' with $\text{Sdp}(G') \ge c'$ and $\text{Opt}(G') \le s' + \delta$. Let $G = G' \sqcup K_{m,\epsilon}$. Then $\text{Sdp}(G) \ge (1-\epsilon)c' + \epsilon/2 \ge (1-\epsilon)c' = c$. Further, $\text{Opt}(G) \le (1-\epsilon)(s'+\delta) + (\frac{1}{2} + \frac{1}{m})\epsilon \le s' + (\frac{1}{2} + \frac{1}{m} - s')\epsilon + \delta < s' - \delta + \delta = s'$. We conclude that $\text{Gap}_{\text{SDP}}(c) < s' = \text{Gap}_{\text{SDP}}(c')$. Thus $\text{Gap}_{\text{SDP}}(c)$ is indeed strictly increasing. $\qquad\square$

4.12 SDP Gaps Based on Infinite, Self-looped Graphs

In this Section we prove Proposition 4.3.1.

Proof. Write $G_0 = G$. We will transform G_0 into G_1, an infinite graph on vertex set B_d; then G_1 into G_2, a finite graph (with self-loops); then G_2 into G_3, a self-loopless graph; then G_3 into G_4, an unweighted graph. The desired graph will then be $G' = G_4$. The first transformation uses the idea of embedded graphs, and the remaining transformations are all previously known.

Let $g : \mathbb{R}^d \to B_d$ achieving the sup in the definition of $\text{Sdp}(G_0)$ to within ϵ. Let G_1 be the infinite graph on B_d given by pushing forward G_0 via g, i.e., $G_1(A, B) = G_0(g^{-1}(A), g^{-1}(B))$ (here we're identifying a graph with the probability measure defining its 'edge weights'). We immediately get $\mathbf{E}_{(x,y) \sim G_1}[\frac{1}{2} - \frac{1}{2}x \cdot y] \ge c - \epsilon$. We can think of this as saying:

$$\text{`Sdp}(G_1)\text{'} \ge c - \epsilon, \tag{4.33}$$

with the identity mapping as the embedding. Further,

$$\text{Opt}(G_1) \le s, \tag{4.34}$$

because for any fractional cut $h : B_d \to [-1, 1]$ for G_1, the cut $h \circ g : \mathbb{R}^d \to [-1, 1]$ for G_0 achieves the same value, $\mathbf{E}_{(x,y) \sim G_1}[\frac{1}{2} - \frac{1}{2}h(x)h(y)] = \mathbf{E}_{(x,y) \sim G_0}[\frac{1}{2} - \frac{1}{2}(h \circ g(x))(h \circ g(y))]$.

We next discretize G_1 in the manner of, say, Feige and Schechtman [50]. Choose an ϵ-net $\mathcal{N}$ within B_d of size at most $O(1/\epsilon)^d$. Further, partition B_d into Voronoi cells based on $\mathcal{N}$, with a disjoint cell C_v for each $v \in \mathcal{N}$. Now define the (finite) graph G_2 on $\mathcal{N}$ by taking $G_2(u, v) = G_1(C_u, C_v)$ (again, we identify a graph with its edge distribution). We claim

$$\text{Sdp}(G_2) \ge c - 3\epsilon. \tag{4.35}$$

To see this, recall that the identity embedding for G_1 achieves $\mathbf{E}_{(x,y) \sim G_1}[\frac{1}{2} - \frac{1}{2}x \cdot y] \ge c - \epsilon$. Now if x is in the cell C_u and y is in the cell C_y, then $x \cdot y = (u + \eta_1) \cdot (v + \eta_2)$ for some vectors η_1, η_2 of length at most ϵ; this implies $|x \cdot y - u \cdot v| \le 3\epsilon$. Since we can draw from G_2 by drawing $(x, y) \sim G_1$ and then taking (u, v) such that $x \in C_u$ and $y \in C_v$, we conclude that $\mathbf{E}_{(u,v) \sim G_2}[\frac{1}{2} - \frac{1}{2}u \cdot v] \ge c - \epsilon - \frac{3}{2}\epsilon$. We conclude that (4.35) holds with the identity map as the embedding. The fact that

$$\text{Opt}(G_2) \le s \tag{4.36}$$

follows for the same reason as (4.34) — any cut for G_2 can be extended to an equally good cut for G_1.

We now eliminate self-loops from G_2, forming G_3, using the construction in the Section of Khot and O'Donnell [102], which itself is based on a trick of Arora, Berger, Hazan, Kindler, and Safra [8]. It is shown therein that for any $\epsilon > 0$, we can take G_3 to have $O(1/\epsilon)^2$ times as many vertices as G_2, and satisfy

$$\mathrm{Sdp}(G_3) \geq \mathrm{Sdp}(G_2) \geq c - 3\epsilon, \tag{4.37}$$

and

$$\mathrm{Opt}(G_3) \leq \mathrm{Opt}(G_2) \leq s + \epsilon. \tag{4.38}$$

Finally, we form $G' = G_4$ from G_3, converting weighted edges to unweighted edges. There is a simple randomized way to do this (see, e.g., [19, 35]), taking a weighted graph on m vertices into an unweighted one on $\mathrm{poly}(m/\epsilon)$ vertices, such that

$$\mathrm{Sdp}(G_4) \geq \mathrm{Sdp}(G_3) - \epsilon \geq c - 4\epsilon, \tag{4.39}$$

and

$$\mathrm{Opt}(G_4) \leq \mathrm{Opt}(G_3) + \epsilon \leq s + 2\epsilon. \tag{4.40}$$

Since G_3 has $O(1/\epsilon)^{d+2}$ vertices, our G_4 has $n = (1/\epsilon)^{O(d)}$ vertices, as claimed. The proof follows after replacing ϵ by $\epsilon/4$. $\qquad\square$

4.13 RPR^2 — Implementation Issues

In this section we mention a few implementation issues that arise in the use of the RPR^2 framework and discuss how they affect our algorithmic guarantees. All of these issues have been considered before; see [46, 48, 50, 59, 112].

Exact Solving of the SDP. The SDP-solving guarantee one actually has is that a solution within ϵ of optimum can be found in time $\mathrm{poly}(n) \cdot \log(1/\epsilon)$. We have already treated this issue in the proof of Corollary 4.4.4. Another related issue is that the vectors returned by the SDP-solver may not lie precisely on the unit sphere, something we assumed in our analysis. This can be taken care of by shrinking all vectors slightly so that they lie within the unit ball, and then adding a fictitious extra coordinate with tiny values to make the vectors have length exactly 1.

Choosing Gaussian Random Variables. Again, this can not be done precisely, but the approximation methods of Mahajan and Ramesh [112] shows that one can occur ϵ loss at the expense only of $\mathrm{poly}(n, 1/\epsilon)$ time.

Expectation vs. High probability vs. Deterministic. Our results have been concerned with showing the expected value of the fractional cut produced by the (randomized) RPR^2 algorithm is at least $S(c)$. One can turn this into a high-probability result, losing only an additive ϵ in cut value, by using $\mathrm{poly}(n, 1/\epsilon)$ independent repetitions. Alternatively, one can derandomize the RPR^2 framework, again losing only an additive ϵ in the cut value, via the method of conditional expectations; this can be done in $\mathrm{poly}(n, 1/\epsilon)$ time [112] or

$O(n) \cdot 2^{\mathrm{poly}(1/\epsilon)}$ time [46]. Having done either of these, one has a fractional cut with value at least $S(c) - \epsilon$. This can be converted into a proper cut with at least the same value by the method of conditional expectations.

Multiple Rounding Functions. As discussed in section 4.1.3, we also want to try a *collection* $\mathscr{R}$ of rounding functions. For a high-probability results, we can simply repeat the algorithm $O(|\mathscr{R}|\log|\mathscr{R}|)$ times for each rounding function and this will achieve what the best of them does. Alternatively, we can just use the derandomized algorithms once for each $r \in \mathscr{R}$.

Proper Cuts when G Has Self-loops. Given a graph G with self-loops, we cannot actually find proper cuts with value at least $S(\mathrm{Sdp}(G))$. For example, if G consists of a single self-loop then $\mathrm{Sdp}(G) = \frac{1}{2}$ (via the embedding mapping the vertex to 0), but there is no proper cut of value $\frac{1}{2}$. The way to interpret our guarantee for graphs G with self-loops is as follows: First, remove the self-loops from G, forming G' — note that this does not change the value of the optimal proper cut. Then our algorithm achieves at least $S(\mathrm{Sdp}(G')) - \epsilon \geq S(O(G)) - \epsilon$, where $O(G)$ denotes the value of the optimal proper cut in G.

4.14 Improved Asymptotics of $S(\frac{1}{2} + \epsilon)$

As described in section 4.1.8, Charikar and Wirth [30] established $\mathrm{Gap}_{\mathrm{SDP}}(\frac{1}{2} + \epsilon) \geq \frac{1}{2} + \Omega(\epsilon/\ln(1/\epsilon))$ and Khot and O'Donnell [102] established $\mathrm{Gap}_{\mathrm{SDP}}(\frac{1}{2} + \epsilon) \leq \frac{1}{2} + O(\epsilon/\ln(1/\epsilon))$. In this Section we carefully examine these proofs and conclude the following:

Theorem 4.14.1. $\mathrm{Gap}_{\mathrm{SDP}}(\frac{1}{2} + \epsilon) = S(\frac{1}{2} + \epsilon) = \frac{1}{2} + (\frac{1}{2} \pm o(1)) \cdot \epsilon/\ln(1/\epsilon)$.

Proof. We upper-bound $S(c)$ essentially by repeating the argument in [102], paying more attention to the constants. Take P to be the $(1, \rho_0)$-distribution with weight $p = \frac{2}{3} + \frac{4}{3}\epsilon$ on $\rho_0 = -\frac{1}{2}$ and weight $\frac{1}{3} - \frac{4}{3}\epsilon$ on 1. Now if $r : \mathbb{R} \to [-1, 1]$ is any odd one-dimensional rounding function, we have

$$\mathrm{val}_{\mathscr{G}_P}(r) = \tfrac{1}{2} - \tfrac{1}{2}\left[(\tfrac{1}{3} - \tfrac{4}{3}\epsilon)\mathbb{S}_1(r) + (\tfrac{2}{3} + \tfrac{4}{3}\epsilon)\mathbb{S}_{-1/2}(r)\right] = \tfrac{1}{2} - \sum_{\mathrm{odd}\ s}\left(\tfrac{1}{6} - \tfrac{2\epsilon}{3} + (\tfrac{1}{3} + \tfrac{2\epsilon}{3})(-\tfrac{1}{2})^s\right)\widehat{r}(s)^2$$

$$\leq \tfrac{1}{2} + \epsilon\widehat{r}(1)^2 - \sum_{\mathrm{odd}\ s \geq 3}(\tfrac{1}{8} - \tfrac{3}{4}\epsilon)\widehat{r}(s)^2 = \tfrac{1}{2} + \epsilon\widehat{r}(1)^2 - (\tfrac{1}{8} - \tfrac{3}{4}\epsilon)\mathbf{E}[(r - Lr)^2], \quad (4.41)$$

where L denotes the 'projection to degree 1' operator; i.e., $Lr(x) = \widehat{r}(x)x$. As in [102] we consider the value of $\sigma^2 := \widehat{r}(1)^2 = \mathbf{E}[(Lr)^2]$, the variance of the Gaussian $Lr(x)$. Using $|r| \leq 1$, we lower-bound

$$\mathbf{E}[(r - Lr)^2] \geq \mathbf{E}[\mathbf{1}_{\{|Lr| \geq 1\}} \cdot (\mathrm{sgn}(r) - Lr)^2],$$

which asymptotically is $\sigma^{\Theta(1)} \cdot \exp(-1/2\sigma^2)$. If $\sigma \gg 1/\sqrt{2\ln(1/\epsilon)}$ then the final term in (4.41) will exceed ϵ, making the overall quantity less than $\frac{1}{2}$. Thus in upper-bounding (4.41) we can assume $\sigma \leq (1 + o(1))/\sqrt{2\ln(1/\epsilon)}$, and thus we get an upper bound of $\frac{1}{2} + (\frac{1}{2} + o(1))\epsilon/\ln(1/\epsilon)$,

c	$S(c)$	c	$S(c)$	c	$S(c)$	c	$S(c)$
0.505	0.5008	0.590	0.5414	0.675	0.6012	0.760	0.6694
0.510	0.5021	0.595	0.5446	0.680	0.6050	0.765	0.6736
0.515	0.5036	0.600	0.5478	0.685	0.6089	0.770	0.6778
0.520	0.5053	0.605	0.5510	0.690	0.6127	0.775	0.6820
0.525	0.5072	0.610	0.5544	0.695	0.6167	0.780	0.6862
0.530	0.5092	0.615	0.5577	0.700	0.6206	0.785	0.6905
0.535	0.5113	0.620	0.5611	0.705	0.6245	0.790	0.6947
0.540	0.5136	0.625	0.5646	0.710	0.6285	0.795	0.6990
0.545	0.5160	0.630	0.5681	0.715	0.6325	0.800	0.7033
0.550	0.5185	0.635	0.5716	0.720	0.6365	0.805	0.7076
0.555	0.5211	0.640	0.5752	0.725	0.6406	0.810	0.7119
0.560	0.5238	0.645	0.5788	0.730	0.6446	0.815	0.7162
0.565	0.5265	0.650	0.5825	0.735	0.6487	0.820	0.7206
0.570	0.5294	0.655	0.5861	0.740	0.6528	0.825	0.7249
0.575	0.5323	0.660	0.5898	0.745	0.6569	0.830	0.7293
0.580	0.5352	0.665	0.5936	0.750	0.6611	0.835	0.7336
0.585	0.5383	0.670	0.5974	0.755	0.6652	0.840	0.7380

Table 4.1: Value of $S(c)$

as claimed.

To lower-bound $\mathrm{Gap}_{\mathrm{SDP}}(\frac{1}{2} + \epsilon)$ we refer to [30, equation (11)], which shows that

$$\mathrm{Gap}_{\mathrm{SDP}}(\tfrac{1}{2} + \epsilon) \geq \frac{1}{2} + \frac{\epsilon}{T^2} - 4e^{-T^2/2}$$

for every $T \geq 1$. By taking $T = (1 - o(1)) \cdot \sqrt{2\ln(1/\epsilon)}$, we get a lower bound of $\frac{1}{2} + (\frac{1}{2} - o(1)) \cdot \epsilon/\ln(1/\epsilon)$. □

4.15 Approximate Values of $S(c)$

is alsso equivalent to the following graph problem. Given an undirected graph
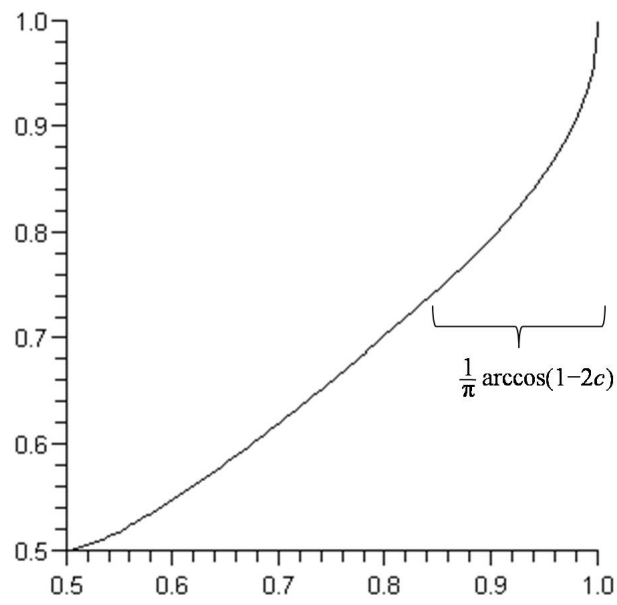
Figure 4.2: $S(c)$ vs. c.

Chapter 5

Conditional Hardness of Approximating Satisfiable 3-CSPs

5.1 Introduction

In this chapter, we study the approximability of 3-CSP; in particular we are interested in instances of 3-CSP that are satisfiable (i.e., instance with optimum value being 1). We study the largest s such that we can still $(1, s)$-approximate MAX 3-CSPs in polynomial time. In addition to the importance of the problem by itself, it has a strong motivation in the study of Probability Checkable Proof.

5.1.1 The PCP Characterization of NP

The famous PCP (Probabilistic Checkable Proof) Theorem states that any language in NP has a proof system where the proofs can be probabilistically checked in a query-efficient way. The notation $\text{PCP}_{c,s}(r(n), q(n))$ stands for the class of languages M verifiable by a proof system with the following parameters: for an input x of length n, the verifier uses $r(n)$ random bits and query $q(n)$ bits in the proof to decide in polynomial time whether x is in M or not. The verifier has the following performance guarantees: i) if x is in M, there exists a proof that passes with probability c and ii) if x is not in the M, no proof passes with probability more than s. We call c the *completeness* and s the *soundness* of the verifier.

If the verifier makes all its queries at one time based only on x and the $r(n)$ random bits, it is called *nonadaptive*. On the other hand, if the verifier picks next query's location based on x, the random bits and all the previous queries, it is called *adaptive*. The notation aPCP and naPCP is used to distinguish languages verifiable by the adaptive and the nonadaptive verifier. The adaptive verifier has better performance while the nonadaptive verifier has more natural implication to the hardness of approximation for CSPs (See Theorem 5.1.3 for more discussions). We mainly focus on the nonadaptive proof system in this work.

Formally, the PCP Theorem [9, 10] states that:

Theorem 5.1.1. $\text{NP} \subseteq \text{naPCP}_{1,1/2}(O(\log n, O(1))$.

We can see that c is 1 in the PCP Theorem; i.e., when the input x is in the language, there exists a proof that passes with probability 1. Such a verifier is said to have perfect completeness, which is a natural and desirable property of the proof system. Much effort is devoted to optimizing the tradeoff between $q(n)$ and s (as well as some other parameters such as proof length, adaptativity, free bit complexity) [19, 66, 76, 130]. It is known that in order to make c to be 1 and bound s away from 1, the minimum number of queries that the verifier need to make is 3. The subject study in this work is then to optimize the soundness s for the 3-query nonadaptive PCP systems with perfect completeness. Formally, we examine the following question:

Question 5.1.2. *What is the smallest s that makes* $\text{NP} \subseteq \text{naPCP}_{1,s}[O(\log n), 3]$

This problem was first studied in [19] where Bellare, Goldreich and Sudan showed that $\text{NP} \subseteq \text{naPCP}_{1,0.8999+\epsilon}[O(\log n), 3]$. Håstad [76] further improved this result to $\text{NP} \subseteq \text{naPCP}_{1,3/4+\epsilon}(O(\log(n)), 3)$. Around the same time, Zwick [141] showed that $\text{naPCP}_{1,5/8}(O(\log(n)), 3) \subseteq \text{BPP}$ by giving a randomized polynomial-time 5/8-approximation algorithm for satisfiable 3CSP. Therefore unless $\text{NP} \subseteq \text{BPP}$, the best s must be bigger than 5/8. Zwick further conjectured that this algorithm is optimal, i.e., $\text{NP} \subseteq \text{naPCP}_{1,5/8+\epsilon}(O(\log n), 3)$ (See Section 5.1.2

for more discussions). After nearly a decade, Khot and Saket [104] showed that $NP \subseteq naPCP_{1,20/27+\epsilon}(O(\log(n)), 3)$.

We note that certain relaxations of the problem are well understood. If we allow the verifier to be adaptive, Guruswami et al. [66] proved that $NP \subseteq aPCP_{1,1/2+\epsilon}(O(\log(n)), 3)$. If we allow an arbitrarily small loss of completeness for the nonadaptive verifier, Håstad [76] showed that $NP \subseteq naPCP_{1-\epsilon,1/2+\epsilon}(O(\log(n)), 3)$. Both of above results achieved optimal soundness assuming $NP \subseteq BPP$ [141].

We think that Question 5.1.2 addresses an important missing part in understanding the 3-query PCP systems. In addition, as is mentioned the answer to this question is equivalent to deciding the optimal hardness of approximation ratio for satisfiable 3-CSPs.

5.1.2 Hardness of Approximation and Khot's Conjectures

The relationship between PCP and MAX 3-CSP is due to the following well known connection between PCP and hardness of approximation:

Theorem 5.1.3. *Let Φ be a set of predicates with arity no more than k. Following two statements are equivalent: i) MAX Φ (c, s) is NP-hard . ii)For some NP complete language L, there is a PCP system with nonadaptive verifier using predicates only from Φ has completeness c and soundness s.* [1]

Note that the *nonadaptiveness* is crucial in Theorem 5.1.3. If the verifier is adaptive in above theorem, the hardness result would hold only for CSPs with predicate set: "depth k decision tree with predicate in Φ at each node".

As a direct application of the theorem, we have that Question 5.1.2 is equivalent to the following question:

Question 5.1.4. *What is the smallest s such that MAX 3-CSP $(1, s)$ is NP-hard .*

We can also see why unless $NP \subseteq BPP$, Zwick's 5/8-approximation randomized algorithm for satisfiable MAX 3-CSP [141] mentioned earlier implies that the smallest s in Question 5.1.2 and 5.1.4 must be bigger than 5/8. And Zwick's conjecture is that $s = 5/8 + \epsilon$ in both questions 5.1.2 and 5.1.4.

5.1.3 Satisfiable Max NTW

The optimal hardness of satisfiable 3CSP is also corresponding to a open problem in the conclusion of the seminal paper of Håstad [76]. The problem he asked is to decide for *satisfiable* Max-NTW whether there exists an polynomial-time approximation algorithm beyond the random assignment threshold 5/8. Following is the formal statement of Håstad's open problem:

Question 5.1.5. *For any constant $\epsilon > 0$, given a satisfiable Max-NTW instance $\mathscr{I}$, is it NP-hard to find an assignment that satisfies more than an $5/8 + \epsilon$ fraction of the constraints?*

Question 5.1.5 is important as a "Yes" answer to it will also resolve Question 5.1.2 and 5.1.4 since Max-NTW is a special Max-3CSP.

---

[1]Strictly speaking, the hardness result in Theorem 5.1.3 is only for weighted MAX k-CSPs. As for Satisfiable Max k-CSPs, the inapproximability is the same for weighted and unweighted instances due to the reduction in [35].

As a result of Theorem 5.1.3, Question 5.1.5 is equivalent to decide whether there is such a nonadaptive PCP system for some NP Complete problem that the verifier has perfect completeness and soundness $5/8 + \epsilon$ and it uses the same predicate set as Max-NTW. Constructing such a PCP system for the d-to-1 LABEL-COVER is the main focus of the remaining work.

5.2 Our Contribution and Methods

5.2.1 Main Results

Our main result is that we can solve Håstad's Open Problem (Question 5.1.5) assuming the Khot's d-to-1 Conjecture hold for any finite d. Formally, our main theorem can be stated as follows:

Theorem 5.2.1. *Assuming Khot's d-to-1 Conjecture holds for any finite positive integer d,* MAX NTW $(1, 5/8 + \epsilon)$ *is* NP *hard. Equivalently speaking, there is a 3-query PCP system for* NP *that has perfect completeness and soundness $5/8 + \epsilon$ for any $\epsilon > 0$. In addition, the verifier is nonadaptive and uses the same predicates as* MAX NTW.

Above theorem implies the answer to Question 5.1.2 and 5.1.4 and confirms Zwick's conjecture.

Corollary 5.2.2. *Assuming Khot's d-to-1 Conjecture holds for any finite positive integer d and $\epsilon > 0$,* MAX 3-CSP $(1, 5/8 + \epsilon)$ *is* NP *hard. Equivalently speaking,* NP $\subseteq$ naPCP$_{1, 5/8 + \epsilon}(O(\log(n)), 3)$. *Further assuming that* NP $\not\subseteq$ BPP, *Zwick's $5/8$-approximation algorithm for satisfiable 3-CSPs is optimal and $5/8 + \epsilon$ is the optimal s for both Question 5.1.2 and 5.1.4.*

5.2.2 Methods

In the proof we build a PCP system for the d-to-1 LABEL-COVER that reads three bits and checks the NTW predicate on the literals of them.

The main technicalities of this work are i) designing the verifier for the d-to-1 LABEL-COVER; ii) analyzing the soundness of the proof system.

Our verifier can be viewed as a generalization of the 3-query dictator test in [121]. The dictator test in [121] generates queries from the sample space $\{-1, 1\}^n \times \{-1, 1\}^n \times \{-1, 1\}^n$ for testing "one function". For the use of d-to-1 LABEL-COVER, roughly speaking we need the verifier to address query space $\{-1, 1\}^n \times \{-1, 1\}^{dn} \times \{-1, 1\}^{dn}$ for testing "two functions".

In the analysis of the PCP system, the main challenge (as usual) is to bound the expectation of certain quadratic and cubic term. The problem is more complicated compared with [121] and some very different techniques are used in the work. We analyze the quadratic term in the resulting Fourier analysis based on some novel arguments about the positivity of certain linear operators. For the cubic term, we use the Invariance Principle style arguments similar to [115, 116]. However, none of these invariance theorems in [115, 116] can be applied to our proof as a black box since our distribution does not have 2-wise independence. In addition, unlike the other proofs using the Invariance Principle that are usually dependent on some properties in *Gaussian space*, we prove some invariant properties between two distributions over *Boolean cube*. These two distributions have the

same (non-zero) 2-wise correlation while Invariance Principle helps to handle the hard-to-analyze 3-wise correlation.

5.2.3 Related and Subsequent Work

Related Work Most of the hardness reduction from UNIQUE-GAMES to Max-Φ problems involves designing a "dictator test" that only uses predicate from Φ. In [121], we proposed a 3-query dictator test based on the NTW predicate with soundness $5/8 + \epsilon$ and completeness 1. However, the test can only be directly used to build PCP system for the Unique Label Cover and such a proof system therefore does not have perfect completeness.

Subsequent Work After our work, there have been several results built on top of the techniques developed in this work. In [136], the authors studied the problem of k-query PCP with perfect completeness. Their main contribution is a k-query Dictator test with perfect completeness. While it remains a open question how to compose their Dictator Test with proper outer verifier. In another recent work [137], the author investigated the 3-query PCP system over $\mathbb{Z}_q$ with perfect completeness and obtained improved soundness assuming the d-to-1 conjecture.

5.2.4 PCP System Framework

The high level framework of our PCP system is similar to Håstad's construction for Max 3Lin [76]. Given is a d-to-1 LABEL-COVER instance $\mathscr{L} : (U, V, E, P, R_1, R_2, \Pi)$. A "proof" for $\mathscr{L}$ consists of a collection of the truth table of Boolean functions for each vertex. More specifically, for each vertex $u \in U$, there is an associated Boolean function $f_u(x) : \{-1, 1\}^{R_1} \to \{-1, 1\}$ and for each vertex $v \in V$, there is an associated Boolean function $g_v(y) : \{-1, 1\}^{R_2} \to \{-1, 1\}$. The proof contains all the truth table of these Boolean functions and the length of the proof is therefore $|U|2^{R_1} + |V|2^{R_2}$. From now on we always view -1 as True and 1 as False.

The verifier checks the proof by following procedure:

- Pick an edge $e = (u, v)$ from distribution P.
- Generate a triple (x, y, z) from some distribution $\mathscr{T}_e$ on $\{-1, 1\}^{R_1} \times \{-1, 1\}^{R_2} \times \{-1, 1\}^{R_2}$ ($\mathscr{T}_e$ is specified later).
- Accept if $\text{NTW}(f_u(x), g_v(y)), g_v(z))) = 1$.

Folding The prover can write the constant "1" function for every f_u, g_v and such a proof always passes. To address this, the standard "folding trick"[19] is used for our system. For example, for query $x = (x_1, x_2, ..x_{R_1})$ on f_u, the verifier always use the value of $\text{sgn}(x_1)f_u(1, sgn(x_1)x_2, ..sgn(x_1)x_n)$ (instead of $f_u(x)$). Similar strategy is applied for query y and z on g_v. Suffice to say, we can assume that all the functions f_u, g_v are odd.

For above PCP system, we will show:

1. If $opt(\mathscr{L}) = 1$, there is a proof that passes the test with probability 1. (completeness)

2. For any $\epsilon > 0$, if there exists a proof that passes with probability at least $5/8 + \epsilon$, then $opt(\mathscr{L}) > \eta$, where $\eta > 0$ is some constant only depend on ϵ and d. (soundness)

Assuming the d-to-1 Conjecture, such a proof system shows that $\text{NP} \subseteq \text{naPCP}_{1,5/8+\epsilon}(O(\log n, 3))$ and a $5/8 + \epsilon$ hardness result for approximating satisfiable Max 3-NTW.

5.3 The Test and the Analysis

Given above PCP reduction framework, the remaining of the work constructs the distribution $\mathcal{T}_e$ and analyzes the completeness and soundness for the associated verifier. The reader is assumed to be familiar with the basics of Fourier analysis of Boolean functions.; see e.g., [134]. As a reminder, our default representation for bits will be $+1$ and -1 rather than 0 and 1. We use $\hat{f}(S)$ to denote the Fourier coefficients of function f on set S.

5.3.1 Idea of Constructing $\mathcal{T}_e$

Recall that the verifier first picks an edge $e = (u, v)$. Then it generates (x, y, z) by distribution $\mathcal{T}_e$ and accept if $\text{NTW}(f_u(x), g_v(y), g_v(z)) = 1$. This section we define the distribution $\mathcal{T}_e$.

For the picked edge e, we write $d_i = |\pi_e^{-1}(i)|$ for $i = 1, 2...R_1$. By the property of d-to-1 projection, we know $1 \le d_i \le d$.

We express $f_u : \{-1, 1\}^{R_1} \to \{-1, 1\}$ as

$$f_u : \mathcal{X}^1 \times \mathcal{X}^2 \times \cdots \times \mathcal{X}^{R_1} \to \{-1, 1\},$$

where each $\mathcal{X}^i = \{-1, 1\}^{\{i\}}$ (a slightly complicated way to write $\{-1, 1\}$), and $g_v : \{-1, 1\}^{R_2} \to \{-1, 1\}$ as

$$g_v : \mathcal{Y}^1 \times \mathcal{Y}^2 \times \cdots \times \mathcal{Y}^{R_1} \to \{-1, 1\},$$

where each $\mathcal{Y}^i = \{-1, 1\}^{\pi_e^{-1}(i)}$. Let we also write $\mathcal{Z}^i = \{-1, 1\}^{\pi_e^{-1}(i)}$.

We construct $\mathcal{T}_e$ as a product distribution: $(\prod_{i=1}^R \mathcal{X}^i \times \mathcal{Y}^i \times \mathcal{Z}^i, \mathcal{T}_e) = \prod_{i=1}^R (\mathcal{X}^i \times \mathcal{Y}_i \times \mathcal{Z}_i, \mathcal{T}_e^i)$. To define each $\mathcal{T}_e^i$, we start by defining some general distributions on $\{-1, 1\} \times \{-1, 1\}^m \times \{-1, 1\}^m$. We denote $\{-1, 1\} \times \{-1, 1\}^m \times \{-1, 1\}^m$ by $\mathcal{X} \times \mathcal{Y} \times \mathcal{Z}$ here.

Definition 5.3.1. *Define distribution $\mathcal{H}(m)$ generating $(x, y_1, y_2, ..y_m, z_1, z_2, ..z_m)$ as follows: first $x, y_1, y_2, ..y_m$ are generated as independent random bits and then for each $1 \le i \le m$, z_i is set to be $-y_i x$.*

By definition, x and $y_1, ...y_m$ are independent. In addition, the distribution is the same if we first generate $(x, z_1, ...z_m)$ and then set $y_i = -x_i z_i$. Therefore x and $z_1, ..z_m$ are also independent.

The distribution $\mathcal{H}(m)$ is also the basis of Håstad's construction of the XOR_3 verifier. Håstad's verifier checked $\text{XOR}_3(f_u(x), g_v(y), g_v(z)) = 0$ where (x, y, z) is first generated by $\prod \mathcal{T}_e^i$ for $\mathcal{T}_e^i = \mathcal{H}_\delta(d_i)$ and then each bit in (x, y, z) is reset to be some random bit independently with probability δ. Such a PCP system has near perfect completeness and soundness $1/2 + \delta$. The random noise is added to each bit to make sure that big parity function passes with small probability.

We have a similar situation here. $\mathcal{H}(m)$ is good for us as (x, y_i, z_i) can only be one of $(1, 1, -1), (1, -1, 1), (-1, 1, 1), (-1, -1, -1)$ and these tripes are all in the support of NTW. We can not add random noise to $\mathcal{H}(m)$ though as we need the perfect completeness. Notice

that $(1,1,1)$ is also in the support of NTW. We then make a tweak on $\mathcal{H}(m)$ by including $(1,1,1)$ as a possible value for (x, y_i, z_i).

Definition 5.3.2. *Define distribution $\mathcal{N}(m)$ generating $(x, y_1, y_2, ..y_m, z_1, z_2, ..z_m)$ as follows. First, we pick a random integer k from 1 to m. Next, we generated $x, y_1, y_2, ..y_{k-1}, y_{k+1}, ..y_m$ as independent random bits. Last, we set y_k, z_k to be equal to x and for any $i \in [m], i \neq k$, we set z_i to be equal to $-y_i x$. Define distribution $\mathcal{H}_\delta(m) = (1 - \delta)\mathcal{H}(m) + \delta \mathcal{N}(m)$; i.e., $\mathcal{H}_\delta(m)$ generates $(x, y_1, y_2, ..y_m, z_1, ..z_m)$ by $\mathcal{H}(m)$ with probability $1 - \delta$ and by $\mathcal{N}(m)$ with probability δ.*

It is easy to check that the margin distribution of $\mathcal{H}(m)$, $\mathcal{N}(m)$ and $\mathcal{H}_\delta(m)$ on $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z}$ are all uniform.

We are now ready to define the $\mathcal{T}_e$:

Definition of $\mathcal{T}_e$

Definition 5.3.3. *We have $\mathcal{T}_e = \prod \mathcal{T}_e^i$ where each $\mathcal{T}_e^i$ is set to be $\mathcal{H}_\delta(d_i)$ for $i = 1, 2..R_1$ and $\delta = \epsilon^2/2$.*

To analyze $\mathcal{T}_e$, we also need to define serval other distributions.

Definition 5.3.4. *Define distribution $\mathcal{F}(m)$ generating $(x, y_1, y_2, ..y_m, z_1, z_2, ..z_m)$ as follows: first $y_1, y_2..y_m, z_1, z_2..z_m$ are generated by their marginal distribution on $\mathcal{H}(m)$ and x is generated as a random bit independent with $(y_1, y_2, ..y_m, z_1, ..z_m)$.*

By definition, $\mathcal{F}(m)$ and $\mathcal{H}(m)$ have the same marginal distribution on $\mathcal{Y} \times \mathcal{Z}$. Also, they have the same marginal distribution on $\mathcal{X} \times \mathcal{Y}$ and $\mathcal{X} \times \mathcal{Z}$ as $x, y_1, ..y_m$ and $x, z_1, ..z_m$ are independent in both $\mathcal{F}(m)$ and $\mathcal{H}(m)$. In addition, it is easy to check $\mathcal{H}(m)$ and $\mathcal{F}(m)$ have the same "1-wise" marginal distribution (the uniform distribution) on $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$.

We also add the "tweak" $\mathcal{N}(m)$ to $\mathcal{F}(m)$ to define a new distribution:

Definition 5.3.5. *Define distribution $\mathcal{F}_\delta$ to be $\mathcal{F}_\delta(m) = (1 - \delta)\mathcal{F}(m) + \delta \mathcal{N}(m)$.*

It is easy to see $\mathcal{H}_\delta(m)$ and $\mathcal{F}_\delta(m)$ also have the same "1-wise" and "2-wise" correlation; i.e., $\mathcal{H}_\delta(m)$ and $\mathcal{F}_\delta(m)$ have the same marginal distribution on $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$ $\mathcal{X} \times \mathcal{Y}$, $\mathcal{X} \times \mathcal{Z}$ and $\mathcal{Y} \times \mathcal{Z}$. Their "1-wise" marginal distributions are all uniform. The 3-wise correlation of $\mathcal{H}_\delta(m)$ and $\mathcal{F}_\delta(m)$ are different though. Following lemma describes the difference.

Lemma 5.3.6. *For any function $f : \mathcal{X} \to \mathbb{R}$, $g : \mathcal{Y} \to \mathbb{R}$, $\mathcal{Z} \to \mathbb{R}$,*

$$\mathop{\mathbf{E}}_{\mathcal{H}_\delta(m)}[f(x)g(y)h(z)] - \mathop{\mathbf{E}}_{\mathcal{F}_\delta(m)}[f(x)g(y)h(z)] = \sum_{|S| is\ odd, S \subseteq [m]} (1 - \delta)\hat{f}(\{1\})\hat{g}(S)\hat{h}(S)).$$

Proof. Recall that χ_S is defined to be the parity function on set S. By writing each function into its Fourier expansion, we have

$$RHS = \sum_{U \subseteq [1], V \subseteq [m], W \subseteq [m]} \hat{f}(U)\hat{g}(V)\hat{h}(W)(\mathop{\mathbf{E}}_{\mathcal{H}_\delta(m)}[\chi_U(x)\chi_V(y)\chi_W(z)] - \mathop{\mathbf{E}}_{\mathcal{F}_\delta(m)}[\chi_U(x)\chi_V(y)\chi_W(z)]).$$

(5.1)

By the definition of $\mathcal{H}_\delta(m)$ and $\mathcal{F}_\delta(m)$, we know

$$(\mathop{\mathbf{E}}_{\mathcal{H}_\delta(m)}[\chi_U(x)\chi_V(y)\chi_W(z)] - \mathop{\mathbf{E}}_{\mathcal{F}_\delta(m)}[\chi_U(x)\chi_V(y)\chi_W(z)])$$

$$= (1 - \delta)(\mathop{\mathbf{E}}_{\mathcal{H}(m)}[\chi_U(x)\chi_V(y)\chi_W(z)] - \mathop{\mathbf{E}}_{\mathcal{F}(m)}[\chi_U(x)\chi_V(y)\chi_W(z)]). \quad (5.2)$$

Notice that $\mathcal{H}(m)$ and $\mathcal{F}(m)$ have the same margin distribution on $\mathcal{Y} \times \mathcal{Z}$. Therefore to make (5.2) nonzero, U must be nonempty (and therefore must be $\{1\}$). When

111

$U = \{1\}$ we have that $\mathbf{E}_{\mathscr{F}(m)}[\chi_U(x)\chi_V(y)\chi_W(z)] = \mathbf{E}_{\mathscr{F}(m)}[x]\mathbf{E}_{\mathscr{F}(m)}[\chi_V(y)\chi_W(z)] = 0$. Therefore, $\mathbf{E}_{\mathscr{H}(m)}[x\chi_V(y)\chi_W(z)]$ must be nonzero to make (5.2) nonzero. It is not hard to see this happens only when V, W are the same set with odd cardinality and the expectation of $x\chi_V(y)\chi_W(z)$ is 1 . Therefore,

$$(5.1) = \sum_{|S|\text{is odd},S\subseteq[m]} (1-\delta)\hat{f}(\{1\})\hat{g}(S)\hat{h}(S) \underset{\mathscr{H}(m)}{\mathbf{E}}[x\chi_S(y)\chi_S(z)] = RHS.$$

$\square$

Recall the definition of *Correlation* between probability spaces.

Definition 5.3.7. *Let $(\Omega\times\Theta,\mu)$ be correlated finite probability space. Define the correlation between Ω and Θ to be*

$$\rho(\Omega,\Theta;\mu) = sup\{\mathbf{Cov}[f,g] : f\in A, g\in B, \mathbf{Var}[f] = \mathbf{Var}[g] = 1\}.$$

The conditional operator U_μ associated with μ is a mapping from function space $\{f|f:\Theta\to\mathbb{R}\}$ to $\{g|g:\Omega\to\mathbb{R}\}$ defined as follows: for $f:\Theta\to\mathbb{R}$ and any $x\in\Omega$ and random variable (X,Y) drawn from μ, $U_\mu f(x) = \mathbf{E}_Y[f(Y)|X = x]$.

Since $Cov(f,g) = \mathbf{E}[(f - \mathbf{E}[f])(g - \mathbf{E}[g])]$, we can assume without loss of generality that $\mathbf{E}[f] = 0, \mathbf{E}[g] = 0$ when calculating the correlation between the two spaces; i.e.,

$$\rho(\Omega,\Theta;\mathrm{P}) = sup\{\mathbf{E}[fg] : f\in A, g\in B, \mathbf{E}[f] = 0, \mathbf{E}[g] = 0, \mathbf{Var}[f] = \mathbf{Var}[g] = 1\}.$$

For the distributions defined, we have the following properties with proof in Section 5.5.

Lemma 5.3.8. $\rho(\mathscr{X}\times\mathscr{Y},\mathscr{Z};\mathscr{H}_\delta(m)) \le 1 - \frac{\delta^2}{2^{2d+1}d^2}$.

Lemma 5.3.9. $\rho(\mathscr{X},\mathscr{Y};\mathscr{H}_\delta(m)) \le \delta$.

Lemma 5.3.10. $\rho(\mathscr{X},\mathscr{Y}\times\mathscr{Z};\mathscr{F}_\delta(m)) \le \sqrt{\delta}$.

We comment that if we did not add the "tweak distribution" $\mathscr{N}$ to $\mathscr{H}$, we would have that $\rho(\mathscr{X}\times\mathscr{Y},\mathscr{Z};\mathscr{H}(m)) = 1$. As for $\mathscr{H}_\delta$, we still have $\rho(\mathscr{X},\mathscr{Y}\times\mathscr{Z};\mathscr{H}_\delta(m)) = 1$ and this is some tricky part for our analysis.

In [115] (Proposition 2.13), Mossel proved that the correlation of product spaces is decided by the maximum correlation among all the individual correlated spaces:

Proposition 5.3.11. *For i = 1,2..n, let $(\Omega_i\times\Theta_i,\mu_i)$ be a finite probability space. We define product probability space $(\Omega\times\Theta,\mu) = \prod_{i=1}^n(\Omega_i,\Theta_i,\mu_i)$. Then:*

$$\rho(\Omega,\Theta;\mu) = \max_i \rho(\Omega_i,\Theta_i;\mu_i).$$

By applying Proposition 5.3.11, we know:

Lemma 5.3.12.

$$\rho(\prod_{i=1}^{R_1}\mathscr{X}^i\times\mathscr{Y}^i, \prod_{i=1}^{R_1}\mathscr{Z}^i;\mathscr{T}_e) \le 1 - \frac{\delta^2}{2^{2d+1}d^2}$$

$$\rho(\prod_{i=1}^{R_1}\mathscr{X}^i, \prod_{i=1}^{R_1}\mathscr{Y}^i;\mathscr{T}_e) \le \delta$$

$$\rho(\prod_{i=1}^{R_1}\mathscr{X}^i, \prod_{i=1}^{R_1}\mathscr{Y}^i\times\mathscr{Z}^i; \prod_{i=1}^{R_1}\mathscr{F}_\delta(d_i)) \le \sqrt{\delta}$$

112

5.3.2 Analysis of the Verifier

In this section, we analyze the completeness and soundness of our verifier.

Completeness Analysis If $val(L) = 1$ for some labelling L, we can simply take f_u ($u \in U$) to be the $L(u)$th dictator function $\chi_{L(u)}$ and g_v ($v \in V$) to be $\chi_{L(v)}$. By the definition of $\mathcal{T}_e$, for any edge (u,v), we know $(x_{L(u)}, y_{L(v)}, z_{L(v)})$ is always in the support of NTW. Also the dictator function is odd and it does not change by the folding procedure. Such a proof always passes with probability 1.

Soundness Analysis For any ϵ, we show if some proof passes the test with probability more than $5/8 + \epsilon$, then we have $opt(\mathcal{L}) > \eta$. where $\eta > 0$ is some constant depending only on ϵ and d.

First let us we arithmetize the probability the proof passes. We have

$$
\mathbf{Pr}(\mathtt{NTW}(f_u(x), g_v(y), g_v(z)) = 1) =
$$
$$
\mathop{\mathbf{E}}_{e=(u,v)\sim P, \mathcal{T}_e} [\tfrac{5}{8} + \tfrac{1}{8}(f_u(x) + g_v(y) + g_v(z)) + \tfrac{1}{8}(f_u(x)g_v(y) + g_v(y)g_v(z) + f_u(x)g_v(z)) - \tfrac{3}{8}f_u(x)g_v(y)g_v(z)].
$$
$$(5.3)$$

By the folding mentioned in Section 5.2.4 ,we know all the f_u, g_v are odd. Also notice that $\mathcal{T}_e$'s 1-wise marginal distribution are all uniform, therefore

$$
\mathop{\mathbf{E}}_{e=(u,v)\sim P, \mathcal{T}_e} [\tfrac{1}{8}(f_u(x) + g_v(y) + g_v(z))] = 0.
$$

In the following Theorem 5.3.13, 5.3.14, 5.3.26, we analyze the terms $\mathbf{E}[f_u(x)g_v(y) + f_u(x)g_v(z)]$, $\mathbf{E}[g_v(y)g_v(z)]$ and $\mathbf{E}[f_u(x)g_v(y)g_v(z)]$ respectively.

Theorem 5.3.13. *For any odd Boolean functions $f : \{-1,1\}^{R_1} \to \{-1,1\}$ and $g : \{-1,1\}^{R_2} \to \{-1,1\}$,*

$$
\mathop{\mathbf{E}}_{x,y\sim \mathcal{T}_e} [f(x)g(y)] \le \delta.
$$

Proof. This follows directly from the Lemma 5.3.12 that $\rho(\prod_{i=1}^{R_1} \mathcal{X}^i, \prod_{i=1}^{R_2} \mathcal{Y}_i, \mathcal{T}_e) \le \delta$. By definition for any odd Boolean function f, g (therefore with mean 0 and variance 1), they can have correlation at most δ. $\qquad \square$

By applying Theorem 5.3.13 and notice the symmetry between y and z, we have

$$
\mathop{\mathbf{E}}_{e=(u,v)\sim, \mathcal{T}_e} [\frac{1}{8}(f_u(x)g_v(y) + f_u(x)g_v(z))] \le \delta/4.
$$

In the next section we analyze the term $g_v(y)g_v(z)$.

Analyzing $g_v(y)g_v(z)$

Theorem 5.3.14. *For any odd Boolean function $g: \{-1,1\}^{R_2} \to \{-1,1\}$, we have*

$$\mathop{\mathbf{E}}_{y,z \sim \mathcal{T}_e}[g(y)g(z)] \leq \delta.$$

Proof. It can be be checked that $\rho(\mathcal{Y}_i, \mathcal{Z}_i, \mathcal{H}_\delta(d_i)) = 1 - \delta$. We can not use the same simple trick as in Theorem 5.3.13. However, the fact that g is *odd* makes it possible for us to bound $\mathbf{E}_{y,z \sim \mathcal{T}_e}[g(y)g(z)]$.

First we need define the matrix form of the distribution on $\{-1,1\}^m \times \{-1,1\}^m$ with the Fourier basis.

Definition 5.3.15. *Suppose $\mathcal{P}$ is a distribution on $\{-1,1\}^m \times \{-1,1\}^m$. A $2^m \times 2^m$ matrix $M(\mathcal{P})$ is defined as follows: let us use all the subsets of $[m]$ to index number from 1 to 2^m. The $M(\mathcal{P})$ has the following form. For any $S \subseteq [m], T \subseteq [m]$, the element $M(\mathcal{P})_{S,T}$ at S row T column is $\mathbf{E}_{(y,z) \sim \mathcal{P}}[\chi_S(y)\chi_T(z)]$.*

We can also identify function $g: \{-1,1\}^m \to \mathbb{R}$ with a row vector in $\mathbb{R}^{2^m}$ that contains the entire collection of g's Fourier coefficients. The Fourier coefficients are arranged in the same order as their associated sets in Definition 5.3.15

For example, when $m = 2$, the subsets of $\{1,2\}$ are of the order $\emptyset, \{1\}, \{2\}, \{1,2\}$. If the distribution generates $(y_1, y_2, z_1, z_2) \in \{-1,1\}^2 \times \{-1,1\}^2$ as follows: y_1, y_2 is generated as independent random bits and $z_i = y_i$ for $i = 1,2$. Then such a distribution has the following matrix form:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

And we write down function g's vector form as:

$$g = \begin{bmatrix} \hat{g}(\phi) \\ \hat{g}(\{1\}) \\ \hat{g}(\{2\}) \\ \hat{g}(\{1,2\}) \end{bmatrix}.$$

With the new matrix notion, we can write the product of two functions as the multiplication between vectors and matrix:

$$\mathop{\mathbf{E}}_{y,z \sim \mathcal{P}}[f(y)g(z)] = \sum_{T,S \subseteq [m]} \hat{f}(S)\hat{g}(T)\mathop{\mathbf{E}}_{\mathcal{P}}[\chi_S(y)\chi_T(z)] = f^T M(\mathcal{P})g.$$

Following lemma is easy to check.

Lemma 5.3.16. *i) $\mathcal{P}_1$ and $\mathcal{P}_2$ are two distributions on $\{-1,1\}^m \times \{-1,1\}^m$. If $\mathcal{P} = c\mathcal{P}_1 + (1-c)\mathcal{P}_2$, then $M(\mathcal{P}) = cM(\mathcal{P}_1) + (1-c)M(\mathcal{P}_2)$.*

ii) $\mathcal{P}_1$ and $\mathcal{P}_2$ are two distributions on $\{-1,1\}^{m_1} \times \{-1,1\}^{m_1}$ and $\{-1,1\}^{m_2} \times \{-1,1\}^{m_2}$. If $\mathcal{P} = \mathcal{P}_1 \times \mathcal{P}_2$, then $M(\mathcal{P}) = M(\mathcal{P}_1) \otimes M(\mathcal{P}_2)$.

We define the identity distribution $\mathscr{I}(m)$ on $\{-1,1\}^m \times \{-1,1\}^m$ as follows: $y_1,..y_m$ are first generated as independent random bits and $z_i = y_i$ for every i. It is easy to check that $M(\mathscr{I}(m))$ is the *identity matrix*.

Now we are ready to prove Theorem 5.3.14. First let us write $\mathbf{E}_{y,z \sim \mathscr{T}_e}[g(y)g(z)]$ by the multiplication of the vector form of g and the matrix $M(\mathscr{T}_e)$:

$$\mathop{\mathbf{E}}_{y,z \sim \mathscr{T}_e}[g(y)g(z)] = g^T M(\mathscr{T}_e)g = g^T \bigotimes_{i=1}^{R_1} M(\mathscr{H}_\delta(d_i))g.$$

Define distribution $\mathscr{I}_\delta(m) = \delta \mathscr{I}(m) + (1-\delta)\mathscr{H}(m)$. We will show

$$g^T \bigotimes_{i=1}^{R_1} M(\mathscr{H}_\delta(d_i))g \le g^T \bigotimes_{i=1}^{R_1} M(\mathscr{I}_\delta(d_i))g.$$

To see this, we first need prove the following lemma:

Lemma 5.3.17. $M(\mathscr{I}_\delta(m)), M(\mathscr{I}_\delta(m)) - M(\mathscr{H}_\delta(m))$ and $M(\mathscr{I}_\delta(m)) + M(\mathscr{H}_\delta(m))$ are all pos-itive matrices.

Proof. 1) To show $M(\mathscr{I}_\delta(m))$ is positive: Recall that $\mathscr{I}_\delta(m) = \delta \mathscr{I}(m) + (1-\delta)\mathscr{H}(m)$. $\mathscr{I}(m)$ is the identity matrix which is positive. It is easy to check that $M(\mathscr{H}(m))$ is a diagonal matrix with following diagonal elements: for any odd set $S \subseteq [m]$, $M(\mathscr{H}(m))_{S,S} = 0$ and for any even-cardinality set $S \subseteq [m]$, $M(\mathscr{H}(m))_{S,S} = 1$. $M(\mathscr{H}(m))$ is also positive. Therefore, $M(\mathscr{I}_\delta)$ is positive.

2) To show $M(\mathscr{I}_\delta(m)) - M(\mathscr{H}_\delta(m))$ is positive: Since we know $M(\mathscr{I}_\delta(m)) - M(\mathscr{H}_\delta(m)) = \delta(M(\mathscr{I}(m)) - M(\mathscr{N}(m)))$, we only need to show $(M(\mathscr{I}) - M(\mathscr{N}))$ is a positive matrix.

Notice that for any function $h : \{-1,1\}^d \to \mathbb{R}$,

$$h^T M(\mathscr{N}(m))h = \mathop{\mathbf{E}}_{\mathscr{N}(m)}[(h(y)h(z)] \le \mathop{\mathbf{E}}_{\mathscr{N}(m)}[\frac{(h(y)^2 + h(z)^2}{2}]$$

Notice that $\mathscr{N}(m)$ and $\mathscr{I}(m)$ have the same marginal distribution (uniform distribution) on y and z. Then

$$\mathop{\mathbf{E}}_{\mathscr{N}(m)}[\frac{(h(y)^2 + h(z)^2}{2}] = \mathop{\mathbf{E}}_{\mathscr{N}(m)}[(h(y)^2] = \mathop{\mathbf{E}}_{\mathscr{I}(m)}[(h(y)^2] = h^T M(\mathscr{I})h$$

This implies for any h, $h^T(M(\mathscr{I}(m)) - M(\mathscr{N}(m)))h > 0$. Therefore, $M(\mathscr{I}(m)) - M(\mathscr{N}(m))$ is a positive matrix.

3) To show $M(\mathscr{I}_\delta(m)) + M(\mathscr{H}_\delta(m))$ is positive: We know $M(\mathscr{I}_\delta(m)) + M(\mathscr{H}_\delta(m)) = 2(1-\delta)M(\mathscr{H}(m)) + \delta(M(\mathscr{I}) + M(\mathscr{H}(m)))$. We already know $M(\mathscr{H}(m))$ is positive. It remains to prove that $M(\mathscr{I}) + M(\mathscr{H}(m))$ is positive. Notice that

$$-h^T M(\mathscr{N}(m))h = \mathop{\mathbf{E}}_{\mathscr{N}(m)}[-(h(y)h(z)] \le \mathop{\mathbf{E}}_{\mathscr{N}(m)}[\frac{(h(y)^2 + h(z)^2}{2}]$$
$$= \mathop{\mathbf{E}}_{\mathscr{N}(m)}[(h(y)^2] = \mathop{\mathbf{E}}_{\mathscr{I}(m)}[(h(y)^2] = h^T M(\mathscr{I}(m))h. \quad (5.4)$$

Then we have that $M(\mathscr{I}(m)) + M(\mathscr{N}(m))$ is a positive matrix. $\square$

We have shown $M(\mathscr{I}_\delta(m))$, $M(\mathscr{I}_\delta(m)) + M(\mathscr{H}_\delta(m))$, and $M(\mathscr{I}_\delta(m)) - M(\mathscr{H}_\delta(m))$ are all positive matrix. By Lemma 5.6.1 this implies $\bigotimes_{i=1}^{k} M(\mathscr{I}_\delta(d_i)) - \bigotimes_{i=1}^{k} M(\mathscr{H}_\delta(d_i))$ is a positive matrix for any integer $k \geq 1$. Therefore, $\bigotimes_{i=1}^{R_1} M(\mathscr{I}_\delta(d_i)) - \bigotimes_{i=1}^{R_1} M(\mathscr{H}_\delta(d_i))$ is positive. By the property of positive matrix, we have

$$g^T \bigotimes_{i=1}^{R_1} M(\mathscr{H}_\delta(d_i)) g \leq g^T \bigotimes_{i=1}^{R_1} M(\mathscr{I}_\delta(d_i)) g.$$

Now we calculate $g^T \bigotimes_{i=1}^{R_1} M(\mathscr{I}_\delta(d_i)) g$ by g's Fourier Coefficients. Notice that $M(\mathscr{I}_\delta(m))$ is also a diagonal matrix: for any odd-cardinality set $S \subseteq [d]$, $M(\mathscr{I}_\delta(m))_{S,S} = \delta$ and for any even size set S, $M(\mathscr{I}_\delta(m))_{S,S} = 1$. Then $\bigotimes_{i=1}^{R_1} M(\mathscr{I}_\delta(d_i))$ is also diagonal matrix; i.e., for $S, T \subseteq [R_2], S \neq T$, we have

$$\mathop{\mathbf{E}}_{\prod_{i=1}^{R_1} \mathscr{I}_\delta(d_i)} [\chi_S(y)\chi_S(z)] = 0.$$

Also notice that g only has Fourier Coefficients on odd-cardinality set, we can expand $g^T \bigotimes_{i=1}^{R_1} M(\mathscr{I}_\delta(d_i)) g$ as:

$$\sum_{S \subseteq [R_2], |S| \text{ is odd}} \hat{g}(S)^2 \mathop{\mathbf{E}}_{\prod_{i=1}^{R_1} \mathscr{I}_\delta(d_i)} [\chi_S(y)\chi_S(z)].$$

The term $\mathbf{E}_{\prod_{i=1}^{R_1} \mathscr{I}_\delta(d_i)}[\chi_S(y)\chi_S(z)]$ can be further written as:

$$\prod_{i=1}^{R_1} \mathop{\mathbf{E}}_{\mathscr{I}_\delta(d_i)} [\chi_{(S \cap \pi_e^{-1}(i))}(y)\chi_{(S \cap \pi_e^{-1}(i))}(z)].$$

Since S is odd-cardinality set, there must exist some i_0 ($1 \leq i_0 \leq R_1$) such that the intersection set between S and $\pi_e^{-1}(i)$ has odd cardinality. Recall that for odd-cardinality set S, $M(\mathscr{I}_\delta(d_{i_0})_{S,S}) = \delta$. We know therefore

$$\mathop{\mathbf{E}}_{\prod_{i=1}^{R_1} \mathscr{I}_\delta(d_i)} [\chi_S(y)\chi_S(z)] = \prod_{i=1}^{R_1} \mathop{\mathbf{E}}_{\mathscr{I}_\delta(d_i)} [\chi_{(S \cap \pi_e^{-1}(i))}(y)\chi_{(S \cap \pi_e^{-1}(i))}(z)]$$

$$\leq \mathop{\mathbf{E}}_{\mathscr{I}_\delta(d_{i_0})} [\chi_{(S \cap \pi_e^{-1}(i_0))}(y)\chi_{(S \cap \pi_e^{-1}(i_0))}(z)] = M(\mathscr{I}_\delta(d_i))_{S \cap \pi_e^{-1}(i_0), S \cap \pi_e^{-1}(i_0)} = \delta. \quad (5.5)$$

This implies

$$\mathop{\mathbf{E}}_{\prod_{i=1}^{R_1} \mathscr{I}_\delta(d_i)} [g(y)g(z)] \leq \delta \sum_{S \subseteq [R_2]} \hat{g}(S)^2 \leq \delta.$$

$\square$

Overall, we have proved that

$$\mathop{\mathbf{E}}_{\mathscr{T}_e}[g(y)g(z)] \leq \mathop{\mathbf{E}}_{\prod_{i=1}^{R_1} \mathscr{I}_\delta(d_i)} [g(y)(z)] \leq \delta.$$

and therefore $\mathbf{E}_{u,v \sim P, y, z \sim \mathscr{T}_e}[g_v(y)g_v(z)]$ is also bounded by δ.

116

So far, We have shown

$$\mathop{\mathbf{E}}_{\mathscr{T}_e}[\tfrac{1}{8}(f_u(x)g_v(y) + g_v(y)g_v(z) + f_u(x)g_v(z))] \le \frac{3}{8}\delta$$

In the next section we analyze:

$$\mathop{\mathbf{E}}_{(u,v)\sim P, x,y,z \sim \mathscr{T}_e}[-\frac{3}{8}f_u(x)g_v(y)g_v(z)].$$

Analyzing $f_u(x)g_v(y)g_v(z)$

We will first describe some new Fourier analysis tools . Recall the definition of influence on coordinate.

Definition 5.3.18. *Given function $f : \{-1,1\}^n \to \mathbb{R}$, $i \in [n]$, we define the* influence *of i on f to be*

$$\mathrm{Inf}_i(f) = \sum_{S \ni i} \hat{f}(S)^2.$$

In this work we also define the influence on set:

Definition 5.3.19. [2] *For a function $f : \{-1,1\}^n \to \mathbb{R}$, $T \subseteq [n]$, define*

$$\mathrm{Inf}_T(f) = \sum_{S \supseteq T} \hat{f}(S)^2.$$

Recall the definition of Bonami-Beckner noise operator T_ρ.

Definition 5.3.20. *$(\Omega, \mu) = \prod_{i=1}^n (\Omega_i, \mu_i)$ is finite product probability space. For $x = (x_1, ..x_n) \in \Omega$, random variable $y = (y_1, ..y_n) \in \Omega$ is called ρ correlated with x if each y_i is independently set to be x_i with probability ρ and set to be a random sample from (Ω_i, μ_i) with probability $(1-\rho)$.*

The Bonami-Beckner T_ρ is a function mapping from $\{f : \Omega \to \mathbb{R}\}$ to $\{g : \Omega \to \mathbb{R}\}$ defined as follows: $g(x) = T_\rho f(x) = \mathbf{E}[f(y)]$, where y is a random variable ρ correlated with x.

Remark 5.3.21. *In this Chapter, we assume μ_i to be the uniform distribution unless addition explanation. e.g., when we define Bonami-Beckner Operator on $\{-1,1\}^n$, we refer to the product probability space $(\{-1,1\}, \text{uniform distribution on } \{-1,1\})^n$.*

Also Bonami-Beckner Operator is dependent on how we write the product space. For example, $\{-1,1\}^{R_2}$ and $\prod_{i=1}^{R_1}\{-1,1\}^{\pi_e^{-1}(i)}$ have different Bonami-Beckner Operator. By $\{-1,1\}^{R_2}$, we mean product of spaces $\Omega_i = \{-1,1\}$ for $i = 1,2..R_2$; by $\prod_{i=1}^{R_1}\{-1,1\}^{\pi_e^{-1}(i)}$ we mean the product of spaces $\Omega_i = \{-1,1\}^{\pi_e^{-1}(i)}$ for $i = 1,2..R_1$. By definition, the Bonami-Beckner operators are different in above two cases. In this work we mostly use the first operator unless additional explanation.

It is well known fact that the total influence over all coordinates for "smoothed function" $T_{1-\gamma}f$ is bounded. We generalize it by proving that for "smoothed function", its total influence on all constant size sets is also bounded.

Lemma 5.3.22. *For any Boolean function $f : \{-1,1\}^n \to [-1,1]$ and $\gamma < 1, m \in \mathbb{N}$,*

$$\sum_{S \subseteq [n], |S| \le m} \mathrm{Inf}_S(T_{1-\gamma}f) \le (\frac{m}{2\gamma})^m.$$

---

[2]The definition here is different from [83].

Proof. Notice for every set $S \subseteq [n]$, it contains $\sum_{i=0}^{m} \binom{|S|}{i}$ subset with size smaller than m. We know that $\sum_{i=0}^{m} \binom{|S|}{i} \leq (|S|+1)^m$ (imagine the process we select m times from $|S|$ element and every time, we chose to select nothing or one of the $|S|$ element). Then we have

$$\sum_{S \subseteq [n]} (\text{Inf}_S T_{1-\gamma} g) \leq \sum_{S \subseteq [n]} (|S|+1)^m (1-\gamma)^{2|S|} \hat{f}(S)^2. \tag{5.6}$$

With the inequality from Lemma 5.3.23, it can be shown $(|S|+1)^m(1-\gamma)^{2|S|} \leq (\frac{m}{2\gamma})^m$ and therefore

$$(5.6) \leq (\frac{m}{2\gamma})^m \sum_{S \subseteq [n]} \hat{f}(S)^2 = (\frac{m}{2\gamma})^m.$$

$\square$

Lemma 5.3.23. *For $\gamma > 1/2, x > 0, m \in \mathbb{N}, f(x) = (1-\gamma)^{2x}(x+1)^m$, we have $f(x) \leq (m/\gamma)^m$*

Proof. Notice that $1 - \gamma \leq e^{-\gamma}$, we have $f(x) \leq e^{-2x\gamma}(x+1)^m = h(x)$. By some calculus, $h(x)$ reaches its maximum when $x = \frac{m}{2\gamma} - 1$. Then we have $f(x) \leq h(\frac{m}{2\gamma}) \leq (\frac{m}{2\gamma})^m$. $\square$

In this work we extend the hypercontractive inequality into the following form:

Lemma 5.3.24. *Let function $f : \{-1,1\}^n \to [-1,1]$ and $0 < \gamma < 1$, then*

$$\|T_{1-\gamma}f\|_3 \leq \|f\|_2^{\frac{2+2\gamma}{3}}.$$

Proof.

$$\|T_{1-\gamma}f\|_3 = \mathbf{E}[|T_{1-\gamma}f(x)|^3]^{1/3} \leq \mathbf{E}[|T_{1-\gamma}f(x)|^{2+2\gamma}]^{1/3}$$

Notice that $(1-\gamma) \leq \sqrt{\frac{1}{1+2\gamma}}$, we can use hyper-inequality and get:

$$\mathbf{E}[[T_{1-\gamma}f|^{2+\gamma}]^{1/3} = \|T_{1-\gamma}f\|_{2+2\gamma}^{\frac{2+2\gamma}{3}} \leq \|f\|_2^{\frac{2+2\gamma}{3}}$$

$\square$

As a corollary, we have

Corollary 5.3.25.

$$\|T_{1-\gamma}f\|_3 \leq \|(T_{(1-0.5\gamma)}f)\|_2^{\frac{2+\gamma}{3}}.$$

Proof. Since

$$\|T_{1-\gamma}f\|_3 \leq \|T_{1-\gamma+\gamma^2/4}f\|_3 = \|T_{(1-0.5\gamma)}T_{(1-0.5\gamma)}f)\|_3 \leq \|(T_{(1-0.5\gamma)}f)\|_2^{\frac{2+\gamma}{3}}.$$

$\square$

Now we are ready to analyze $f_u(x)g_v(y)g_v(z)$. Following is the key theorem we need:

Theorem 5.3.26. *There exists some positive constant γ, τ depending only on d, δ such that for any odd Boolean functions $f : \{-1, 1\}^{R_1} \to \{-1, 1\}$ and $g : \{-1, 1\}^{R_2} \to \{-1, 1\}$, if for every $1 \le i \le R_1$ and odd-cardinality set $S \subseteq \pi_e^{-1}(i)$,*

$$\min(\mathrm{Inf}_i\, T_{(1-0.5\gamma)} f, \mathrm{Inf}_S\, T_{(1-0.5\gamma)} g) \le \tau,$$

then

$$\left| \mathop{\mathbf{E}}_{x,y,z \sim \mathscr{T}_e} [f(x)g(y)g(z)] \right| \le 3\sqrt{\delta}.$$

Proof. The first idea is that we can apply some smooth operator to f, g and the expectation would not change too much.

Formally, we claim there exists some positive constant γ', γ ($\gamma' > \gamma > 0$) depending only on δ and d such that

$$| \mathop{\mathbf{E}}_{x,y,z \sim \mathscr{T}_e} [f(x)g(y)g(z)] - \mathop{\mathbf{E}}_{x,y,z \sim \mathscr{T}_e} [T_{1-\gamma} f(x) T_{1-\gamma'} g(y) T_{1-\gamma'} g(z)]| \le \sqrt{\delta}, \qquad (5.7)$$

Above claim is proved by Lemma 5.4.2.

From Lemma 5.3.12, we know

$$\rho(\prod_{i=1}^{R_1} \mathscr{X}^i, \prod_{i=1}^{R_1} \mathscr{Y}^i \times \mathscr{Z}^i; \prod_{i=1}^{R_1} \mathscr{F}_\delta(d_i)) \le \sqrt{\delta}$$

Therefore, notice that $\mathbf{E}[T_{1-\gamma} f(x)] = 0$ and both $T_{1-\gamma} f(x)$ and $T_{1-\gamma'} g(y) T_{1-\gamma'} g(z)$ are bounded in $[-1, 1]$ and they have variance less than 1. Therefore, we have

$$| \mathop{\mathbf{E}}_{x,y,z \sim \prod_{i=1}^{R_1} \mathscr{F}_\delta(d_i)} [T_{1-\gamma} f(x) T_{1-\gamma'} g(y) T_{1-\gamma'} g(z)] = |\mathbf{Cov}(T_{1-\gamma} f(x), T_{1-\gamma'} g(y) T_{1-\gamma'} g(z))| \le \sqrt{\delta}.$$

$$(5.8)$$

Following we will prove the expectation of the product of "smoothed" function $(T_{1-\gamma} f)(T_{1-\gamma'} g)(T_{1-\gamma'})$ is very close on distribution $\mathscr{T}_e = \prod_{i=1}^{R_1} \mathscr{H}_\delta(d_i)$ and $\prod_{i=1}^{R_1} \mathscr{F}_\delta(d_i)$. Formally, we will show:

$$| \mathop{\mathbf{E}}_{\prod_{i=1}^{R_1} \mathscr{F}_\delta(d_i)} [T_{1-\gamma} f(x) T_{1-\gamma'} g(y) T_{1-\gamma'} g(z)] - \mathop{\mathbf{E}}_{\mathscr{T}_e = \prod_{i=1}^{R_1} \mathscr{H}_\delta(d_i)} [T_{1-\gamma} f(x) T_{1-\gamma'} g(y) T_{1-\gamma'} g(z)]| \le \sqrt{\delta}$$

$$(5.9)$$

Combining (5.7), (5.8), (5.9), we then prove $\mathbf{E}_{x,y,z \sim \mathscr{T}_e} [f(x)g(y)g(z)] \le 3\sqrt{\delta}$

It remains to prove (5.9). The technique for the proof of (5.9) is similar to the Invariance Principle proof in [116]. Roughly speaking, we show every time we can change the distribution at one coordinate from $\mathscr{H}_\delta(d_i)$ to $\mathscr{F}_\delta(d_i)$ and the change will be bounded by the influence at that coordinate. And then we use the fact that for "smoothed" function the total influence is bounded.

To clarify, let us start by changing the distribution at the first coordinate from $\mathscr{H}_\delta(d_1)$ to $\mathscr{F}_\delta(d_1)$. Without loss of generality, let us assume that $\pi_e^{-1}(1) = \{1, 2, ..d_1\}$. Let us write x' for $(x_2, ..x_n)$ and y' for $(y_{d_1+1}, ..y_{R_2})$ and z' for $(z_{d_1+1}, .z_{R_2})$.

We can think of f as a function only on variable x_1 and write f by its Fourier expansion as :

$$f(x) = F_\emptyset(x') + x_1 F_{\{1\}}(x').$$

119

Similarly, we can view g as a function only on variable $y_1,..y_{d_1}$ and we denote g's Fourier expansion (on $y_1,..y_{d_1}$) as

$$g = \sum_{S \subseteq [d_1]} \chi_S(y) G_S(y').$$

For any $S \subseteq [d_1]$, we can also represent G_S by g's original Fourier coefficients $\hat{g}(Q)$ ($Q \subseteq [R_2]$) as follows:

$$G_S(y') = \mathop{\mathbf{E}}_{y_1,..y_{d_1}} [g(y)\chi_S(y)] = \sum_{Q \subseteq [R_2], Q \cap [d_1] = S,} \hat{g}(Q) \chi_{Q \setminus S}(y').$$

We know therefore $G_S(y') = \mathbf{E}_{y_1,..y_{d_1}}[g(y)\chi_S(y)]$ is always bounded between $[-1,1]$. Similarly $F_{\{1\}} = \sum_{i \in Q} \chi_{Q \setminus \{1\}}(x')$ and it is also bounded in $[-1,1]$.

It is easy to see $T_{1-\gamma}f$ has Fourier Expansion $T_{1-\gamma}F_\phi + x_1(1-\gamma)T_{1-\gamma}F_{\{1\}}$ (on variable x_1) and $T_{1-\gamma'}g$ has Fourier Expansion $\sum_{S \subseteq [d_1]} \chi_S (1-\gamma')^{|S|} G_S$ (on variable $y_1,..y_{d_1}$).

By Lemma 5.3.6, if we take conditional expectation only on $(x, y_1,..y_{d_1}, z_1..z_{d_1})$, we have

$$\mathop{\mathbf{E}}_{\mathscr{F}_\delta(d_1)}[T_{1-\gamma}f(x)T_{1-\gamma'}g(y)T_{1-\gamma'}g(z)] - \mathop{\mathbf{E}}_{\mathscr{H}_\delta(d_1)}[T_{1-\gamma}f(x)T_{1-\gamma'}g(y)T_{1-\gamma'}g(z)$$

$$= -(1-\delta)\sum_S (1-\gamma)(1-\gamma')^{2|S|}T_{1-\gamma}F_{\{1\}}(x')T_{1-\gamma'}G_S(y')T_{1-\gamma'}G_S(z')$$

Further condition on x', y', z', we can calculate the difference of changing the first coordinate as follows:

$$\mathop{\mathbf{E}}_{\mathscr{H}_\delta(d_1) \times \prod_{k=2}^{R_1} \mathscr{F}_\delta(d_k)}[T_{1-\gamma}f(x)T_{1-\gamma'}g(y)T_{1-\gamma'}g(z)] - \mathop{\mathbf{E}}_{\prod_{k=1}^{R_1} \mathscr{F}_\delta(d_k)}[T_{1-\gamma}f(x)T_{1-\gamma'}g(y)T_{1-\gamma'}g(z)]$$

$$= \mathop{\mathbf{E}}_{\prod_{k=2}^{R_1} \mathscr{F}_\delta(d_i)} \left[\mathop{\mathbf{E}}_{\mathscr{H}_\delta(d_1)}[T_{1-\gamma}f(x)T_{1-\gamma'}g(y)T_{1-\gamma'}g(z)] - \mathop{\mathbf{E}}_{\mathscr{F}_\delta(d_1)}[T_{1-\gamma}f(x)T_{1-\gamma'}g(y)T_{1-\gamma'}g(z)] \right]$$

$$= -(1-\delta) \sum_{|S| \text{ is odd}, S \subseteq [d_1]} (1-\gamma)(1-\gamma')^{2|S|} \mathop{\mathbf{E}}_{\prod_{k=2}^{R_1} \mathscr{H}_\delta(d_k)}[T_{1-\gamma}F_{\{1\}}1(x')T_{1-\gamma'}G_S(y')T_{1-\gamma'}G_S(z')].$$

$$(5.10)$$

Notice that $\prod_{i=2}^{R_1} \mathscr{H}_\delta(d_i)$ has uniform marginal distribution on x', y', z'. We have

$$\mathop{\mathbf{E}}_{\prod_{i=2}^{R_1} \mathscr{H}_\delta(d_i)}[T_{1-\gamma}F_1 T_{1-\gamma'}G_S T_{1-\gamma'}G_S] \leq \|T_{1-\gamma}F_{\{1\}}\|_3 \|T_{1-\gamma'}G_S\|_3^2 \quad \text{(Holder's Inequality)}$$

$$\leq \|T_{1-0.5\gamma}F_{\{1\}}\|_2^{\frac{2+\gamma}{3}} \|T_{1-0.5\gamma'}G_S\|_2^{\frac{4+2\gamma'}{3}} \quad \text{(Corollary 5.3.25)}$$

By representing G_S by g's original Fourier coefficients, we have

$$\|T_{1-0.5\gamma'}G_S\|_2^2 = \|T_{1-0.5\gamma'} \sum_{Q:Q \cap [d_1]=S} \hat{g}(Q)\chi_{Q \setminus S}\|_2^2 = \sum_{Q:Q \cap [d_1]=S} (1-0.5\gamma')^{2|Q|-2|S|}\hat{g}(Q)^2$$

$$\leq \sum_{Q:S \subseteq Q \subseteq [R_2]} (1-0.5\gamma')^{2|Q|-2|S|}\hat{g}(Q)^2 = \frac{(\text{Inf}_S T_{1-0.5\gamma'}g)}{(1-0.5\gamma')^{2|S|}}$$

And similarly,

$$\|T_{1-0.5\gamma}F_{\{1\}}\|_2^2 \leq \frac{\text{Inf}_1 T_{1-0.5\gamma}f}{(1-0.5\gamma)^2}$$

120

We can then bounding $|(5.10)|$ by

$$\left| (1-\delta) \sum_{|S| \text{ is odd}, S \subseteq [d_1]} (1-\gamma')^{2|S|} (1-\gamma) \left(\frac{\mathrm{Inf}_1 T_{1-0.5\gamma} f}{(1-0.5\gamma)^2} \right)^{\frac{1+0.5\gamma}{3}} \left(\frac{\mathrm{Inf}_S T_{1-0.5\gamma'} g}{(1-0.5\gamma')^{2|S|}} \right)^{\frac{2+\gamma'}{3}} \right|$$

$$\leq \sum_{|S| \text{ is odd}, S \subseteq [d_1]} (\mathrm{Inf}_i T_{1-0.5\gamma} f)^{\frac{1+0.5\gamma}{3}} (\mathrm{Inf}_S T_{1-0.5\gamma'} g)^{\frac{2+\gamma'}{3}}$$

Take τ to be $(\frac{\sqrt{\delta}}{2(d/0.5\gamma)^d})^{\frac{6}{\gamma}}$ and notice that $\gamma' \geq \gamma$ (which implies $\mathrm{Inf}_S T_{1-0.5\gamma'} g \geq \mathrm{Inf}_S T_{1-0.5\gamma} g$), we have

$$\min(\mathrm{Inf}_1 T_{1-0.5\gamma} f, \mathrm{Inf}_S T_{1-0.5\gamma'} g) \leq \tau$$

and therefore

$$(\mathrm{Inf}_1 T_{1-0.5\gamma} f)^{\frac{1+0.5\gamma}{3}} (\mathrm{Inf}_S T_{1-0.5\gamma'} g)^{\frac{2+\gamma'}{3}} \leq \tau^{\gamma/6} (\mathrm{Inf}_1 T_{1-0.5\gamma} f)^{\frac{1}{3}} (\mathrm{Inf}_S T_{1-0.5\gamma} g)^{\frac{2}{3}}.$$

Recall $d_1 \leq d$, then we can bound the difference of changing the first coordinate from $\mathcal{H}_\delta(d_1)$ to $\mathcal{F}_\delta(d_i)$ by

$$\tau^{\gamma/6} \sum_{|S| \text{ is odd}, S \subseteq [d_1]} (\mathrm{Inf}_1 T_{1-0.5\gamma} f)^{\frac{1}{3}} (\mathrm{Inf}_S T_{1-0.5\gamma} g)^{\frac{2}{3}} \leq \tau^{\gamma/6} \sum_{|S| \text{ is odd}, S \subseteq [d_1]} (\mathrm{Inf}_1 T_{1-0.5\gamma} f + \mathrm{Inf}_S T_{1-0.5\gamma} g)$$

$$\leq \tau^{\gamma/6} (2^{d-1} \mathrm{Inf}_1 T_{1-0.5\gamma} f + \sum_{|S| \text{ is odd}, S \subseteq [d_1]} \mathrm{Inf}_S T_{1-0.5\gamma} g) \quad (5.11)$$

Similarly calculation will show that for any i,

$$\left| \mathop{\mathbf{E}}_{\prod_{k=1}^{i} \mathcal{F}_\delta(d_k) \times \prod_{k=i+1}^{R_1} \mathcal{H}_\delta^k(d_k)} [T_{1-0.5\gamma} f(x) T_{1-0.5\gamma'} g(y) T_{1-0.5\gamma'} g(z)] - \right.$$

$$\left. \mathop{\mathbf{E}}_{\prod_{k=1}^{i-1} \mathcal{F}_\delta(d_k) \times \prod_{k=i}^{R_1} \mathcal{H}_\delta^k(d_k)} [T_{1-0.5\gamma} f(x) T_{1-0.5\gamma'} g(y) T_{1-0.5\gamma'} g(z)] \right|$$

$$\leq \tau^{\gamma/6} (2^{d-1} \mathrm{Inf}_i T_{1-0.5\gamma} f + \sum_{|S| \text{ is odd}, S \subseteq \pi^{-1}(i)} \mathrm{Inf}_S T_{1-0.5\gamma} g)$$

If we sum above inequality over i from 1 to R_1, we have

$$\left| \mathop{\mathbf{E}}_{\prod_{i=1}^{R_1} \mathcal{F}_\delta(d_i)} [T_{1-0.5\gamma} f(x) T_{1-0.5\gamma'} g(y) T_{1-0.5\gamma'} g(z)] - \mathop{\mathbf{E}}_{\mathcal{T}_e = \prod_{i=1}^{R_1} \mathcal{H}_\delta(d_i)} [T_{1-0.5\gamma} f(x) T_{1-0.5\gamma'} g(y) T_{1-0.5\gamma'} g(z)] \right|$$

$$\leq \tau^{\gamma/6} \sum_{i=1}^{R_1} (2^{d-1} \mathrm{Inf}_i T_{1-0.5\gamma} f + \sum_{S \subseteq \pi^{-1}(i)} \mathrm{Inf}_S T_{1-0.5\gamma} g)$$

$$\leq \tau^{\gamma/6} (2^{d-1}(1/\gamma) + (d/\gamma)^d) \quad \text{(By Lemma 5.3.22)}$$

$$\leq \tau^{\gamma/6} (2(d/\gamma)^d) = \sqrt{\delta}.$$

This proves (5.9). $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \square$

121

Soundness Proof

Now we prove the following soundness theorem.

Theorem 5.3.27. *For any ϵ, if some proof passes with probability more than $5/8 + \epsilon$, then we have $opt(\mathcal{L}) > \eta$. Here $\eta > 0$ is some positive constant only dependent on ϵ and d.*

Proof. Recall that $\epsilon = 2\sqrt{\delta}$. Suppose some proof pass the test with probability $5/8 + \epsilon$, then

$$\mathop{\mathbf{E}}_{e=(u,v)\sim P,\mathcal{T}_e}[\tfrac{5}{8} + \tfrac{1}{8}(f_u(x) + g_v(y) + g_v(z)) + \tfrac{1}{8}(f_u(x)g_v(y) + g_v(y)g_v(z) + f_u(x)g_v(z))$$
$$- \tfrac{3}{8}f_u(x)g_v(y)g_v(z)] > \tfrac{5}{8} + 2\sqrt{\delta}.$$

By the oddness of f_u, g_v and Theorem 5.3.13, 5.3.14, we know

$$\mathop{\mathbf{E}}_{e=(u,v)\sim P,\mathcal{T}_e}[\tfrac{5}{8} + \tfrac{1}{8}(f_u(x)+g_v(y)+g_v(z)) + \tfrac{1}{8}(f_u(x)g_v(y)+g_v(y)g_v(z)+f_u(x)g_v(z))] < \frac{5}{8}+\frac{3}{8}\delta < \frac{5}{8}+\frac{3}{8}\sqrt{\delta}.$$

Therefore,

$$|\mathop{\mathbf{E}}_{e=(u,v)\sim P,\mathcal{T}_e}[f_u(x)g_v(y)g_v(z)]| > \frac{13}{3}\sqrt{\delta} > 4\sqrt{\delta}.$$

Then by an average argument, for $\sqrt{\delta}$ fraction of the edges (u,v), we have

$$\mathop{\mathbf{E}}_{x,y,z\sim\mathcal{H}_\delta^R}[f_u(x)g_v(y)g_v(z)] > 3\sqrt{\delta}$$

We call these edges "good". By Theorem 5.3.26, we know for every "good" edge (u,v), there must exists some i, and odd set $S \subseteq L(i)$ such that:

$$\min(\mathrm{Inf}_i T_{(1-0.5\gamma)}f_u, \mathrm{Inf}_S T_{(1-0.5\gamma)}g_v) \geq \tau. \tag{5.12}$$

We can define the following randomized label strategy for $\mathcal{L}$:
For $u \in U$, define

$$S_u = \{i | \mathrm{Inf}_i(T_{1-0.5\gamma}f_u) \geq \tau\}$$

and $v \in V$, define

$$S_v = \{j | j \in S, \mathrm{Inf}_S(T_{1-0.5\gamma}g_v) \geq \tau, |S| \leq d, |S| \text{ is odd}\}.$$

Given (10.24), for good "edges" (u,v), S_u, S_v must be both nonempty and there exists some $i \in S_u$ such that $\pi_e(i) \cap S_v \neq \emptyset$.

Also, by Lemma 5.3.22, we know that $\sum_S \mathrm{Inf}_S(T_{1-0.5\gamma}g_v) \leq (d/\gamma)^d$. Therefore, the number of S that satisfies $\mathrm{Inf}_S(T_{1-0.5\epsilon}g_v) \geq \tau$ is at most $(d/\gamma)^d/\tau$ and therefore $|S_v| \leq d(d/\gamma)^d/\tau$. Similarly, we have $|S_u| \leq 1/(\gamma\tau)$.

For every vertex $w \in U \cup V$, our labelling strategy is to randomly pick a label from S_w for w. We know for ever "good edge", they are satisfied by probability at least $\frac{1}{|S_u||S_v|}$. Overall, our randomized strategy satisfies at least

$$\eta = \frac{\sqrt{\delta}}{|S_u||S_v|} \geq \sqrt{\delta}(\frac{\gamma}{d})^{d+1}\tau^2$$

fraction of all the edges. Notice here η, γ, τ are positive constant depending on δ and d. $\square$

5.4 Noise Operator

For the product space $\{-1, 1\}^n$ with uniform distribution at each coordinate, it is easy to check that Fourier Expansion is the Efron-Stein Decomposition ; i.e., $f = \sum_{S \subseteq [n]} f_S(x)$ where $f_S(x) = \hat{f}(S)\chi_S(x)$. It is also easy to check for the product space $\prod_{i=1}^{R_2} \left(\{-1, 1\}^{\pi_e^{-1}(i)} \right)$ with uniform distribution on each $\{-1, 1\}^{\pi_e^{-1}(i)}$, $f(x) = \sum_{S \subseteq R_2} f_S(x)$ with $f_S(x) = \sum_{\pi_e(T)=S} \hat{f}(T)\chi_T(x)$ is the Efron-Stein Decomposition.

Following Lemma is proved in [115] (Proposition 2.12)

Lemma 5.4.1. *Let* $(\Omega \times \Theta, \mu) = \prod_{i=1}^n (\Omega_i \times \Theta_i, \mu_i)$ *be a finite product probability spaces. And* $\rho(\Omega_i, \Theta_i; \mu_i) < \rho_i$. *Suppose* $f : \Theta \to R$ *has the Efron-Stein Decomposition* $\sum_{S \subseteq n} f_S$ *on* $\prod_{i=1}^n (\Omega_i, \mu_1)$. *And let* U_μ *be the conditional operator of* μ *mapping function* $f : \Theta \to \mathbb{R}$ *to* $g : \Omega \to \mathbb{R}$, *then*

$$\|U_\mu f_S\|_2 \le \left(\prod_{i \in S} \rho_i \right) \|f\|_2.$$

Now we prove that for our distribution $\mathcal{T}_e$, the expectation of $f(x)g(y)g(z)$ is closed to its smoothed version $T_{1-\gamma'}f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)$ for some small constant γ, γ'.

Lemma 5.4.2. *Let* f *be a function mapping from* $\{-1, 1\}^{R_1} \to [-1, 1]$ *and* g *be function mapping from* $\{-1, 1\}^{R_2} \to [-1, 1]$. *For any small constant* $\beta > 0$, *let* $\rho_0 = 1 - \frac{\delta^2}{2^{2d+1}d^2}, \gamma = \frac{\beta^3}{d}(1 - \rho_0), \gamma' = \frac{\beta^3 \gamma^{2d}}{2}$, *we have:*

$$\underset{x,y,z \sim \mathcal{T}_e}{\mathbf{E}} [T_{1-\gamma'}f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)] - \underset{x,y,z \sim \mathcal{T}_e}{\mathbf{E}} [f(x)g(y)g(z)] \le 3\beta.$$

Proof. By Lemma 5.3.8 , we know $\rho(\mathcal{X} \times \mathcal{Y}, \mathcal{Z}; \mathcal{T}_e) \le 1 - \frac{\delta^2}{2^{2d+1}d^2} = \rho_0$.

Let us write the Efron-Stein Decomposition for $g(z)$ on $\prod_{i=1}^{R_1}(\mathcal{Z}^i, T_e^i)$ as $\sum_{S \subseteq [R_1]} g_S(z)$ and we know $g_S(z) = \sum_{\pi(T)=S} \hat{g}(S)\chi_T(z)$. We also write $(f(x)g(y))$'s Efron-Stein Decomposition on $\prod_{i=1}(\mathcal{X}^i \times \mathcal{Y}^i, \mathcal{T}_e^i)$ as $\sum_{S \subseteq [R_1]} F_S(x,y)$. Let $U_{\mathcal{T}_e}$ be the conditional operator associated with correlated probability space $(\prod_{i=1}^{R_1} \mathcal{X}^i \times \mathcal{Y}^i, \prod_{i=1}^{R_1} \mathcal{Z}^i, \mathcal{T}_e)$. We denote I as the identity operator. Then we have

$$\mathbf{E}[f(x)g(y)g(z)] - \mathbf{E}[(f(x)g(y)T_{1-\gamma}g(z)] = \mathbf{E}[f(x)g(y)(I - T_{1-\gamma})g(z)]$$
$$= \sum_{S \subseteq [R_1]} \mathbf{E}[F_S(x,y)U_{\mathcal{T}_e}(I - T_{1-\gamma})g_S(x,y)] \quad \text{(By the orthogonality)} \quad (5.13)$$

It is easy to check that for the Efron-Stein decomposition of function $(I - T_{1-\gamma})g$, we have

$$((I - T_{1-\gamma})g)_S = (I - T_{1-\gamma})(g_S).$$

Then Lemma 5.4.1, we have that

$$\|U_{\mathcal{T}_e}(I - T_{1-\gamma})g_S\|_2 \le \rho_0^{|S|}\|(I - T_{1-\gamma})g_S\|_2.$$

Denote $\pi_e(Q)$ as $\{i | \pi_e(i) \in Q\}$. As π_e is d-to-1 projection, we have $|Q| \le |S|d$. For any $Q \subseteq [R_2]$, We know

$$\|(I - T_{1-\gamma})g_S\|_2^2 = \sum_{Q \subseteq [R_2], \pi_e(Q)=S} (1 - (1-\gamma)^{2|Q|})^2 \hat{g}(Q)^2 \chi_T \le (1 - (1-\gamma)^{2|S|d})^2 \|g_S\|_2^2.$$

Therefore,

$$||U_{\mathcal{T}_e}(I - T_{1-\gamma})g_S||_2 \le \rho_0^{|S|}\sqrt{1-(1-\gamma)^{2|S|d}}||g_S||_2 \le \min(\rho_0^{|S|}, \sqrt{1-(1-\gamma)^{2|S|d}})||g_S||_2.$$

When $|S| \ge \frac{\log\beta}{\log\rho_0}$, we know $\rho_0^{|S|} \le \beta$. When $|S| \le \frac{\log\beta}{\log\rho_0}$, we have

$$1 - (1-\gamma)^{2|S|d} = 1 - \left(1 - \frac{\beta^3(1-\rho_0)}{d}\right)^{2|S|d} \le 1 - (1-\beta^3(1-\rho_0))^{2\frac{\log\beta}{\log\rho_0}} = o(\beta^2).$$

Therefore,

$$\min(\rho_0^{|S|}, \sqrt{1-(1-\gamma)^{2|S|d}}) \le \beta.$$

By Cauchy-Schwarz, we get:

$$(5.13) \le \sqrt{\sum_{S\subseteq[R_1]}||F_S||_2^2 \sum ||U_{\mathcal{T}_e}(I-T_{1-\gamma})g_S||_2^2} \le \beta\sqrt{\sum_{S\subseteq[R_1]}||F_S||_2^2 \sum_{S\subseteq[R_1]}||g_S||_2^2} \le \beta \quad (5.14)$$

If we apply the same calculation above when treating $f(x)T_{1-\gamma}g(z)$ as a whole and notice that $\rho(\mathcal{X} \times \mathcal{Z}, \mathcal{Y}; \mathcal{T}_e) \le \rho_0$, we would get:

$$|\mathbf{E}[f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)] - \mathbf{E}[(f(x)g(y)T_{1-\gamma}g(z))]| \le \beta \quad (5.15)$$

It remains to show

$$|\mathbf{E}_{\mathcal{T}_e}[f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)] - \mathbf{E}_{\mathcal{T}_e}[T_{1-\gamma'}f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)]| \le \beta.$$

However, we can not apply the same trick again as that $\rho(\mathcal{X}^i, \mathcal{Y}^i \times \mathcal{Z}^i, \mathcal{H}_\delta(d_i)) = 1$.

Recall the definition of the Bonami-Beckner operator, we can rewrite $\mathbf{E}_{\mathcal{T}_e}[f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)]$ as $\mathbf{E}_{(\mathcal{T}_e)^*}[f(x)g(y^*)g(z^*)]$ where $(\mathcal{T}_e)^*$ is the distribution as follows: first we generate x, y, z by distribution $\mathcal{T}_e$ and then we independently reset each bits in y, z with some independent random bit with probability γ and get y^*, z^*.

Recall that $\mathcal{T}_e = \prod_{i=1}^{R_1}\mathcal{T}_e^i$ where each distribution $\mathcal{T}_e^i$ (on $\mathcal{X}^i \times \mathcal{Y}^i \times \mathcal{Z}^i$) is set to be $\mathcal{H}_\delta(d_i)$. It is easy to check that $(\mathcal{T}_e)^* = \prod_{i=1}^{R_1}(\mathcal{T}_e^i)^*$ where $(\mathcal{T}_e^i)^* = \mathcal{H}_\delta^*(d_i)$ is the distribution such that we first generate $(x, y_1..y_{d_i}, z_1..z_{d_i})$ by $\mathcal{H}_\delta(d_i)$ and then we independently reset every coordinate y_i and z_i to be some random bit with probability γ.

Now we will show $\rho(\mathcal{X}^i, \mathcal{Y}^i \times \mathcal{Z}^i; \mathcal{H}_\delta^*(d_i)) \le (1-\gamma^{2d}/2)$. By the definition of $\mathcal{H}_\delta^*(d_i)$, there is probability $(\gamma)^{2d_i}$ such that y_i, z_i are all reset. When this happen, x is independent with y, z. We call $\mathbf{V}$ the event that "y_i, z_i are all reset". Then we have

$$\rho(\mathcal{X}^i, \mathcal{Y}^i \times \mathcal{Z}^i; \mathcal{H}_\delta^*(d_i)) = \sup_{\substack{f, G, \mathbf{E}[f]=\mathbf{E}[G]=0 \\ \mathbf{E}[f^2]=\mathbf{E}[G^2]=1}} (1-\gamma^{2d_i})\mathbf{E}[f(x)G(y,z)|\overline{\mathbf{V}}]$$

$$+ \gamma^{2d_i}\mathbf{E}[f(x)G(y,z)|\mathbf{V}] \quad (5.16)$$

Notice that event $\mathbf{V}$ is independent with x, we have $\mathbf{E}[f(x)G(y,z)|\mathbf{V}] = \mathbf{E}[f(x)|V]\mathbf{E}[G(y,z)|\mathbf{V}] = \mathbf{E}[f(x)]\mathbf{E}[G(y,z)|\mathbf{V}] = 0$.

124

Also since

$$1 = \mathbf{E}[G^2] = (1 - \gamma^{2d_i})\mathbf{E}[G^2|\overline{\mathbf{V}}] + \gamma^{2d_i}\mathbf{E}[G^2|\mathbf{V}]$$

we have $\mathbf{E}[G(y,z)^2|\overline{\mathbf{V}}] \le 1/(1 - \gamma^{2d_i})$. Therefore

$$\rho(\mathcal{X}^i, \mathcal{Y}^i \times \mathcal{Z}^i; \mathcal{H}_\delta^*(d_i)) = (1 - \gamma^{2d_i})\mathbf{E}[f(x)G(y,z)|\overline{\mathbf{V}}] \le (1 - \gamma^{2d_i})\sqrt{\mathbf{E}[f^2|\overline{\mathbf{V}}]\mathbf{E}[G(y,z)^2|\overline{\mathbf{V}}]}$$

$$\le (1 - \gamma^{2d_i})\sqrt{\mathbf{E}[f^2]/(1 - \gamma^{2d_i})} \le \sqrt{1 - \gamma^{2d_i}} \le (1 - \gamma^{2d}/2) \quad (5.17)$$

We have shown $\rho(\mathcal{X}^i, \mathcal{Y}^i \times \mathcal{Z}^i; \mathcal{H}_\delta^*(d_i)) \le (1 - \gamma^{2d}/2)$. By applying Proposition 5.3.11,

$$\rho(\prod_{i=1}^{R_1} \mathcal{X}^i, \prod_{i=1}^{R_1} \mathcal{Y}^i \times \mathcal{Z}^i; \mathcal{T}_e^*) \le 1 - \gamma^{2d}/2.$$

Notice that

$$\mathop{\mathbf{E}}_{\mathcal{T}_\beta}[f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)] - \mathop{\mathbf{E}}_{\mathcal{T}_e}[T_{1-\gamma'}f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z)]$$

$$= \mathop{\mathbf{E}}_{\mathcal{T}_e^*}[f(x)g(y^*)g(z^*)] - \mathop{\mathbf{E}}_{\mathcal{T}_e^*}[T_{1-\gamma'}f(x)g(y^*)g(z^*)] = \mathop{\mathbf{E}}_{\mathcal{T}_e^*}[g(y^*)g(z^*)(I - T_{1-\gamma'})f(x)] \quad (5.18)$$

Now we can view $g(y^*)g(z^*)$ as a whole. Similar to the proof of (5.15) and (5.13), we can bound the $|(5.18)|$ by β. Overall, we prove that

$$|\mathbf{E}[f(x)g(y)g(z)] - \mathbf{E}[T_{1-\gamma^*}(f(x)T_{1-\gamma}g(y)T_{1-\gamma}g(z))]| \le 3\beta.$$

$\square$

Above proof is similar to the setup of Lemma 6.2 in [115]. The main reason we can not use it directly is our distribution has $\rho(\mathcal{X}, \mathcal{Y} \times \mathcal{Z}; \mathcal{T}_e) = 1$. In addition, the Bonami-Beckner Operator we need to use is different from the one used in that Lemma.

5.5 Probability Space

Proof of Lemma 5.3.8:

Proof. Let us first prove a graph property of $\mathcal{H}_\delta$.

Lemma 5.5.1. *Define a bipartite graph $G(\mathcal{X} \times \mathcal{Y}, \mathcal{Z})$ as follows: if*

$$\mathbf{Pr}_{\mathcal{H}_\delta}((x, y_1, ..y_d, z_1..z_d) > 0,$$

$(x, y_1, ...y_d), (z_1, ..z_d)$ are in G and there is an edge between them. Then G is connected.

Proof. This is a bipartite graph with no isolated nodes since nodes are included only if they are on some edge. Notice that $\mathcal{H}_\delta = (1 - \delta)\mathcal{H} + \delta\mathcal{N}$. By definition of $\mathcal{N}$, we know $(z_1, ..z_d)$ has edge with $(x_i = z_1, y_1 = z_1, y_2 = -x_1 z_2..y_d = -x_1 z_d)$. And by definition of $\mathcal{H}$, $(x_1 = z_1, y_1 = z_1, y_2 = -x_1 z_2 ... y_d = -x_1 z_d)$ has edge with $(-1, z_2...z_d)$. Therefore, $(z_1, ..z_d)$ is

connected with $(-1, z_2...z_d)$ if they are not the same node. Essentially, $(z_1,..z_d)$ has edge with the node that set one of the z_i to -1: $(z_1, z_2, ..z_{i-1}, z_i = -1, z_{i+1}, ..z_d)$ if they are not the same nodes. Notice that $(1,...1)$ is in $\mathcal{Z}$ and it can reach any nodes in $\mathcal{Z}$ by setting some coordinates to -1. Also, every node in $\mathcal{X} \times \mathcal{Y}$ is connected with some node in $\mathcal{Z}$. The graph is therefore fully connected. $\qquad \square$

Since $\mathcal{H}_\delta$ is connected. The smallest probability event in $\mathcal{H}_\delta$ is $\frac{\delta}{d2^d}$. By applying Lemma 2.9 in [115], we know: $\rho(\mathcal{X} \times \mathcal{Y}, \mathcal{Z}; \mathcal{H}_\delta) \le 1 - \frac{\delta^2}{2^{2d+1}d^2}$. $\qquad \square$

Proof of Lemma 10.5.1

Proof. Notice that $\mathcal{F}_\delta$, $\mathcal{F}$ and $\mathcal{N}$'s marginal distributions on $\mathcal{X}$ are all uniform.

$$(\mathcal{X}, \mathcal{Y} \times \mathcal{Z}; \mathcal{F}_\delta) = \sup_{\substack{f, G, \mathbf{E}[f] = \mathbf{E}[G] = 0 \\ \mathbf{E}[f^2] = \mathbf{E}[G^2] = 1}} \mathbf{E}_{\mathcal{F}_\delta}[fG] = (1-\delta)\sup_{f,G}(\mathbf{E}_{\mathcal{F}}[fG] + \delta \mathbf{E}_{\mathcal{N}}[fG]) =$$

$$(1-\delta)\mathbf{E}_{\mathcal{F}}[f]\mathbf{E}[G] + \delta \mathbf{E}_{\mathcal{N}}[fG] \le 0 + \delta \sqrt{\mathbf{E}_{\mathcal{N}}[|f|^2]\mathbf{E}_{\mathcal{N}}[G^2]}. \quad (5.19)$$

We know $\mathbf{E}_{\mathcal{N}}[f^2] = \mathbf{E}_{\mathcal{F}_\delta}[f^2] = 1$. Also notice that $1 = \mathbf{E}_{\mathcal{F}_\delta}[G^2] = (1-\delta)\mathbf{E}_{\mathcal{F}}[G^2] + \delta \mathbf{E}_{\mathcal{N}}[G^2] \ge \delta \mathbf{E}_{\mathcal{N}}[G^2]$. We have $\mathbf{E}_{\mathcal{N}}[G^2] \le 1/\delta$ and therefore we can bound (5.19) by $\sqrt{\delta}$. $\qquad \square$

Proof of Lemma 5.3.9

Proof. We know $\mathcal{X}, \mathcal{Y}$ are independent in $\mathcal{H}$. Also by definition $\mathcal{H}_\delta = (1-\delta)\mathcal{H} + \delta \mathcal{N}$. Notice that the marginal distributions of both $\mathcal{H}_\delta$ and $\mathcal{N}$ on $\mathcal{Y}$ and $\mathcal{Z}$ are the same (uniform distribution), we have

$$\rho(\mathcal{X}, \mathcal{Y}; \mathcal{H}_\delta) = \sup_{\substack{f, g, \mathbf{E}[f] = \mathbf{E}[g] = 0 \\ \mathbf{E}[f^2] = \mathbf{E}[g^2] = 1}} \mathbf{E}_{\mathcal{H}_\delta}[f(x)g(y)] = (1-\delta)\sup_{f,g}(\mathbf{E}_{\mathcal{H}}[f(x)g(y)] + \delta \mathbf{E}_{\mathcal{N}}[f(x)g(y)])$$

$$= \delta \mathbf{E}_{\mathcal{N}}[fg] \le \delta \sqrt{\mathbf{E}_{\mathcal{N}}[f^2]\mathbf{E}_{\mathcal{N}}[g^2]} = \delta.$$

$\qquad \square$

5.6 Matrix Theory

Lemma 5.6.1. *A_i and B_i are $m_i \times m_i$ matrix. And we know A_i, $A_i + B_i$ and $A_i - B_i$ are positive matrices. Then for any n, $\bigotimes_{i=1}^n A_i - \bigotimes_{i=1}^n B_i$ and $\bigotimes_{i=1}^n A_i + \bigotimes_{i=1}^n B_i$ are both positive matrices.*

Proof. We prove the claim by induction on n.

Base: The base case $n = 1$ is already known fact.

Induction step: Suppose it is hold for $n = k$, for $n = k+1$, we know

$$2\left(\bigotimes_{i=1}^{k+1} A_i - \bigotimes_{i=1}^{k+1} B_i\right) = (A_{k+1} + B_{k+1})(\bigotimes_{i=1}^{k} A_i - \bigotimes_{i=1}^{k} B_i) + (A_{k+1} - B_{k+1})(\bigotimes_{i=1}^{k} A_i + \bigotimes_{i=1}^{k} B_i).$$

By Induction, $(A_{k+1} + B_{k+1}), (\otimes_{i=1}^{k} A_i - \otimes_{i=1}^{k} B_i), (A_{k+1} - B_{k+1}), (\otimes_{i=1}^{k} A_i + \otimes_{i=1}^{k} B_i)$ are all positive matrices. Therefore, $\otimes_{i=1}^{k+1} A_i - \otimes_{i=1}^{k+1} B_i$ is positive.

By a similar argument, since we know

$$2\left(\bigotimes_{i=1}^{k+1} A_i + \bigotimes_{i=1}^{k+1} B_i\right) = (A_{k+1} + B_{k+1})(\bigotimes_{i=1}^{k} A_i + \bigotimes_{i=1}^{k} B_i) + (A_{k+1} - B_{k+1})(\bigotimes_{i=1}^{k} A_i - \bigotimes_{i=1}^{k} B_i),$$

we have that $\otimes_{i=1}^{k+1} A_i + \otimes_{i=1}^{k+1} B_i$ is also positive. $\qquad\square$

Chapter 6

SDP gaps for variants of Label Cover

6.1 Introduction

In this chapter, we mainly study the SDP gap for 2-to-1 LABEL-COVER (as well some other variants).

6.1.1 Motivations

The main reason to study d-to-1 LABEL-COVER is to understand the approximability of satisfiable instance. For example, in Chapter 5, we study the applications d-to-1 conjecture on the 3-CSP examples. Another hardness result for satisfiable instance is the graph coloring problem: an important result is due to Dinur, Mossel, and Regev [43] who used the "2-to-1 Conjecture" as well as the "2-to-2 Conjecture", and the "α-Constraint Conjecture". (These conjectures will be described formally in Section 6.3.) An instance of LABEL-COVER with α-constraints was also implicit in the result of Dinur and Safra [44], on the hardness of approximating minimum vertex cover.

6.1.2 Statements of the Conjectures

We have already defined d-to-1 LABEL-COVER in Section 2.5. 2-to-1 LABEL-COVER is the special case of $d = 2$. We restate its definition here:

Definition 1. *A mapping* $\pi : [R] \to [R]$ *is said to be* 2-*to-*1 *if for each element* $j \in [R]$ *we have* $|\pi^{-1}(j)| \leq 2$. *A* LABEL-COVER *instance is said to be* 2-*to-*1 *if all its constraints are* 2-*to-*1 *projections.*

Conjecture 1. *[97]* (**2-to-1 Conjecture**) *For any* $\delta > 0$*, for* 2-*to-*1 LABEL-COVER *with alphabet size large enough (while still being a constant depending only on* δ*), it is NP-hard to* $(1, 1 - \delta)$-*approximate the problem.*

6.1.3 Evidence for and against

Despite significant work, the status of the UGC— as well as the 2-to-1, 2-to-2, and α-Constraint Conjectures — is unresolved. Towards *disproving* the conjectures, the best algorithms known are due to Charikar, Makarychev, and Makarychev [28]. Using somewhat strong SDP relaxations, those authors gave polynomial-time SDP-rounding algorithms which achieve:

- Value $K^{-\epsilon/(2-\epsilon)}$ (roughly) for Unique LABEL-COVER instances with SDP value $1 - \epsilon$ over alphabets of size K.

- Value $K^{-3+2\sqrt{2}-\epsilon}$ for 2-to-1 LABEL-COVER instances with SDP value $1 - \Theta(\epsilon)$ over alphabets of size K.

The best evidence in *favor* of the UGC is probably the existence of strong SDP gaps. The first such gap was given by Khot and Vishnoi [107]: they constructed a family of Unique LABEL-COVER instances over alphabet size K with SDP value $1 - \epsilon$ and integral optimal value $K^{-\Theta(\epsilon)}$. In addition to roughly matching the CMM algorithm, the Khot–Vishnoi gaps have the nice property that they even hold with Triangle Inequality constraints added into

the SDP. Even stronger SDP gaps for UGC were obtained recently by Raghavendra and Steurer [127].

Standing in stark contrast to this is the situation for the 2-to-1 Conjecture and related variants with perfect completeness. Prior to this work, there were *no known* SDP gap families for these problems with SDP value 1 and integral optimal value tending to 0 with the alphabet size. Indeed, there was hardly any evidence for these conjectures, beyond the fact that the algorithm in [28] failed to disprove them.

6.1.4 SDP gaps as a reduction tool

In addition to being the only real evidence towards the validity of the UGC, SDP gaps for UNIQUE-GAMES have served another important role: they serve as starting points for strong SDP gaps for other important optimization problems. A notable example of this comes in the work of Khot and Vishnoi [107] who used the UG gap instance to construct a super-constant integrality gap for the Sparsest Cut-SDP with triangle inequalities, thereby refuting the Goemans-Linial conjecture that the gap was bounded by $O(1)$. They also used this approach to show that the integrality gap of the Max-Cut SDP remains 0.878 when triangle inequalities are added. Indeed the approach via UNIQUE-GAMES remains the *only known* way to get such strong gaps for Max Cut. Recently, even stronger gaps for Max-Cut were shown using this framework in [96, 127]. Another example of a basic problem for which a SDP gap construction is only known via the reduction from UNIQUE-GAMES is Maximum Acyclic Subgraph [67].

In view of these results, it is fair to say that SDP gaps for UNIQUE-GAMES are significant unconditionally, regardless of the truth of the UGC. Given the importance of 2-to-1 and related conjectures in reductions to satisfiable CSPs and other problems like coloring where perfect completeness is crucial, SDP gaps for 2-to-1 LABEL-COVER and variants are worthy of study even beyond the motivation of garnering evidence towards the associated conjectures on their inapproximability.

6.2 Our Results

LABEL-COVER admits a natural SDP relaxation (see Figure 6.1). In this work, we show the following results on the limitations of the basic SDP relaxation for LABEL-COVER instances with 2-to-1, 2-to-2, and α constraints:

- There is an instance of 2-to-2 LABEL-COVER with alphabet size K and optimum value $O(1/\log K)$ on which the SDP has value 1.

- There are instances of 2-to-1 and α-constraint LABEL-COVER with alphabet size K and optimum value $O(1/\sqrt{\log K})$ on which the SDP has value 1.

In both cases the instances have size $2^{\Omega(K)}$.

We note that if we only require the SDP value to be $1 - \epsilon$ instead of 1, then integrality gaps for all these problems easily follow from gaps from UNIQUE-GAMES, constructed by Khot and Vishnoi [107] (by duplicating labels appropriately to modify the constraints). However, the motivation behind these conjectures is applications where it is important that the completeness is 1. Another difference between the 2-to-1 LABEL-COVER and the

Unique LABEL-COVER is the fact that for 2-to-1 instances, it is consistent with known algorithmic results of [28] that Opt be as low as K^{-c} for some $c > 0$ independent of ϵ, when the SDP value is $1 - \epsilon$. It is an interesting question if Opt can be indeed this low even when the SDP value is 1. Our constructions do not address this question, as we only show $OPT = O(1/\sqrt{\log K})$.

We also point out that our integrality gaps are for special cases of the LABEL-COVER problem where the constraints can be expressed as difference equations over $\mathbb{F}_2$-vector spaces. For example, for 2-to-2 LABEL-COVER, each constraint ϕ_e is of the form $x - y \in \{\alpha, \alpha + \gamma\}$ where $\alpha, \gamma \in \mathbb{F}_2^k$ are constants. However, treating the coordinates $(x_1, \ldots, x_k)$ and $(y_1, \ldots, y_k)$ as separate Boolean variables, and introducing an auxiliary Boolean variable z_e for the constraint, we can re-write it as a conjunction of linear equations over $\mathbb{F}_2$:

$$\bigwedge_{i=1}^{k} \left(x_i - y_i - z_e \cdot \gamma_i = \alpha_i \right).$$

Here $x_i, y_i, \alpha_i, \gamma_i$ denote the ith coordinates of the corresponding vectors. Then the problem of deciding whether the instance is completely satisfiable ($OPT = 1$) or not ($OPT < 1$), reduces to deciding whether the system of linear equations as above, is satisfiable. This can be easily done in polynomial time.

Despite this tractability, the SDPs fail badly to decide satisfiability. This situation is similar to the very strong SDP gaps known for problems such as 3-XOR (see [131], [139]), for which deciding complete satisfiability is easy.

6.3 Preliminaries and Notation

6.3.1 2-to-1, 2-to-2 and α LABEL-COVER Problems

Recall that a LABEL-COVER instance $\mathscr{L}$ is defined by a tuple $(U, V, E, P, R_1, R_2, \Pi)$. Here U and V are the two vertex sets of a bipartite graph and E is the set of edges between U and V. P is an explicitly given probability distribution on E. R_1 and R_2 are integers with $1 \le R_1 \le R_2$. Π is a collection of "projections", one for each edge: $\Pi = \{\pi_e : [R_2] \to [R_1] \mid e \in E\}$.

Here an edge (u, v) is satisfied by a assignment L if $L(u) = \pi_{u,v}(L(v))$. The constraint on each edge is a projection. As for the 2-to-2 and α LABEL-COVER, the constraint on each edge is called 2-to-2 and α defined as follows.

Definition 2. *A constraint $\pi \subseteq \{1, \ldots, 2R\}^2$ is said to be a* 2-to-2 *constraint if there are two permutations $\sigma_1, \sigma_2 : \{1, \ldots, 2R\} \mapsto \{1, \ldots, 2R\}$ such that $(i, j) \in \pi$ if and only if $(\sigma_1(i), \sigma_2(j)) \in T$ where*

$$T := \{(2l - 1, 2l - 1), (2l - 1, 2l), (2l, 2l - 1), (2l, 2l)\}_{l=1}^{R}$$

A LABEL-COVER instance is said to be 2-to-2 if all its constraints are 2-to-2 constraints.

A constraint $\pi \subseteq \{1, \ldots, 2R\}^2$ is said to be an α-constraint if there are two permutations $\sigma_1, \sigma_2 : \{1, \ldots, 2R\} \mapsto \{1, \ldots, 2R\}$ such that $(i, j) \in \pi$ if and only if $(\sigma_1(i), \sigma_2(i)) \in T'$ where

$$T' := \{(2l - 1, 2l - 1), (2l - 1, 2l), (2l, 2l - 1)\}_{l=1}^{R}$$

A LABEL-COVER instance is said to be α if all its constraints are α constraints.

$$\begin{aligned}
\text{maximize} \quad & \mathbf{E}_{e=(u,v)\in E}\left[\sum_{(i,j)\in\pi_e}\left\langle\mathbf{z}_{(u,i)},\mathbf{z}_{(v,j)}\right\rangle\right] \\
\text{subject to} \quad & \sum_{i\in[R]}\left\|\mathbf{z}_{(v,i)}\right\|^2 = 1 && \forall\, v\in V \\
& \left\langle\mathbf{z}_{(v,i)},\mathbf{z}_{(v,j)}\right\rangle = 0 && \forall\, i\neq j\in[R], v\in V
\end{aligned}$$

Figure 6.1: SDP for LABEL-COVER

Conjecture 2. *[43] (**2-to-2 Conjecture***) For any $\delta > 0$, it is NP-hard to decide whether a 2-to-2* LABEL-COVER *instance $\mathscr{L}$ has:-*

- $OPT(\mathscr{L}) = 1$
- $OPT(\mathscr{L}) \leq \delta$

It was shown in [43] that the 2-to-2 Conjecture is no stronger than the 2-to-1 Conjecture.

Conjecture 3. *[43] (α **Conjecture***) For any $\delta > 0$, it is NP-hard to decide whether a α* LABEL-COVER *instance $\mathscr{L}$ has:-*

- $OPT(\mathscr{L}) = 1$
- $OPT(\mathscr{L}) \leq \delta$

By abuse of notation, for the 2-to-2 or α LABEL-COVER, we use $\pi_{u,v}$ to denote the constraint on an edge (u,v) and it an edge is satisfied if $(L(u),L(v))\in\pi_{u,v}$. For the case of 2-to-1 where each $\pi_{u,v}$ is a projection relationship, we use $(L(u),L(v))\in\pi_{u,v}$ as a equivalent statement of $\pi_{u,v}(L(v)) = L(u)$.

In Figure 6.1, we write down a natural SDP relaxation for the LABEL-COVER problem. The relaxation is over the vector variables $\mathbf{z}_{(v,i)}$ for every vertex $v\in V$ and label $i\in[R]$.

6.3.2 Fourier Analysis

We will use the Fourier Analysis on $\mathbb{F}_2^k$ where $\mathbb{F}_2 = \{0,1\}$. [1]

Let $\mathcal{V} := \{f : \mathbb{F}_2^k \to \mathscr{R}\}$ denote the vector-space of all real functions on $\mathbb{F}_2^k$, where addition is defined as point-wise addition over $\mathbb{F}_2$ (or . We always think of $\mathbb{F}_2^k$ as a probability space under the uniform distribution, and therefore use notation such as $\|f\|_p := \mathbf{E}_{x\in\mathbb{F}_2^k}[|f(x)|^p]$. For $f, g \in \mathscr{F}$, we also define the inner product $\langle f, g\rangle := \mathbf{E}[f(x)g(x)]$.

For any $\alpha \in \mathbb{F}_2^k$ the $\chi_\alpha \in \mathscr{F}$ as $\chi_\alpha(x) := (-1)^{\alpha\cdot x}$, $\forall x\in\mathbb{F}_2^k$. The Fourier characters form an orthonormal basis for $\mathcal{V}$ with respect to the above inner product, hence every function $f\in\mathcal{V}$ has a unique representation as $f = \sum_{\alpha\in\mathbb{F}_2^k}\widehat{f}(\alpha)\chi_\alpha$, where the Fourier coefficient $\widehat{f}(\alpha) := \langle f, \chi_s\rangle$.

We also sometimes identify each α with the set $S_\alpha = \{i|\alpha_i = 1\}$ and denote the Fourier coefficients as $\widehat{f}(S)$. We use the notation $|\alpha|$ for $|S_\alpha|$, the number of coordinates where α

---

[1]The reader may notice that in other Chapters, we are using Harmonical Analysis over $\{-1,1\}^n \to \mathbb{R}$. The switch (to $\mathbb{F}_2$) is mainly for the notational convenience in this Chapter.

is 1.

We need the following result due to Talagrand ("Proposition 2.3" in [135]), proven using hypercontractivity methods:

Theorem 6.3.1. *Suppose $F : \mathbb{F}_2^k \to \mathbb{R}$ has $\mathbf{E}[F] = 0$. Then*

$$\sum_{\alpha \in \mathbb{F}_2^k \setminus \{0\}} \widehat{F}(\alpha)^2 / |\alpha| = O\left(\frac{\|F\|_2^2}{\ln(\|F\|_2/(e\|F\|_1))}\right).$$

More precisely, we will need the following easy corollary:

Corollary 6.3.2. *If $F : \mathbb{F}_2^k \to \{0,1\}$ has mean $1/K$, then*

$$\widehat{F}(0)^2 + \sum_{\alpha \in \mathbb{F}_2^k \setminus \{0\}} \widehat{F}(\alpha)^2 / |\alpha| = O(1/(K \log K))$$

Proof. We have $\widehat{F}(0)^2 = \mathbf{E}[F]^2 = 1/K^2 \le O(1/(K \log K))$, so we can disregard this term. As for the sum, we apply Theorem 6.3.1 to the function $F' = F - 1/K$, which has mean 0 as required for the theorem. It is easy to calculate that $\|F'\|_2 = \Theta(1/\sqrt{K})$ and $\|F'\|_1 = \Theta(1/K)$, and so the result follows. $\qquad \square$

6.4 Integrality Gap for 2-to-2 Games

We first give an integrality gap for LABEL-COVER with 2-to-2 constraints. The instance for 2-to-1 LABEL-COVER will be an extension of the one below. In fact, our analysis of Opt in the 2-to-1 case will follow simply by reducing it to the analysis of Opt for the 2-to-2 instance below.

The vertex set V in our instance is same as the vertex set of the UNIQUE-GAMES integrality gap instance constructed in [107]. Let $\mathscr{F} := \{f : \mathbb{F}_2^k \mapsto \mathbb{F}_2\}$ denote the family of all Boolean functions on $\mathbb{F}_2^k$. For f, $g \in \mathscr{F}$, define the product fg as $(fg)(x) := f(x)g(x)$. Consider the equivalence relation $\sim$ on $\mathscr{F}$ defined as $f \sim g \Leftrightarrow \exists \alpha \in \mathbb{F}_2^k$ s.t. $f \equiv g\chi_\alpha$. This relation partitions $\mathscr{F}$ into equivalence classes $\mathscr{P}_1, \ldots, \mathscr{P}_n$, with $n := 2^K/K$. The vertex set V consists of the equivalence classes $\{\mathscr{P}_i\}_{i \in [n]}$. We denote by $[\mathscr{P}_i]$ the lexicographically smallest function in the class $\mathscr{P}_i$ and by $\mathscr{P}_f$, the class containing f.

We take the label set to be of size K and identify $[K]$ with $\mathbb{F}_2^k$ in the obvious way. For each tuple of the form (γ, f, g) where $\gamma \in \mathbb{F}_2^k \setminus \{0\}$ and $f, g \in \mathscr{F}$ are such that $(1 + \chi_\gamma)f \equiv (1 + \chi_\gamma)g$, we add a constraint $\pi_{(\gamma, f, g)}$ between the vertices $\mathscr{P}_f$ and $\mathscr{P}_g$. Note that the condition on f and g is equivalent to saying that $\chi_\gamma(x) = 1 \implies f(x) = g(x)$. If $f = [\mathscr{P}_f]\chi_\alpha$ and $g = [\mathscr{P}_g]\chi_\beta$ and if $A : [n] \to \mathbb{F}_2^k$ denotes the labeling, the relation $\pi_{(\gamma, f, g)}$ is defined as

$$(A(\mathscr{P}_f), A(\mathscr{P}_g)) \in \pi_{(\gamma, f, g)} \quad \Leftrightarrow \quad (A(\mathscr{P}_f) + \alpha) - (A(\mathscr{P}_g) + \beta) \in \{0, \gamma\}.$$

Note that for any $\omega \in \mathbb{F}_2^k$, the constraint maps the labels $\{\omega, \omega + \gamma\}$ for $\mathscr{P}_f$ to the labels $\{\omega + \alpha - \beta, \omega + \alpha - \beta + \gamma\}$ for $\mathscr{P}_g$ in a 2-to-2 fashion. We denote the set of all constraints by π. We remark that, as in [107], our integrality gap instances contain multiple constraints on each pair of vertices.

6.4.1 SDP Solution

We give below a set of feasible vectors $\mathbf{z}_{(\mathscr{P}_i, \alpha)} \in \mathbb{R}^K$ for every equivalence class $\mathscr{P}_i$ and every label α, achieving SDP value 1. Identifying each coordinate with an $x \in \mathbb{F}_2^k$, we define the vectors as

$$\mathbf{z}_{(\mathscr{P}_i, \alpha)}(x) := \frac{1}{K}([\mathscr{P}_i]\chi_\alpha)(x).$$

It is easy to check that $\left\|\mathbf{z}_{(\mathscr{P}_i, \alpha)}\right\|^2 = 1/K$ for each of the vectors, which satisfies the first constraint. Also, $\mathbf{z}_{(\mathscr{P}_i, \alpha)}$ and $\mathbf{z}_{(\mathscr{P}_i, \beta)}$ are orthogonal for $\alpha \neq \beta$ since

$$\left\langle \mathbf{z}_{(\mathscr{P}_i, \alpha)}, \mathbf{z}_{(\mathscr{P}_i, \beta)} \right\rangle = \frac{1}{K^2} \left\langle [\mathscr{P}_i]\chi_\alpha, [\mathscr{P}_i]\chi_\beta \right\rangle = \frac{1}{K^2} \left\langle \chi_\alpha, \chi_\beta \right\rangle = 0$$

using the fact that $[\mathscr{P}_i]^2 = 1$. The following claim proves that the solution achieves SDP value 1.

Claim 6.4.1. *For any edge e indexed by a tuple (γ, f, g) with $f(1 + \chi_\gamma) \equiv g(1 + \chi_\gamma)$, we have*

$$\sum_{\omega_1, \omega_2 \in \pi_{(\gamma, f, g)}} \left\langle \mathbf{z}_{(\mathscr{P}_f, \omega_1)}, \mathbf{z}_{(\mathscr{P}_g, \omega_2)} \right\rangle = 1$$

Proof. Let $f \equiv [\mathscr{P}_f]\chi_\alpha$ and $g \equiv [\mathscr{P}_g]\chi_\beta$. Then, $(\omega_1, \omega_2) \in \pi_e$ iff $(\omega_1 + \alpha) - (\omega_2 + \beta) \in \{0, \gamma\}$. Therefore, the above quantity equals (divided by 2 to account for double counting of ω)

$$\frac{1}{2} \cdot \sum_\omega \left(\left\langle \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha)}, \mathbf{z}_{(\mathscr{P}_g, \omega+\beta)} \right\rangle + \left\langle \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha+\gamma)}, \mathbf{z}_{(\mathscr{P}_g, \omega+\beta)} \right\rangle \right.$$
$$\left. + \left\langle \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha)}, \mathbf{z}_{(\mathscr{P}_g, \omega+\beta+\gamma)} \right\rangle + \left\langle \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha+\gamma)}, \mathbf{z}_{(\mathscr{P}_g, \omega+\beta+\gamma)} \right\rangle \right)$$
$$= \frac{1}{2} \sum_\omega \left\langle \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha)} + \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha+\gamma)}, \mathbf{z}_{(\mathscr{P}_f, \omega+\beta)} + \mathbf{z}_{(\mathscr{P}_f, \omega+\beta+\gamma)} \right\rangle \tag{6.1}$$

However, for each ω, we have $\mathbf{z}_{(\mathscr{P}_f, \omega+\alpha)} + \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha+\gamma)} = \mathbf{z}_{(\mathscr{P}_f, \omega+\beta)} + \mathbf{z}_{(\mathscr{P}_f, \omega+\beta+\gamma)}$, since for all coordinates x,

$$\mathbf{z}_{(\mathscr{P}_f, \omega+\alpha)}(x) + \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha+\gamma)}(x) = \frac{1}{K}([\mathscr{P}_f]\chi_{\omega+\alpha}(x) + [\mathscr{P}_f]\chi_{\omega+\alpha+\gamma}(x))$$
$$= \frac{1}{K}(f(x) + f\chi_\gamma)\chi_\omega(x)$$
$$= \frac{1}{K}(g(x) + g\chi_\gamma)\chi_\omega(x)$$
$$= \frac{1}{K}([\mathscr{P}_g]\chi_{\omega+\beta}(x) + [\mathscr{P}_g]\chi_{\omega+\beta+\gamma}(x))$$
$$= \mathbf{z}_{(\mathscr{P}_f, \omega+\beta)}(x) + \mathbf{z}_{(\mathscr{P}_f, \omega+\beta+\gamma)}(x).$$

This completes the proof as the value of (6.1) then becomes

$$\frac{1}{2} \sum_\omega \left\| \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha)} + \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha+\gamma)} \right\|^2 = \frac{1}{2} \sum_\omega \left(\left\| \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha)} \right\|^2 + \left\| \mathbf{z}_{(\mathscr{P}_f, \omega+\alpha+\gamma)} \right\|^2 \right) = 1$$

$\square$

6.4.2 Soundness

We now prove that any labeling of the instance described above, satisfies at most $O(1/\log K)$ fraction of the constraints. Let $A : V \to \mathbb{F}_2^k$ be a labeling of the vertices. We extend it to a labeling of all the functions in $\mathscr{F}$ by defining $A([\mathscr{P}_i]\chi_\alpha) := A(\mathscr{P}_i) + \alpha$.

For each $\alpha \in \mathbb{F}_2^k$, define $A_\alpha : \mathscr{F} \to \{0,1\}$ to be the indicator that A's value is α. By definition, the fraction of constraints satisfied by the labeling A is

$$
\begin{aligned}
\text{val}(A) \;&=\; \mathop{\mathbf{E}}_{(\gamma,f,g)\in\pi}\left[\sum_{\alpha\in\mathbb{F}_2^k} A_\alpha(f)(A_\alpha(g)+A_{\alpha+\gamma}(g))\right] \\
&=\; \mathop{\mathbf{E}}_{(\gamma,f,g)\in\pi}\left[\sum_{\alpha\in\mathbb{F}_2^k} A_\alpha(f)(A_\alpha(g)+A_\alpha(g\chi_\gamma))\right] \;=\; 2\cdot \mathop{\mathbf{E}}_{(\gamma,f,g)\in\pi}\left[\sum_{\alpha\in\mathbb{F}_2^k} A_\alpha(f)(A_\alpha(g)\right] (6.2)
\end{aligned}
$$

where the last equality used the fact that for every tuple $(\gamma,f,g) \in \pi$, we also have $(\gamma,f,g\chi_\gamma) \in \pi$.

Note that the extended labeling $A : \mathscr{F} \to \mathbb{F}_2^k$ takes on each value in $\mathbb{F}_2^k$ an equal number of times. Hence

$$
\mathbf{E}_f[A_\alpha(f)] = \mathbf{Pr}_f[A(f) = \alpha] = 1/K \quad \text{for each } \alpha \in \mathbb{F}_2^k. \tag{6.3}
$$

For our preliminary analysis, we will use only this fact to show that for any $\alpha \in \mathbb{F}_2^k$ it holds that

$$
\mathop{\mathbf{E}}_{(\gamma,f,g)\in\pi}[A_\alpha(f)A_\alpha(g)] \le O(1/(K\log K)). \tag{6.4}
$$

It will then follow that the soundness (6.2) is at most $O(1/\log K)$. Although this tends to 0, it does so only at a rate proportional to the logarithm of the alphabet size, which is $K = 2^k$.

Beginning with the left-hand side of (6.4), let's write $F = A_\alpha$ for simplicity. We think of the functions f and g being chosen as follows. We first choose a function $h : \gamma^\perp \to \mathbb{F}_2$. Note that $\gamma^\perp \subseteq \mathbb{F}_2^k$ is the set of inputs where $\chi_\gamma = 1$ and hence $f = g$, and we let $f(x) = g(x) = h(x)$ for $x \in \gamma_\perp$. The values of f and g on the remaining inputs are chosen independently at random. Then

$$
\mathop{\mathbf{E}}_{(\gamma,f,g)\in\pi}[F(f)F(g)] = \mathop{\mathbf{E}}_{\gamma}\mathop{\mathbf{E}}_{h:\gamma^\perp\to\mathbb{F}_2}\big[\mathbf{E}_{f,g|h}[F(f)F(g)]\big] = \mathop{\mathbf{E}}_{\gamma}\mathop{\mathbf{E}}_{h:\gamma^\perp\to\mathbb{F}_2}\big[\mathbf{E}_{f|h}[F(f)]\mathbf{E}_{g|h}[F(g)]\big]. \tag{6.5}
$$

Let us write $P_\gamma F(h)$ for $\mathbf{E}_{f|h}F(f)$, which is also equal to $\mathbf{E}_{g|h}F(g)$. We now use the Fourier expansion of F. Note that the domain here is $\mathbb{F}_2^K$ instead of $\mathbb{F}_2^k$. To avoid confusion with characters and Fourier coefficients for functions on $\mathbb{F}_2^k$, we will index the Fourier coefficients below by sets $S \subseteq \mathbb{F}_2^k$. Given an $f \in V$, we'll write f^S for $\prod_{x\in S} f(x)$ (which is a Fourier character for the domain $\mathbb{F}_2^K$). Now for fixed γ and h,

$$
P_\gamma F(h) = \mathbf{E}_{f|h}[F(f)] = \mathbf{E}_{f|h}\left[\sum_{S\subseteq\mathbb{F}_2^k}\widehat{F}(S)f^S\right] = \sum_{S\subseteq\mathbb{F}_2^k}\widehat{F}(S)\cdot\mathbf{E}_{f|h}[f^S].
$$

The quantity $\mathbf{E}_{f|h}[f^S]$ is equal to h^S if $S \subseteq \gamma^\perp$ as is 0 otherwise. Thus, using the Parseval identity, we deduce that (6.5) equals

$$\underset{\gamma}{\mathbf{E}}\ \underset{h:\gamma^\perp \to \mathbb{F}_2}{\mathbf{E}} \left[(P_\gamma F(h))^2 \right] = \underset{\gamma}{\mathbf{E}} \left[\sum_{S \subseteq \gamma^\perp} \left(\widehat{F}(S) \right)^2 \right] = \sum_{S \subseteq \mathbb{F}_2^k} \mathbf{Pr}_\gamma [S \subseteq \gamma^\perp] \cdot \left(\widehat{F}(S) \right)^2.$$

Recalling that $\gamma \in \mathbb{F}_2^k \setminus \{0\}$ is chosen uniformly, we have that

$$\sum_{S \subseteq \mathbb{F}_2^k} \mathbf{Pr}_\gamma [S \subseteq \gamma^\perp] \cdot \left(\widehat{F}(S) \right)^2 = \sum_{S \subseteq \mathbb{F}_2^k} 2^{-\dim(S)} \cdot \left(\widehat{F}(S) \right)^2,$$

where we are writing $\dim(S) = \dim(\operatorname{span} S)$ for shortness (and defining $\dim(\emptyset) = 0$). For $|S| \geq 1$ we have $\dim(S) \geq \log_2 |S|$ and hence $2^{-\dim(S)} \geq 1/|S|$. Thus

$$\sum_{S \subseteq \mathbb{F}_2^k} 2^{-\dim(S)} \cdot \widehat{F}(S)^2 \leq \widehat{F}(\emptyset)^2 + \sum_{\emptyset \neq S \subseteq \mathbb{F}_2^k} \widehat{F}(S)^2 / |S|.$$

Corollary 6.3.2 shows that this is at most $O(1/(K \log K))$. This completes the proof, as

$$\operatorname{val}(A) = 2 \cdot \sum_{\alpha \in \mathbb{F}_2^k} \mathbf{E}_{(\gamma,f,g) \in \pi}[A_\alpha(f) A_\alpha(g)] \leq 2 \cdot \sum_{\alpha \in \mathbb{F}_2^k} 2^{-\dim(S)} \widehat{A_\alpha}(S)^2 = O(1/\log K).$$

6.5 Integrality Gap for 2-to-1 LABEL-COVER

The instances for 2-to-1 LABEL-COVER are bipartite. We denote such instances as (U, V, E, R_1, R_2, Π) where $R_2 = 2R_1$ denote the alphabet sizes on the two sides. For a bipartite instance, the LABEL-COVER SDP can be written in the following form involving vectors $\mathbf{y}_{(u,i)}$ for each $u \in U, i \in [R_1]$ and vectors $\mathbf{z}_{(v,j)}$ for each $v \in v, j \in [R_2]$.

$$
\begin{array}{lll}
\text{maximize} & \mathbf{E}_{e=(u,v) \in E} \left[\displaystyle\sum_{i \in [R_2]} \left\langle \mathbf{y}_{(u,\pi_e(i))}, \mathbf{z}_{(v,j)} \right\rangle \right] & \\[3ex]
\text{subject to} & \displaystyle\sum_{i \in [R_1]} \left\| \mathbf{y}_{(u,i)} \right\|^2 = 1 & \forall\ u \in U \\[3ex]
& \displaystyle\sum_{i \in [R_2]} \left\| \mathbf{z}_{(v,i)} \right\|^2 = 1 & \forall\ v \in V \\[2ex]
& \left\langle \mathbf{y}_{(u,i)}, \mathbf{y}_{(u,j)} \right\rangle = 0 & \forall\ i \neq j \in [R_1], u \in U \\[1ex]
& \left\langle \mathbf{z}_{(v,i)}, \mathbf{z}_{(v,j)} \right\rangle = 0 & \forall\ i \neq j \in [R_2], v \in V
\end{array}
$$

Figure 6.2: SDP for 2-to-1 games

6.5.1 Gap Instance

As in the case of 2-to-2 games, the set V consists of equivalence classes $\mathscr{P}_1, \ldots, \mathscr{P}_n$, which partition the set of functions $\mathscr{F} = \{f : \mathbb{F}_2^k \to \mathbb{F}_2\}$, according to the equivalence relation $\sim$ defined as $f \sim g \Leftrightarrow \exists \alpha \in \mathbb{F}_2^k \ s.t. f \equiv g\chi_\alpha$. The label set $[R_2]$ is again identified with $\mathbb{F}_2^k$ and is of size $K = 2^k$.

To describe the set U, we further partition the vertices in V according to other equivalence relations. For each $\gamma \in \mathbb{F}_2^k, \gamma \neq 0$, we define an equivalence relation $\cong_\gamma$ on the set $\mathscr{P}_1, \ldots, \mathscr{P}_n$ as

$$\mathscr{P}_i \cong_\gamma \mathscr{P}_j \quad \Leftrightarrow \quad \exists f \in \mathscr{P}_i, g \in \mathscr{P}_j \ \ s.t. \ \ f(1 + \chi_\gamma) \equiv g(1 + \chi_\gamma)$$

This is equivalent to saying

$$\mathscr{P}_i \cong_\gamma \mathscr{P}_j \quad \Leftrightarrow \quad \exists f \in \mathscr{P}_i, g \in \mathscr{P}_j \ \ s.t. \ \ fg(x) = -1 \Rightarrow \chi_\gamma(x) = -1 \forall x \in F_2^k$$

This partitions $\mathscr{P}_1, \ldots, \mathscr{P}_n$ (and hence also the set $\mathscr{F}$) into equivalence classes $\mathscr{Q}_1^\gamma, \ldots, \mathscr{Q}_m^\gamma$. Here $m = 2^{K/2+1}/K$ (this is immediate from the second definition and the fact that $n = 2^K/K$) and the partition is different for each γ. The set U has one vertex for each class of the form $\mathscr{Q}_i^\gamma$ for all $i \in [m]$ and $\gamma \in \mathbb{F}_2^k \setminus \{0\}$. As before, we denote by $[\mathscr{Q}_i^\gamma]$ the lexicographically smallest function in the class $\mathscr{Q}_i^\gamma$, and by $\mathscr{Q}_f^\gamma$ the class under $\cong_\gamma$ containing f. Note that if $f \in \mathscr{Q}_i^\gamma$, then there exists a $\beta \in \mathbb{F}_2^k$ such that $f(1 + \chi_\gamma) \equiv [\mathscr{Q}_i^\gamma]\chi_\beta(1 + \chi_\gamma)$.

The label set R_1 has size $K/2$. For each vertex $\mathscr{Q}_i^\gamma \in U$, we think of the labels as pairs of the form $\{\alpha, \alpha + \gamma\}$ for $\alpha \in \mathbb{F}_2^k$. More formally, we identify it with the space $\mathbb{F}_2^k/\langle \gamma \rangle$. We impose one constraint for every pair of the form (γ, f) between the vertices $\mathscr{P}_f$ and $\mathscr{Q}_f^\gamma$. If $f \equiv [\mathscr{P}_f]\chi_\alpha$ and $f(1 + \chi_\gamma) \equiv [\mathscr{Q}_i^\gamma]\chi_\beta(1 + \chi_\gamma)$, then the corresponding relation $\pi_{(\gamma, f)}$ is defined by requiring that for any labelings $A : V \to [R_2]$ and $B : U \to [R_1]$,

$$(B(\mathscr{Q}_f^\gamma), A(\mathscr{P}_f)) \in \pi_{(\gamma, f)} \quad \Leftrightarrow \quad A(\mathscr{P}_f) + \alpha \in B(\mathscr{Q}_f^\gamma) + \beta.$$

Here, if $B(\mathscr{Q}_f^\gamma)$ is a pair of the form $\{\omega, \omega + \gamma\}$, then $B(\mathscr{Q}_f^\gamma) + \beta$ denotes the pair $\{\omega + \beta, \omega + \gamma + \beta\}$.

6.5.2 SDP Value

As before, we give a set of vectors $\mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\alpha, \alpha + \gamma\})}$ and $\mathbf{z}_{(\mathscr{P}_i, \alpha)}$ in $\mathbb{R}^K$, identifying each coordinate with an $x \in \mathbb{F}_2^k$. We define the vectors as

$$\mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\alpha, \alpha + \gamma\})}(x) := \frac{1}{K}\left([\mathscr{Q}_i^\gamma]\chi_\alpha(1 + \chi_\gamma)\right)(x) \qquad \text{and} \qquad \mathbf{z}_{(\mathscr{P}_i, \alpha)}(x) := \frac{1}{K}\left([\mathscr{P}_f]\chi_\alpha\right)(x).$$

We have already shown that $\langle \mathbf{z}_{(\mathscr{P}_i, \alpha)}, \mathbf{z}_{(\mathscr{P}_i, \beta)} \rangle = 0$ for $\alpha \neq \beta$ and $\|\mathbf{z}_{(\mathscr{P}_i, \alpha)}\|^2 = 1/K$. It again follows by the orthogonality of characters that for disjoint pairs $\{\alpha, \alpha + \gamma\}$ and $\{\beta, \beta + \gamma\}$, the vectors $\mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\alpha, \alpha + \gamma\})}$ and $\mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\beta, \beta + \gamma\})}$ are orthogonal. It is also easy to verify that $\left\|\mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\alpha, \alpha + \gamma\})}\right\|^2 = 2/K$. Hence, the vectors form a feasible solution.

To show that the SDP value is equal to 1, we consider an arbitrary constraint indexed by the pair (γ, f). Let $f \equiv [\mathscr{P}_f]\chi_\alpha$ and $f(1 + \chi_\gamma) \equiv [\mathscr{Q}_i^\gamma]\chi_\beta(1 + \chi_\gamma)$. Then for any $\omega \in \mathbb{F}_2^k$, this

constraint maps the label $\omega + \alpha$ for $\mathscr{P}_f$ to the pair $\{\omega + \beta, \omega + \gamma + \beta\}$ for $\mathscr{Q}_f^\gamma$. Hence, the value of the SDP solution on this constraint is given by

$$\sum_{\omega \in \mathbb{F}_2^k} \left\langle \mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\omega + \beta, \omega + \beta + \gamma\})}, \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega)} \right\rangle$$

We will show that for every ω, $\mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\omega + \beta, \omega + \beta + \gamma\})} = \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega)} + \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega + \gamma)}$. This will complete the proof as the above expression then becomes

$$\sum_{\omega \in \mathbb{F}_2^k} \left\langle \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega)} + \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega + \gamma)}, \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega)} \right\rangle \;=\; \sum_{\omega \in \mathbb{F}_2^k} \left\| \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega)} \right\|^2 \;=\; 1.$$

To show the vector identity, we simply note that for each coordinate x, we have

$$\begin{aligned}
\mathbf{y}_{(\mathscr{Q}_i^\gamma, \{\omega + \beta, \omega + \beta + \gamma\})}(x) = \frac{1}{K} \left([\mathscr{Q}_i^\gamma] \chi_\beta (1 + \chi_\gamma) \right)(x) &\;=\; \frac{1}{K} \left(f(1 + \chi_\gamma) \right)(x) \\
&\;=\; \frac{1}{K} \left([\mathscr{P}_f] \chi_\alpha + [\mathscr{P}_f] \chi_{\alpha + \gamma} \right)(x) \\
&\;=\; \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega)}(x) + \mathbf{z}_{(\mathscr{P}_i, \alpha + \omega + \gamma)}(x).
\end{aligned}$$

6.5.3 Soundness

We now bound the fraction of constraints satisfied by any pair of labelings $A : V \to [K]$ and $B : U \to [K/2]$. Let $\mathbf{1}_{\{\mathscr{E}\}}$ denote the indicator of the event $\mathscr{E}$, and $N(u)$ denote the neighborhood of a vertex $u \in U$. Then, the fraction of constraints satisfied by any assignments A, B, can be bound by an application of Cauchy-Schwarz as

$$\begin{aligned}
\mathrm{val}(A, B) \;&=\; \mathbf{E}_{u \in U} \mathbf{E}_{v \in N(u)} \left[\mathbf{1}_{\{\pi_{uv}(A(v)) = B(u)\}} \right] \\
&\leq\; \left(\mathbf{E}_{u \in U} \left(\mathbf{E}_{v \in N(u)} \left[\mathbf{1}_{\{\pi_{uv}(A(v)) = B(u)\}} \right] \right)^2 \right)^{1/2} \\
&=\; \left(\mathbf{E}_{u \in U} \mathbf{E}_{v_1, v_2 \in N(u)} \left[\mathbf{1}_{\{\pi_{uv_1}(A(v_1)) = B(u) = \pi_{uv_2}(A(v_2))\}} \right] \right)^{1/2} \\
&\leq\; \left(\mathbf{E}_{u \in U} \mathbf{E}_{v_1, v_2 \in N(u)} \left[\mathbf{1}_{\{\pi_{uv_1}(A(v_1)) = \pi_{uv_2}(A(v_2))\}} \right] \right)^{1/2}
\end{aligned}$$

Note that if π_{uv_1} and π_{uv_2} are 2-to-1 projections, then the inner quantity in the last expression denotes the value of a 2-to-2 LABEL-COVER instance, each of whose constraints is defined by two 2-to-1 constraints in the original instance. For the 2-to-1 instance described above, we will show that the inner quantity in fact denotes the fraction of constraints satisfied by A for the 2-to-2 instance described in Section 6.4. This will show that the fraction of constraints satisfied by any assignment in the above 2-to-1 instance can be at most $O(1/\sqrt{\log K})$.

To see this, note that a vertex $u \in U$ and a vertex $v_1 \in V$ can be sampled jointly by picking a pair (γ, f) and taking $u = \mathscr{Q}_f^\gamma$ and $v_1 = \mathscr{P}_f$. Sampling $v_2 \in N(u)$ corresponds to choosing a class $\mathscr{P}_i$ such that for some $\beta \in \mathbb{F}_2^k$ $[\mathscr{P}_i] \chi_\beta (1 + \chi_\gamma) \equiv f(1 + \chi_\gamma)$. Thus, v_2 can be sampled by choosing a random g such that $f(1 + \chi_\gamma) \equiv g(1 + \chi_\gamma)$ and taking $v_2 = \mathscr{P}_g$.

Also, if $f \equiv [\mathscr{P}_f]\chi_{\alpha_1}$ and $g \equiv [\mathscr{P}_g]\chi_{\alpha_2}$, then the constraint $\pi_{uv_1}(A(v_1)) = \pi_{uv_2}(A(v_2))$ simply requires that for some $\omega \in \mathbb{F}_2^k$, $A(\mathscr{P}_f) + \alpha_1$ and $A(\mathscr{P}_g) + \alpha_2$ both lie in the set $\{\omega, \omega + \gamma\}$ and hence

$$(A(\mathscr{P}_f) + \alpha_1) - (A(\mathscr{P}_g) + \alpha_2) \in \{0, \gamma\}.$$

6.6 From 2-to-1 Constraints to α-constraints

In this section we show that any integrality gap instance for 2-to-1 games, with sufficiently many edges, can be converted to an integrality gap instance for games with α-constraints. The SDP we consider for these games is identical to the ones considered before, except for the objective function.

Theorem 6.6.1. *Let $\mathscr{L} = (U, V, E, R, 2R, \pi)$ be a bipartite instance of 2-to-1 LABEL-COVER problem with $OPT(\mathscr{L}) \le \delta$ and SDP value 1. Also, let $|E| \ge 4(|U| + |V|)\log(R)/\epsilon^2$. Then there exists another instance $\mathscr{L}' = (U, V, E, 2R, \pi')$ of LABEL-COVER with α-constraints having SDP value 1 and $OPT(\mathscr{L}') \le \delta + \epsilon + 1/R$.*

Proof. The proof simply follows by adding R "fake" labels for each vertex $u \in U$, and then randomly augmenting the constraints to make them of the required form. In particular, let the new labels we add for each $u \in U$ be $R + 1, \ldots, 2R$. Let $e = (u, v)$ be an edge. Since the constraints in π are 2-to-1 type, there exist permutations $\sigma_{1,e} : [R] \to [R]$ and $\sigma_{2,e} : [2R] \to [2R]$ such that after permuting the labels on each side, the projection π_e maps labels $(2i - 1, 2i)$ to i i.e. $\pi_e(\sigma_{2,e}^{-1}(2i - 1)) = \pi_e(\sigma_{2,e}^{-1}(2i)) = \sigma_{1,e}^{-1}(i)$.

To incorporate the new labels into the constraint, choose a random bijection $\sigma_{1,e}' : \{R + 1, \ldots, 2R\} \to [R]$. We now construct a new permutation $\tilde{\sigma}_{1,e} : [2R] \to [2R]$ as $\tilde{\sigma}_{1,e}(i) = 2\sigma_{1,e}(i) - 1$ if $i \le R$ and $\tilde{\sigma}_{1,e}(i) = 2\sigma_{1,e}'(i)$ if $i > R$ i.e. the new labels are mapped to the even positions $2, 4, \ldots, 2R$ while the others are mapped to the odd positions.

The original 2-to-1 constraints are satisfied by a labeling A iff the pair $(\tilde{\sigma}_{1,e}(A(u)), \sigma_{2,e}(A(v)))$ is of the form $(2i - 1, 2i - 1)$ or $(2i - 1, 2i)$ for some $i \le R$. We augment the constraint by also allowing $(\tilde{\sigma}_{1,e}(A(u)), \sigma_{2,e}(A(v)))$ to be $(2i, 2i - 1)$ for some i. Note that if the constraint is satisfied in this way, then u must get one of the new labels. Also, note that the augmentation is random as we choose the map $\sigma_{1,e}'$ independently at random for each edge e.

Given a vector solution $\{\mathbf{y}_{(u,i)}\}_{u \in U, i \in [R]}$ and $\{\mathbf{z}_{(v,j)}\}_{v \in V, j \in [2R]}$ for π, we leave the vectors $\mathbf{z}_{(v,j)}$ unchanged and for each $u \in U$, take $\mathbf{z}_{(u,i)} = \mathbf{y}_{(v,i)}$ if $i \le R$ and 0 otherwise. It is immediate that the solution is feasible. Also, the value of the objective is the same as the value of the 2-to-1 SDP, as all the additional terms in the objective involve some vector $\mathbf{z}_{(u,i)}$ for some $i > R$ and are hence 0. Thus, the SDP value for the new instance is 1.

To bound the optimal value of any labeling $A : U \cup V \to [2R]$, we split it as

$$\mathbf{E}_{e=(u,v)\in E}\left[\mathbf{1}_{\{(A(u),A(v))\ \text{satisy}\ e\}}\right] \;=\; \mathbf{E}_{e=(u,v)\in E}\left[\mathbf{1}_{\{A(u)\le R\}} \cdot \mathbf{1}_{\{(A(u),A(v))\ \text{satisy}\ e\}}\right]$$
$$+\, \mathbf{E}_{e=(u,v)\in E}\left[\mathbf{1}_{\{A(u)>R\}} \cdot \mathbf{1}_{\{(A(u),A(v))\ \text{satisy}\ e\}}\right]$$

Note that the first term is simply the number of 2-to-1 constraints satisfied by A and it at most δ by assumption.

Also, for any fixed labeling A, the probability over the choice of the random maps $\{\sigma_{1,e}'\}_{e \in E}$, that $(A(u), A(v))$ satisfy e given that $A(u) > R$, is at most $1/R$. By a Chernoff

bound, the fraction of edges (u,v) satisfied with $A(u) > R$ is at most $1/R + \epsilon$ with probability $exp(-\epsilon^2 |E|/3)$ over the choice of the random maps. By a union bound and the condition on ϵ, the second term is at most $1/R + \epsilon$ for all labelings A, with high probability over the choice of $\{\sigma'_{1,e}\}_{e \in E}$. Picking an instance with the appropriate choice of the maps $\sigma'_{1,e}$ gives the required instance $\mathcal{L}'$. $\qquad\square$

6.7 Discussion

The instances we construct have SDP value 1 only for the most basic SDP relaxation. It would be desirable to get gaps for stronger SDPs, beginning with the most modest extensions of this basic SDP. For example, in the SDP for 2-to-1 LABEL-COVER from Figure 6.2, we can add valid nonnegativity constraints for the dot product between every pair of vectors in the set

$$\{\mathbf{y}_{(u,i)} \mid u \in U, i \in [R_1]\} \bigcup \{\mathbf{z}_{(v,j)} \mid v \in V, j \in [R_2]\} \,,$$

since in the integral solution all these vectors are $\{0,1\}$-valued. The vectors we construct do *not* obey such a nonnegativity requirement. For the case of UNIQUE-GAMES, Khot and Vishnoi [107] were able to ensure nonnegativity of all dot products by simply taking tensor products of the vectors with themselves and defining new vectors $\mathbf{y}'_{(u,i)} = \mathbf{y}^{\otimes 2}_{(u,i)} = \mathbf{y}_{(u,i)} \otimes \mathbf{y}_{(u,i)}$ and $\mathbf{z}'_{(v,j)} = \mathbf{z}^{\otimes 2}_{(v,j)} = \mathbf{z}_{(v,j)} \otimes \mathbf{z}_{(v,j)}$. Since $\langle \mathbf{a}^{\otimes 2}, \mathbf{b}^{\otimes 2} \rangle = \langle \mathbf{a}, \mathbf{b} \rangle^2$, the desired nonnegativity of dot products is ensured.

We cannot apply this tensoring idea in our construction as it does not preserve the SDP value at 1. For example, for 2-to-1 Label Cover, if we have $\mathbf{y}_{(u,i)} = \mathbf{z}_{(v,j_1)} + \mathbf{z}_{(v,j_2)}$ (so that these vectors contribute 1 to the objective value to the SDP of Figure 6.2), then upon tensoring we no longer necessarily have $\mathbf{y}^{\otimes 2}_{(u,i)} = \mathbf{z}^{\otimes 2}_{(v,j_1)} + \mathbf{z}^{\otimes 2}_{(v,j_2)}$. Extending our gap instances to obey the nonnegative dot product constraints is therefore a natural question that we leave open. While this seems already quite challenging, one can of course be more ambitious and ask for gap instances for stronger SDPs that correspond to certain number of rounds of some hierarchy, such as the Sherali-Adams hierarchy together with consistency of vector dot products with pairwise marginals. For UNIQUE-GAMES, gap instances for several rounds of such a hierarchy were constructed in [127].

Chapter 7

Unique Games over Integers

7.1 Introduction

In this Chapter, assuming the UGC we prove that it is NP-hard to $(1-\epsilon,\epsilon)$-approximate MAX 2-LIN$_\mathbb{Z}$ (and MAX 2-LIN$_\mathbb{R}$).

7.1.1 Motivation

As we have discussed, Khot *et al.* [99] showed that the UGC is equivalent to the following statement: for any $\epsilon > 0$, MAX 2-LIN$_q(1-\epsilon,\epsilon)$ is NP-hard for large enough q.

   An obvious question left open is whether the UGC also implies hardness of solving two-variable linear equations over the *integers*, rather than over the integers modulo a large constant.

Question 7.1.1. *Is it true that for all constant $\epsilon,> 0$, the* MAX 2-LIN$_\mathbb{Z}(1-\epsilon,\epsilon)$ *problem is* NP-*hard assuming the UGC?*

We believe that lack of an additional quantifier over q here gives this question a certain aesthetic appeal.

7.1.2 Related Work

The version of Question 7.1.1 for MAX 3-LIN (i.e., equations of the form $v_i - v_j + v_k = c_{ijk}$) took a relatively long time to be resolved. Håstad proved his celebrated NP-hardness result for MAX 3-LIN$_q(1-\epsilon, 1/q+\epsilon)$ in 1997 [74]; however, it was not until a decade later that Guruswami and Raghavendra [69] showed that indeed MAX 3-LIN$_\mathbb{Z}(1-\epsilon,\epsilon)$ is NP-hard for all constant $\epsilon > 0$. A relatively simple observation allowed Guruswami and Raghavendra to also deduce that MAX 3-LIN$_\mathbb{R}(1-\epsilon,\epsilon)$ is NP-hard; here the equations are still of the form $v_i - v_j + v_k = c_{ijk}$ for $c_{ijk} \in \mathbb{Z}$, but the variables can be assigned values in $\mathbb{R}$.

   A version of the MAX 3-LIN$_\mathbb{R}$ problem is also being studied by Khot and Moshkovitz in ongoing work. In their formulation, called ROBUST-MAX 3-LIN$_\mathbb{R}$, the constants c_{ijk} are all 0; however certain conditions are placed on how the variables v_i may be assigned real values, so as to eliminate the trivial solution $v_i \equiv 0$. Assuming the UGC, Khot and Moshkovitz [101] show that given a system with a $(1-\epsilon)$-good solution, roughly speaking it is NP-hard to find a solution in which a constant fraction of the equations are satisfied to within $\pm\Omega(\sqrt{\epsilon})$. Very recently they have eliminated the need for the UGC. The motivation for their work is the hope of establishing the same sort of result for ROBUST-MAX 2-LIN$_\mathbb{R}$, a problem closely connected with UNIQUE-GAMES.

7.1.3 Statement of Our Results

In this work we show a positive answer to Question 7.1.1. In fact, our main theorem is the following stronger result:

Theorem 7.1.2. *Assume the UGC. For any small constants $\epsilon > 0$, there exists a constant $q = q(\epsilon) \in \mathbb{N}$ such that the following holds: Given an instance $\mathcal{I}$ of of linear equations overl variables $\{x_k\}_{k=1}^n$ with the form $x_i - x_j = c_{ij}$ in which the integer constants c_{ij} are in the range $[-q,q]$, it is* NP-*hard to distinguish the following two cases:*

- *There is a $(1-\epsilon)$-good integer assignment to the variables.*

- *When equations are evaluated* modulo any integer $m \geq q$, *There is no assignment to the variables which is ϵ-good (i.e.,satisfies more than ϵ fraction of the equations)*

Assuming $\epsilon < 0.1$, it suffices for $q(\epsilon)$ to be large enough that $\widetilde{O}(1/q)^{\epsilon/(2-\epsilon)} \leq \epsilon$.

An interesting and somewhat novel aspect of this result is that it gives hardness even for a "multi-objective" problem. In the search version of Theorem 9.1.1's algorithmic task, although the algorithm is promised there is an extremely good *integer* solution to the given equations, it may attempt to find a slightly good solution modulo *any $m \geq \widetilde{O}(1/q)^{\epsilon/(2-\epsilon)}$ of its choosing*. We show that even still, the task is hard assuming the UGC.

From our main result Theorem 9.1.1, we immediately deduce the following corollaries:

Corollary 7.1.3. *Assuming the UGC, for all $\epsilon, > 0$ the* MAX 2-LIN$_\mathbb{Z}(1-\epsilon,\epsilon)$ *problem is* NP-*hard.*

Proof. If there is a ϵ-good integer assignment to the variables, then this assignment is also ϵ-good modulo q (or any other integer $m \geq q$). $\square$

Corollary 7.1.4. *Assuming the UGC, for all $\epsilon > 0$ there exists q such that the* MAX 2-LIN$_m(1-\epsilon,\epsilon)$ *problem is* NP-*hard for any $m \geq q$, even for $m = m(n)$ which is super-constant. In particular, the algorithmic task in Theorem 7.1.2 is equivalent to the UGC.*

Proof. If there is a $(1-\epsilon)$-good integer assignment to the variables, it is also $(1-\epsilon)$-good modulo m. $\square$

Corollary 7.1.5. *Assuming the UGC, for all $\epsilon > 0$ the* MAX 2-LIN$_\mathbb{R}(1-\epsilon,\epsilon)$ *problem is* NP-*hard.*

Proof. Certainly any $(1-\epsilon)$-good integer assignment to the variables is also a $(1-\epsilon)$-good real assignment. Further, as each constraint in Theorem 9.1.1 is of the form $v_i - v_j = c_{ij} \in \mathbb{Z}$, any ϵ-good real assignment to the variables v_i can be converted into a ϵ-good integer assignment simply by dropping all the fractional parts. $\square$

7.2 Overview of Our Proof

We now describe the new ideas we introduce to prove Theorem 9.1.1. In this section, we assume the reader is closely familiar with the proof of the Khot–Kindler–Mossel–O'Donnell (KKMO) UGC-hardness result for MAX 2-LIN$_q(1-\epsilon,\epsilon)$. Our discussions will also not be completely formal.

As KKMO showed, given $\epsilon > 0$ it is sufficient to construct a Dictator Test for functions $f : \mathbb{Z}_q^L \to \mathbb{Z}_q$ using 2Lin-constraints, with the following two properties: (i) dictator functions $f(x) = x_i$ pass the test with probability at least $1-\epsilon$; (ii) any $f : \mathbb{Z}_q^L \to \Delta_q$ with all influences smaller than τ passes the test with probability at most $1/q^{\epsilon/(2-\epsilon)} + \kappa$, where the "error term" $\kappa = \kappa(q,\epsilon,\tau)$ can be made arbitrarily small by taking $\tau > 0$ to be a sufficiently small constant *independent of L*. Here Δ_q is the convexification of $\mathbb{Z}_q$; i.e., the set of all probability distributions over $\mathbb{Z}_q$.

As a first step one might try extending the KKMO analysis to MAX 2-LIN$_m$, where m is "super-constant". The essential difficulty is that applying the key tool, the Majority Is

Stablest Theorem, to τ-small-influence functions $f : [m]^L \to [0,1]$ introduces an error term $\kappa(m,\epsilon,\tau)$ which depends on m. If m is super-constant, even as a function of L, this will cause the KKMO reduction from UNIQUE-GAMES$_L$ to fail; in particular, it means that in the soundness case, one would decode such f's to $\omega_L(1)$ many labels in $[L]$, which is unacceptable.

Since we presumably must use the Majority Is Stablest Theorem, and since we also care about constraints modulo a super-constant m, we are led to consider Dictator Tests for functions $f : [q]^L \to \mathbb{Z}_m$. We are not aware of any prior work on testing such functions, with differing domain and range (arguably, the work on hardness of ordering constraints [67] has some of the same flavor). An initial difficulty in working with such functions is that the usual method of "folding" no longer makes sense. Our first observation is that one need not fold by the usual method of restricting the domain by a factor of q; instead, one can build folding directly into the KKMO test. I.e., KKMO's result could be obtained via the following Dictator Test for functions $f : \mathbb{Z}_q^L \to \mathbb{Z}_q$: Choose $x, x' \sim \mathbb{Z}_q^L$ to be $(1-\epsilon)$-correlated random strings, choose also $c, c' \in \mathbb{Z}_q$ uniformly and independently, and then test the 2Lin constraint

$$f(x + (c,c,\ldots,c)) - c = f(x' + (c',c',\ldots,c')) - c'. \tag{7.1}$$

To analyze the soundness of this test, one introduces the "randomized (or d) function" $g : \mathbb{Z}_q^L \to \Delta_q$ defined by $g(x) = g(x+(c,\ldots,c))-c$, in which case the probability that f passes the test is $\mathbb{S}_{1-\epsilon}[g]$. One then observes that $\mathbb{E}[g_a(x)] = 1/q$ for each coordinate output function $g_a : \mathbb{Z}_q^L \to [0,1]$, $a \in \mathbb{Z}_q$. Thus one can apply the Majority Is Stablest to bound $\mathbb{S}_{1-\epsilon}[g]$ by

$$q(\Gamma(1/q) + \kappa(q,\epsilon,\tau)) \le (1/q)^{\epsilon/(2-\epsilon)} + o_L(1),$$

as necessary.

We will show how to extend this analysis to functions $f : [q]^L \to \mathbb{Z}_m$, where $m \ge q$. Proceeding with the same "built-in folding", we obtain the function $g : [q]^L \to \Delta_m$ which has the property that $\mathbb{E}[g_a(x)] \le 1/q$ for each $a \in [m]$. Our main technical result, Lemma 7.4.3, shows that this is sufficient to prove

$$\mathbb{S}_{1-\epsilon}[g] = \sum_{a \in [m]} \mathbb{S}_{1-\epsilon}[g_a] \le (1/q)^{\epsilon/(2-\epsilon)} + q^{\log q}\kappa(q,\epsilon,\tau) = (1/q)^{\epsilon/(2-\epsilon)} + o_L(1).$$

The key point here is that the error term does not depend at all on m, and hence the overall analysis works even for m super-constant. To evade dependence on m, the idea is that one can obtain the bound $\mathbb{S}_{1-\epsilon}[g_a] \le \mathbb{E}[g_a](1/q)^{\epsilon/2}$ without any small-influences assumption at all if $\mathbb{E}[g_a] \le q^{-\log q}$; one only needs to use hypercontractivity.

These ideas let us obtain UG-hardness of MAX 2-LIN$_m(1-\epsilon,\epsilon)$ even for super-constant m. To complete the proof of our main Theorem 9.1.1, we need to improve the completeness aspect of the Dictator Test so that even integer-valued dictators $f : [q]^L \to \mathbb{Z}$ pass with probability close to 1. An observation here is that an integer-valued dictator $f(x) = x_i$ already passes our test with probability close to 1/2: Ignoring the ϵ-noise, the test (7.1) fails only if one of $x_i + c$, $x + c'$ "wraps around" modulo q but the other doesn't.

There is a very simple idea for decreasing the probability of such wraparound: choose c and c' from a range smaller than $[q]$. E.g., if we choose $c, c' \sim [q/t]$, then we get wraparound in $x_i + c$ with probability at most $1/t$. Hence integer-valued dictators $f : [q]^L \to \mathbb{Z}$,

$f(x) = x_i$ will pass the test in (7.1) with probability at least $1 - \epsilon - 2/t$. How does this restricted folding affect the soundness analysis? It means that the associated randomized function $g : [q]^L \to \Delta_m$ will only satisfy $\mathbf{E}[g_a] \leq t/q$ for each $a \in [m]$, rather than $\mathbf{E}[g_a] \leq 1/q$. But this is still sufficient for our technical Lemma 7.4.3 to bound $\mathbb{S}_{1-\epsilon}[g]$ by roughly $(t/q)^{\epsilon/(2-\epsilon)}$. Thus by taking $t = \log(q)$, say, we get a 2Lin-based Dictator Test having integer-valued completeness $1 - \epsilon - O(1/\log(q))$ and $\mathbb{Z}_m$-valued soundness $\tilde{O}(1/q)^{\epsilon/(2-\epsilon)}$ for any $m \geq q$. This suffices to establish our main Theorem 9.1.1.

7.2.1 Comparison with Guruswami–Raghavendra

Here we briefly compare our methods with those Guruswami and Raghavendra [69] used to establish hardness for MAX 3-LIN$_\mathbb{Z}$. Although they also mentioned MAX 3-LIN$_m$ for very large m in the overview of their work, their methods are somewhat more integer-specific than ours. In particular, they worked with Dictator Tests on functions $f : \mathbb{Z}_+^L \to \mathbb{Z}$, using a certain exponential distribution on the domain $\mathbb{Z}_+$. (Ultimately, of course, they truncated the distribution to a finite range.) This necessitated introducing and analyzing a somewhat technical method of decoding functions f to coordinates associated to sparse Fourier frequencies $\omega \in [0, 2\pi]^L$ with large Fourier coefficients.

Guruswami and Raghavendra also described their Dictator Tests as "derandomized versions" of Håstad's tests, where the amount of randomness of the test depends only on the soundness. The same could be said of our result vis-a-vis KKMO's Dictator Tests: we get MAX 2-LIN$_m$ Dictator Tests in which the size of the domain elements, q, depends only on the desired soundness of the test.

7.3 Definitions and analytic tools

7.3.1 Notation

For $r \in \mathbb{R}^+$ we let $[r]$ denote $\{1, 2, \ldots, \lfloor r \rfloor\}$. Given $m \in \mathbb{N}$ we write $\oplus_m$ for addition modulo m. It will also be convenient to use the following slightly unusual notation:

Definition 7.3.1. *We write $\mathbb{Z}_m$ for the group of integers modulo m. We will also sometimes identify this set with $[m] \subset \mathbb{Z}$, not with the more standard $\{0, 1, \ldots, m-1\}$. Finally, we extend the notation to $m = \infty$, in which case we understand $\mathbb{Z}_m$ to mean simply the integers, $\mathbb{Z}$.*

Definition 7.3.2. *We write Δ_m for the set of probability distributions over $\mathbb{Z}_m$ with finite support; when $m \neq \infty$ we can identify Δ_m with the standard $(m-1)$-dimensional simplex in $\mathbb{R}^m$. We also identify an element $a \in \mathbb{Z}_m$ with a distribution in Δ_m, namely, the distribution that puts all of its probability mass on a.*

7.3.2 Noise stability and influences on $[q]^n \to \mathbb{R}^m$

Here we need to used a generalization of the Harmonic Analysis introduced in Section 3.1.1. We will be considering functions of the form $f : [q]^n \to \mathbb{R}^m$ (as oppose to $\mathbb{R}$), where $q, n, m \in \mathbb{N}$. We will also allow $m = \infty$, in which case we interpret the range as all sequences in

$\mathbb{R}^{\mathbb{Z}}$ with at most finitely many nonzero coordinates. The set of all functions $f : [q]^n \to \mathbb{R}^m$ forms an inner product space with inner product

$$\langle f, g \rangle = \mathop{\mathbf{E}}_{x \sim [q]^n}[\langle f(x), g(x) \rangle];$$

here we mean that x is uniformly random and the $\langle \cdot, \cdot \rangle$ inside the expectation is the usual inner product in $\mathbb{R}^m$. We also write $\|f\| = \sqrt{\langle f, f \rangle}$ as usual.

For $0 \le \rho \le 1$, we define T_ρ to be the linear operator on this inner product space given by

$$T_\rho f(x) = \mathop{\mathbf{E}}_{y}[f(y)],$$

where y is a random string in $[q]^L$ which is ρ-correlated to x. We define the *noise stability of f at ρ* to be

$$\mathbb{S}_\rho[f] = \langle f, T_\rho f \rangle.$$

For $i \in [n]$, we define the *influence of i on $f : [q]^n \to \mathbb{R}^m$* to be

$$\mathrm{Inf}_i[f] = \mathop{\mathbf{E}}_{x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_n \sim [q]}\left[\mathbf{Var}_{x_i \sim [q]}[f(x)]\right],$$

where $\mathbf{Var}[f]$ is defined to be $\mathbf{E}[\|f\|^2] - \|\mathbf{E}[f]\|^2$. More generally, for $0 \le \eta \le 1$ we define the *η-noisy-influence of i on f* to be

$$\mathrm{Inf}_i^{(1-\eta)}[f] = \mathrm{Inf}_i[T_{1-\eta}f].$$

One may observe that

$$\mathrm{Inf}_i^{(1-\eta)}[f] = \sum_{j=1}^{m} \mathrm{Inf}_i^{(1-\eta)}[f_j],$$

where $f_j : [q]^n \to \mathbb{R}$ denotes the jth-coordinate output function of f. (When $m = \infty$ the sum should be over $j \in \mathbb{Z}$.)

We will need the following "convexity of noisy-influences" fact:

Proposition 7.3.3. *Let $f^{(1)}, \ldots, f^{(t)}$ be a collection of functions $[q]^n \to \mathbb{R}^m$. Then*

$$\mathrm{Inf}_i^{(1-\eta)}\left[\mathop{\mathrm{avg}}_{k \in [t]}\left\{f^{(k)}\right\}\right] \le \mathop{\mathrm{avg}}_{k \in [t]}\left\{\mathrm{Inf}_i^{(1-\eta)}[f^{(k)}]\right\}.$$

Here for any $c_1, c_2, \ldots c_t \in \mathbb{R}$ (or $\mathbb{R}^m$), we use the notation $\mathrm{avg}(c_1, \ldots, c_t)$ to denote their average:

$$\frac{\sum_{i=1}^{t} c_i}{t}.$$

Following fact is well known:

Fact 7.3.4. *For any η,*

$$\sum_{i=1}^{n} \mathrm{Inf}_i(T_{1-\eta}f) \le \frac{\mathbf{Var}(f)}{2e\eta}.$$

7.3.3 Hypercontractivity and Majority Is Stablest

Recall the hypercontractivity on $[q]^n$.

Theorem 7.3.5. *Let $q \geq 2$, $f : [q]^n \to \mathbb{R}$, and $0 \leq \epsilon < 1$. Then*

$$\|T_{\sqrt{1-\epsilon}} f\|_2 \leq \|f\|_p, \quad \text{where } p = p(q, \epsilon) = 1 + (1 - \epsilon)^{(2 - 4/q)/\log(q-1)}.$$

The second tool we need is the Majority is the Stablest Theorem.

Theorem 7.3.6. *Suppose $f : [q]^n \to [0,1]$ has $\mathrm{Inf}_i^{(1-\eta)}[f] \leq \tau \leq (\log q)^{-(\log q)/c}$ for all $i \in [n]$, where $\eta < c(\log q)/\log(1/\tau)$ and $c > 0$ is a certain universal constant. Let $\mu = \mathbf{E}[f]$. Then for any $0 < \epsilon < 1$,*

$$\mathbb{S}_{1-\epsilon}[f] \leq \Gamma_{1-\epsilon}(\mu) + \frac{\log q}{c\epsilon} \cdot \frac{\log\log(1/\tau)}{\log(1/\tau)}.$$

This is essentially a special case of Theorem 3.2.4, with the error bound explicitly given.

Proposition 7.3.7. *Assume $0 < \epsilon < .1$ and $0 \leq \mu \leq \exp(-1/\sqrt{\epsilon})/\sqrt{\epsilon}$. Then $\Gamma_{1-\epsilon}(\mu) \leq \mu^{1+\epsilon/(2-\epsilon)}$.*

This estimate follows from Corollary 10.2 in [99]. (The expression in that corollary is in fact an upper bound on $\Gamma_{1-\epsilon}(\mu)$ for all $0 < \epsilon < 1$ and $0 \leq \mu \leq 1/2$, as can be verified using the inequality in Proposition 6.1 of [99]. The simplified bound $\mu^{1+\epsilon/(2-\epsilon)}$ holds when $\epsilon < .1$ and $\mu \leq \exp(-1/\sqrt{\epsilon})/\sqrt{\epsilon}$.)

7.4 Dictator Tests

In this work we will be considering two-variable linear equation constraints; specifically, testing functions $f : [q]^n \to \mathbb{Z}_m$ using constraints of the form $f(x) - f(y) = c$, where $c \in \mathbb{Z}$.

Before defining Dictator Tests we need to introduce another small technical detail, that of testing *averages* of functions. Given a test for functions $f : [q]^n \to \mathbb{Z}_m$, say, we can think of it more generally as a test for functions $f : [q]^n \to \Delta_m$. To understand this, one should think of a function with range Δ_m as a "randomized" function into $\mathbb{Z}_m$. I.e., to apply the test $\mathcal{T}$ to a function $f : [q]^L \to \Delta_m$, one first chooses a random constraint as usual in $\mathcal{T}$; say it is $f(x) - f(y) = c$. One then chooses $a \sim f(x)$ and $a \sim f(y)$ (independently) and finally, one checks the constraint $a - y = c$.

We may now informally state what a *Dictator vs. Small Noisy-Influences Test* is. It is a test for functions $f : [q]^n \to \Delta_m$ with the following two properties: (i) *Dictator functions* — i.e., functions of the form $f(x) = x_i$ — pass the test with high probability. (Here we are interpreting the integer $x_i \in [q]$ also as an element of $\mathbb{Z}_m$, and thus also as an element of Δ_m.) In other words, $\mathrm{Val}_{\mathcal{T}}(f)$ is large when f is a dictator. (ii) Functions f satisfying $\mathrm{Inf}_i^{(1-\eta)}[f] \leq \tau$ for all $i \in [n]$ pass the test with low probability, where here η and τ should be thought of as very small constants. More formally:

Definition 7.4.1. *Let $\mathcal{T}$ be a test for functions $f : [q]^n \to \Delta_m$. We say that $\mathcal{T}$ has complete-ness at least c if every dictator function $f(x) = x_i$ passes the test with probability at least c. We say that $\mathcal{T}$ has (τ, η)-soundness at most s if every function $f : [q]^n \to \Delta_m$ satisfying $\mathrm{Inf}_i^{(1-\eta)}[f] \leq \tau$ for all $i \in [n]$ passes the test with probability at most s. Finally, given a family of tests $(\mathcal{T}_n)$, where $\mathcal{T}_n$ test functions $f : [q]^n \to \Delta_m$, we say it has soundness s if for every $\kappa > 0$ there exists $\tau, \eta > 0$ such that each $\mathcal{T}_n$ has (τ, η)-soundness at most $s + \kappa$.*

We now state our new family of Dictator vs. Small Noisy-Influences Tests. Given parameters $0 < \epsilon < 1$ and $q \in \mathbb{N}$, we define the following test $\mathcal{T}_{q,\epsilon}$ for functions f with domain $[q]^L$:

Test $\mathcal{T}_{q,\epsilon}$:

- Choose $x, x' \sim [q]^L$ to be a pair of $(1-\epsilon)$-correlated random strings.
- Choose $c, c' \sim [q/\log(q)]$ independently and uniformly.
- Define $y = x \oplus_q (c, c, \ldots, c)$, and define $y' = x' \oplus_q (c, c, \ldots, c)$.
- Test the constraint " $f(y) - c = f(y') - c'$ " (equivalently, " $f(y) - f(y') = c - c'$ ").

As discussed, one can also think of this test as an explicit weighted CSP of MAX 2-LIN type over the variable set $[q]^L$. The constraint $f(y) - c = f(y') - c'$ should be thought of as a formal expression, since we have not yet specified the *range* of the assignment f. In fact, we will analyze the test's properties when the range of f varies over different $\mathbb{Z}_m$'s.

To prove our main Theorem 9.1.1 it will suffice (as we verify in Section 7.5) to show the following.

Theorem 7.4.2. *The Dictator Test $\mathcal{T}_{q,\epsilon}$ uses integer constants c_{ij} in $[-q/\log(q), q/\log(q)]$ and has the following two properties:*

Completeness: *For each $m \in \mathbb{N} \cup \{\infty\}$, the L dictator functions $f : [q]^L \to \mathbb{Z}_m$ pass the test $\mathcal{T}_{q,\epsilon}$ with probability at least $1 - \epsilon - O(1/\log(q))$.*

Soundness: *Assume $0 < \epsilon < .1$ and that $q \geq \exp(1/\sqrt{\epsilon})$ an integer. Assume $f : [q]^L \to \Delta_m$ satisfies $\mathrm{Inf}_i^{(1-\eta)}[f] \leq \tau \leq (\log q)^{-(\log q)/c}$ for all $i \in [L]$, where $\eta < c(\log q)/\log(1/\tau)$ (and c is the constant from Theorem 7.3.6). Assume further that $q/\log(q) \leq m \leq \infty$. Then f passes the test $\mathcal{T}_{q,\epsilon}$ with probability less than*

$$\widetilde{O}(1/q)^{\epsilon/(2-\epsilon)} + \frac{\widetilde{O}(q^{\log q})}{\epsilon} \cdot \frac{\log\log(1/\tau)}{\log(1/\tau)}.$$

The Completeness part of Theorem 7.4.2 is easy to verify:

Proof. Suppose $f(x) = x_j$ for some $j \in [L]$. In the test $\mathcal{T}_{q,\epsilon}$ we have $x_j = x'_j$ except with probability at most ϵ. When the event happens, write b for the common value. We further have that b is at most $q - \lfloor q/\log(q) \rfloor$ except with probability at most $O(1/\log(q))$. Thus with probability at least $1 - \epsilon - O(1/\log(q))$ we have both $y_j = b + c$ and $y'_j = b + c'$ as integers in $[q]$; i.e., the $\oplus_q$ does not cause "wrap-around". Thus $f(y)$ will equal the integer $b + c$ within $\mathbb{Z}_m$, and similarly $f(y')$ will equal $b + c'$ within $\mathbb{Z}_m$, and the tested constraint will be satisfied. $\qquad \square$

The next two subsections of the work are devoted to the proof of the Soundness part of Theorem 7.4.2. In the first subsection we prove a technical lemma bounding the noise stability of functions $f : [q]^L \to \Delta_m$ which have $\|f_j\|_\infty$ small for each $j \in \mathbb{Z}_m$. In the subsequent subsection, we complete the proof of the soundness of our test.

7.4.1 Technical lemma

Our soundness analysis relies on the following technical lemma; the crucial aspect of it is that the upper bound we give on the noise stability does not depend on m.

Lemma 7.4.3. *Fix $0 < \epsilon < .1$ and let $q \geq \exp(1/\sqrt{\epsilon})$ be an integer. Further, let $L, m \in \mathbb{N}$ and $0 < \eta < 1$. Assume $g : [q]^L \to \Delta_m$ satisfies $\mathrm{Inf}_i^{(1-\eta)}[g] \leq \tau \leq (\log q)^{-(\log q)/c}$ for all $i \in [L]$, where $\eta < c(\log q)/\log(1/\tau)$ (and c is the constant from Theorem 7.3.6).*

Then if $\mathbf{E}_x[g(x)_a] \leq \log(q)/q$ for all $a \in [m]$, it follows that

$$\mathbb{S}_{1-\epsilon}[g] < \widetilde{O}(1/q)^{\epsilon/(2-\epsilon)} + \frac{\widetilde{O}(q^{\log q})}{\epsilon} \cdot \frac{\log\log(1/\tau)}{\log(1/\tau)}.$$

Proof. Write $\mu_a = \mathbf{E}_x[g(x)_a]$. We use two different bounds for $\mathbb{S}_{1-\epsilon}[g_a]$ depending on the magnitude of μ_a. The first bound uses the small noisy-influences of g_a (which are certainly smaller than those of g) and the Majority Is Stablest Theorem (Theorem 7.3.6), yielding

$$\mathbb{S}_{1-\epsilon}[g_a] \leq \Gamma_{1-\epsilon}(\mu_a) + e(\tau), \qquad e(\tau) := \frac{\log q}{c\epsilon} \cdot \frac{\log\log(1/\tau)}{\log(1/\tau)}.$$

We may also use Proposition 7.3.7 because $\epsilon < .1$ and $\mu_a \leq \log(q)/q \leq \exp(-1/\sqrt{\epsilon})/\sqrt{\epsilon}$; thus

$$\mathbb{S}_{1-\epsilon}[g_a] \leq \mu_a^{1+\epsilon/(2-\epsilon)} + e(\tau). \tag{7.2}$$

Our second bound is more useful when μ_a is extremely small; it only needs the hyper-contractivity theorem (Theorem 7.3.5), and not the small noisy-influences condition. The theorem gives

$$\mathbb{S}_{1-\epsilon}[g_a] = \|T_{\sqrt{1-\epsilon}}g_a\|_2^2 \leq \|g_a\|_p^2 = \mathbf{E}[g_a^p]^{2/p} \leq \mathbf{E}[g_a]^{2/p} = \mu_a^{2/p},$$

where $p = 1 + (1-\epsilon)^{(2-4/q)/\log(q-1)}$ as in Theorem 7.3.5. One can check that $2/p \geq 1 + \epsilon/(1.9\log q)$ for all $0 < \epsilon < 1$ and $q \geq 3$; hence:

$$\mathbb{S}_{1-\epsilon}[g_a] \leq \mu_a^{1+\epsilon/(1.9\log q)}. \tag{7.3}$$

We now put the two bounds together:

$$
\begin{aligned}
\mathbb{S}_{1-\epsilon}[g] &= \sum_{a \in [m]} \mathbb{S}_{1-\epsilon}[g_a] \\
&= \sum_{a : \mu_a \geq q^{-\log q}} \mathbb{S}_{1-\epsilon}[g_a] + \sum_{a : \mu_a < q^{-\log q}} \mathbb{S}_{1-\epsilon}[g_a] \\
&= \sum_{a : \mu_a \geq q^{-\log q}} (\mu_a^{1+\epsilon/(2-\epsilon)} + e(\tau)) + \sum_{a : \mu_a < q^{-\log q}} \mu_a^{1+\epsilon/(1.9\log q)} \quad \text{(using (7.2), (7.3))}.
\end{aligned}
$$

Since g's range is Δ_m we have $\sum_{a \in [m]} \mu_a = 1$. Thus the first sum above is at most

$$q^{\log q} e(\tau) + \sum_{a : \mu_a \geq q^{-\log q}} \mu_a^{1+\epsilon/(2-\epsilon)} \leq q^{\log q} e(\tau) + \max_a \mu_a^{\epsilon/(2-\epsilon)} \leq q^{\log q} e(\tau) + (\log(q)/q)^{\epsilon/(2-\epsilon)}$$

using the assumed upper bound on μ_a. The second sum above is at most

$$\max_{a:\mu_a<q^{-\log q}} \mu_a^{\epsilon/(1.9\log q)} \le (q^{-\log q})^{\epsilon/(1.9\log q)} = q^{-\epsilon/1.9}.$$

Thus we conclude

$$\mathbb{S}_{1-\epsilon}[g] \le q^{\log q} e(\tau) + (\log(q)/q)^{\epsilon/(2-\epsilon)} + q^{-\epsilon/1.9} < \tilde{O}(1/q)^{\epsilon/(2-\epsilon)} + \frac{\tilde{O}(q^{\log q})}{\epsilon} \cdot \frac{\log\log(1/\tau)}{\log(1/\tau)}$$

as claimed. □

7.4.2 Soundness of the test

This section is devoted to the proof of the Soundness part of Theorem 7.4.2.

Proof. Given f as in the statement of the theorem, we introduce another randomized function $g : [q]^L \to \Delta_m$. Specifically, $g(x)$ is defined to be the distribution function on $a \in \mathbb{Z}_m$ given by the following experiment:

- Choose $c \sim [q/\log(q)]$ uniformly at random.
- Choose b according to the distribution $f(x \oplus_q (c,c,\dots,c))$.
- Define $a = b - c \in \mathbb{Z}_m$.

Thus in the test $\mathcal{T}_{q,\epsilon}$, once x and x' are chosen the probability that f passes the test is equal to the probability that independent draws from $g(x)$ and $g(x')$ yield the same value in $\mathbb{Z}_m$. I.e.,

$$\mathbf{Pr}[f \text{ passes the constraint}] = \mathop{\mathbf{E}}_{x,x'}[\langle g(x), g(x')\rangle] = \mathbb{S}_{1-\epsilon}[g].$$

It thus suffices to bound $\mathbb{S}_{1-\epsilon}[g]$.

Our first task is to show that g has small noisy-influences. Define the operator S_c for $c \in \mathbb{Z}_q$ as follows: $S_c h(x) = h(x \oplus_q (c,c,\dots,c))$. Define the operator R_c for $c \in \mathbb{Z}_m$ as follows: $(R_c h(x))_a = h(x)_{a+c}$, where the sum $a+c$ is within $\mathbb{Z}_m$. Hence by definition,

$$g = \mathop{\mathrm{avg}}_{c \in [q/\log(q)]} \{R_c S_c f\}. \tag{7.4}$$

In particular, for each $i \in [L]$ we have

$$\mathrm{Inf}_i^{(1-\eta)}[g] \le \mathop{\mathrm{avg}}_{c \in [q/\log(q)]} \{\mathrm{Inf}_i^{(1-\eta)}[R_c S_c f]\}$$

by the convexity of noisy-influences (Proposition 7.3.3). But it's easy to see that $\mathrm{Inf}_i^{(1-\eta)}[R_c h] = \mathrm{Inf}_i^{(1-\eta)}[h]$ and $\mathrm{Inf}_i^{(1-\eta)}[S_c h] = \mathrm{Inf}_i^{(1-\eta)}[h]$. Hence we conclude $\mathrm{Inf}_i^{(1-\eta)}[g] \le \mathrm{Inf}_i^{(1-\eta)}[f] \le \tau$ for all $i \in [L]$.

We now make the key observation. For $a \in \mathbb{Z}_m$, define $\mu_a = \mathbf{E}_{x \sim [q]^L}[g(x)_a]$. Using the original definition of g we have

$$\mu_a = \mathop{\mathbf{Pr}}_{\substack{x,c \sim [q/\log(q)], \\ b \sim f(x \oplus_q(c,c,\dots,c))}}[b - c = a] = \mathop{\mathbf{E}}_{x,c \sim [q/\log(q)]}[f(x \oplus_q (c,c,\dots,c))_{a+c}],$$

where the expressions $b - c = a$ and $a + c$ are treated within $\mathbb{Z}_m$. But the joint distribution of c and $x \oplus_q (c, c, \ldots, c)$ is identical to the joint distribution of c and y, where $y \sim [q]^L$ is uniform and *independent* of c. Hence

$$\mu_a = \operatorname*{\mathbf{E}}_{y, c \sim [q/\log(q)]}[f(y)_{a+c}] \leq \max_{y \in [q]^L} \left\{ \operatorname*{\mathbf{E}}_{c \sim [q/\log(q)]}[f(y)_{a+c}] \right\} \leq \log(q)/q \quad \text{for all } a \in \mathbb{Z}_m, \quad (7.5)$$

since $\sum_b f(x)_b = 1$ and $m \geq q/\log(q)$.

Having established (7.5) and also $\mathrm{Inf}_i^{(1-\eta)}[g] \leq \tau$ for all i, we may bound $\mathbb{S}_{1-\epsilon}[g]$ and thus complete the proof using the technical Lemma 7.4.3. (In the case that $m = \infty$ we may still apply the lemma because g's outputs are nonzero on only finitely many coordinates; hence we may consider g's range to be a finite-dimensional simplex.) $\qquad\square$

7.5 The Reduction from UNIQUE-GAMES$_L$

In this section we show how to use our Dictator Test to obtain our main UG-hardness result, Theorem 9.1.1. We reiterate that we are essentially using the reduction implicitly proved in [99]; we give the full deduction here for completeness and because we are working in a slightly nonstandard setting.

For technical convenience, we will use the following equivalent version of the UGC due to Khot and Regev [94, Lemma 3.6]:

Theorem 7.5.1. *Assume the UGC. For all small $\zeta, \gamma > 0$, there exists $L \in \mathbb{N}$ such given an unweighted* UNIQUE-GAMES$_L$ *instance $\mathcal{G} = (U, V, E, (\pi_{u,v})_{(u,v) \in E})$ which is U-regular, it is* NP-*hard to distinguish the following two cases:*

1. *There is an assignment $A : (U \cup V) \to [L]$ and a subset $U' \subseteq U$ with $|U'|/|U| \geq 1 - \zeta$ such that A satisfies all constraints incident on U'.*

2. *There is no γ-good assignment A.*

Our main task, which we will carry out in the next subsection, will be to prove the following slight variant of Theorem 9.1.1, wherein we write $s(q, \epsilon) = \tilde{O}(1/q)^{\epsilon/(2-\epsilon)}$ for the main term in the Soundness part of Theorem 7.4.2:

Theorem 7.5.2. *Fix $0 < \epsilon < .1$ rational and $q \geq \exp(1/\sqrt{\epsilon})$ an integer. For any $L \in \mathbb{N}$, there is a polynomial-time reduction mapping non-bipartite, unweighted* UNIQUE-GAMES$_L$ *instances $\mathcal{G}$ into* MAX 2-LIN *instances $\mathcal{I}$ having the following properties:*

- *(Completeness.) If statement 1 in Theorem 8.4.2 holds for $\mathcal{G}$, then there is an integer assignment to the variables in $\mathcal{I}$ satisfying at least $(1-\zeta)(1-\epsilon-O(1/\log(q)))$-weight of the equations.*

- *(Soundness.) If there is no γ-good assignment for $\mathcal{G}$ where $\gamma = \gamma(q, \epsilon) > 0$ is sufficiently small, then there is no integer assignment to the variables in $\mathcal{I}$ which satisfies at least $(3s(q, \epsilon))$-weight of the equations modulo m, for any integer $m \geq q/\log(q)$.*

By combining Theorem 8.4.3 with Theorem 8.4.2, taking $\zeta = 1/\log(q)$ and $\gamma = \gamma(q, \epsilon) > 0$ as necessary, we obtain the following variant of Theorem 9.1.1:

Theorem 7.5.3. *Assume the UGC. For any $0 < \epsilon < .1$ rational and $q \geq \exp(1/\sqrt{\epsilon})$ an integer, the following holds: Given an instance $\mathcal{I}$ of* MAX 2-LIN *in which the integer constants c_{ij} are in the range $[-q/\log(q), q/\log(q)]$, it is* NP-*hard to distinguish the following two cases:*

- *There is a $(1 - \epsilon - O(1/\log(q)))$-good integer assignment to the variables.*
- *There is no assignment to the variables which is $\tilde{O}(1/q)^{\epsilon/(2-\epsilon)}$-good modulo any integer $m \geq q/\log(q)$.*

From this, we can deduce our main Theorem 9.1.1 for ϵ' and ϵ' by taking ϵ in Theorem 7.5.3 a rational of the form $\epsilon' - \Theta(1/\log(q))$.

7.5.1 Proof of Theorem 8.4.3

We now prove Theorem 8.4.3.

Proof. The reduction is essentially as in [99]. Given the UNIQUE-GAMES$_L$ instance $\mathcal{G} = (U, V, E, (\pi_{uv}))$, the reduction produces a weighted MAX 2-LIN instance $\mathcal{I}$ with variable set $V \times [q]^L$. We think of an assignment F to these variables as a collection of functions $f_v : [q]^L \to \mathbb{Z}_m$, one for each $v \in V$. Here we will allow $q/\log(q) \leq m \leq \infty$. For each $u \in V$ we also introduce the randomized function $f_u : [q]^L \to \Delta_m$ defined by

$$f_u(x) = \underset{u:(u,u)\in E}{\mathbf{E}}[f_u^{\pi_{uu}}(x)],$$

where define the functions $f_v^\pi : [q]^L \to \mathbb{Z}_m$ by

$$f_v^\pi(x) = f_v(x \circ \pi^{-1}), \qquad \text{with } x \circ \pi^{-1} \in [q]^L \text{ defined by } (x \circ \pi^{-1})_j = x_{\pi^{-1}(j)}.$$

We now define the instance according to the following probabilistic test:

- Choose $u \in U$ randomly.
- Apply test $\mathcal{T}_{q,\epsilon}$ from Section 7.4 to f_u.

Note that by the definition of applying a test to a randomized function, this indeed makes $\mathcal{I}$ a weighted MAX 2-LIN instance over the variables $V \times [q]^L$. Further, it is easy to check that the reduction from $\mathcal{G}$ to $\mathcal{I}$ thus defined can be carried out in polynomial time assuming ϵ, q, and L are constant.

To prove the Completeness part of Theorem 8.4.3, suppose that assignment A and subset $U' \subseteq U$ are as in statement 1 of Theorem 8.4.2. Define an integer-valued assignment F for $\mathcal{I}$ by taking $f_v(x) = x_{A(v)}$. Then by definition and by the property of A, we will have that $f_u : [q]^L \to \Delta_{\mathbb{Z}}$ is in fact the $A(u)$th dictator function for all $u \in U'$. Thus by the completeness part of Theorem 7.4.2, assignment F will pass the test $\mathcal{I}$ with probability at least $\mathbf{Pr}[u \in U'] \cdot (1 - \epsilon - O(1/\log(q))) \geq (1 - \zeta)(1 - \epsilon - O(1/\log(q)))$. This finishes the Completeness part of Theorem 8.4.3.

As for the Soundness part of Theorem 8.4.3, choose $\tau = \tau(q, \epsilon) > 0$ small enough so that the error term in the Soundness part of Theorem 7.4.2 is at most the main term, $s(q, \epsilon)$; choose also $\eta = \eta(q, \epsilon) > 0$ sufficiently small so that the hypothesis therein holds. By way of proving the contrapositive, suppose that there is an integer $m \geq q/\log(q)$ and a $\mathbb{Z}_m$-valued assignment F to $\mathcal{I}$ which passes the test $\mathcal{I}$ with probability at least $3s(q, \epsilon)$. Then by an averaging argument, there must be some subset $U' \subseteq V$ of fractional size at least $s(q, \epsilon)$ such that when $u \in U'$, the test $\mathcal{T}_{q,\epsilon}$ passes f_u with probability at least $2s(q, \epsilon)$. It follows from the Soundness part of Theorem 7.4.2, along with our choice of τ and η, that

$$\text{for all } u \in U', \quad \exists i_u \in [L] \text{ s.t. } \mathrm{Inf}_{i_u}^{(1-\eta)}[f_u] > \tau. \tag{7.6}$$

By definition of f_u and by the convexity of noisy-influences (Proposition 7.3.3) we deduce that for each such $u \in U'$ and $i_u \in [L]$,

$$\tau < \operatorname*{avg}_{v:(u,v)\in E} \left\{ \mathbf{Inf}^{(1-\eta)}_{i_u}[f_v^{\pi_{uu}}] \right\} = \operatorname*{avg}_{v:(u,v)\in E} \left\{ \mathbf{Inf}^{(1-\eta)}_{\pi_{uv}(i_u)}[f_v] \right\}$$

$$\Rightarrow \quad \tau/2 \le \mathbf{Inf}^{(1-\eta)}_{\pi_{uv}(i_u)}[f_v] \qquad \text{for at least a } \tau/2\text{-fraction of } u\text{'s neighbors } v. \quad (7.7)$$

For each $v \in V$ let us define

$$C(v) = \{ j \in [L] : \mathbf{Inf}^{(1-\eta)}_{j}[f_v] > \tau/2 \};$$

thus by (10.9) we have:

$$\forall\, u \in U', \quad \pi_{uv}(i_u) \in C(v) \text{ for at least a } \tau/2\text{-fraction of } u\text{'s neighbors } v \in V. \qquad (7.8)$$

We claim and will show shortly that $|C(v)| \le 1/(\eta\tau)$ for all v. Having established this, consider choosing a random assignment $A : (U \cup V) \to [L]$ as follows: for $u \in U'$ set $A(u) = i_u$; for $v \in V$, choose $A(v)$ randomly from $C(v)$ (assuming the set is nonempty); finally, set $A(w)$ arbitrarily in $[L]$ for all unassigned vertices w. Now by (7.8), for each $u \in U'$ the expected fraction of constraints incident on u which A satisfies is at least $(\tau/2)/(\eta\tau) = \eta\tau^2/2$. Since $|U'|/|U| \ge s(q,\epsilon)$ and $\mathcal{G}$ is U-regular, we conclude that the expected fraction of all constraints in $\mathcal{G}$ that A satisfies is at least $s(q,\epsilon)\eta\tau^2/2$. Taking $\gamma = \gamma(q,\epsilon) = s(q,\epsilon)\eta\tau^2/2$, we conclude that there must exist a γ-good assignment for $\mathcal{G}$.

It remains to verify the claim that $|C(v)| \le 1/(\eta\tau)$ for all v. This is true by Fact 7.3.4. We also need the following small observation: even for arbitrarily large m, if $h : [q]^L \to \Delta_m$ then $\mathbf{Var}[h] \le 1$. This is because $\mathbf{Var}[h] \le \mathbf{E}[\|h\|^2] \le \max_x\{\|h(x)\|^2\} \le 1$, as every point in Δ_m has Euclidean norm at most 1. Thus

$$|C(v)| = \left| \{ j \in [L] : \mathbf{Inf}^{(1-\eta)}_{j}[f_v] > \tau/2 \} \right| \le \frac{1/(2e\eta)}{\tau/2} \le 1/(\eta\tau),$$

as claimed. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \square$

Chapter 8

On Hardness of vertex pricing

8.1 Introduction

We study the item pricing problem which is a CSP with the constraint being a generalized payoff function.

8.1.1 Motaviation and Background

A informal description of the problem is as follows: a seller has an infinite supply of n different items. There are m buyers, each of which are interested in a subset of the items with certain budget limit. These buyers are all *single minded*; i.e., they either buy all the items they are interested in if the overall cost is within their budget or they will buy none of them. The algorithmic task is to price each item i with a profit margin p_i to maximize the overall profit of the seller.

Serval results were known when the profit margin p_i on each item is required to be *positive*. A $O(\log n + \log m)$ approximation for the general problems is given by Guruswami *et al.* [64]. If we assume that each customer is only interested in a constant number k of the items, a $O(k^2)$-approximation algorithm was given in [26] by Briest and Krysta. Later in [16], Balcan and Blum improved the approximation ratio to $O(k)$. In particular, when $k = 2$ (such a problem is also called *graph vertex pricing*), their algorithm gave an 4-approximation. On the hardness side, an APX-hardness result was obtained for the general problem in [64]. Later, Demaine, Feige, Hajiaghayi, and Salavatipour obtained a poly-logarithmic hardness [38]. As for the case that each customer is only interested in at most 2 of the items, a 2-hardness result was obtained in [93] assuming the Unique Games Conjecture (UGC).

Much less is known when the seller is allowed to assign negative profit margin p_i for some of the items. The motivation behind selling some items below the margin cost is to increase the overall profit by stimulating the sales of other products. These items sold below the cost are usually referred as the *"loss leaders"*. One example of loss leaders is in the market of digital book reader (such as the Kindle and IPad), the seller may price the reading device at a low price so as to make more money on the sales of the digital books.

Studying the problem of pricing loss leaders is formulated as an open problem in [16]; the authors asked: "what kind of approximation guarantees are achievable if one allows the seller to price some items below their margin cost?" Interestingly, the authors found that by optimally pricing some of the items below cost, one could possibly achieve a profit that is $\Omega(\log n)$ times of the maximum profit under the positive price model. The problem of pricing loss leaders is further studied by Balcan *et al.* in [56]. They introduced two new models: the *coupon* and *discount* model. Roughly speaking, the discount model is the item pricing problem with negative profit margin allowed; the coupon model adds an additional assumption that a seller's profit is at least 0 for the entire transactions with each customer. The same $\Omega(\log n)$ "profitability gap" was shown under these models.

In this work, we give a negative result for pricing loss leaders. In particular, we show that obtain a *constant approximation* for item pricing, under either the coupon or discount model, is NP-hard assuming the Unique Games Conjectures; our hardness result holds even for the very simple case that each customer is only interested at most $k = 3$ items. Our result should be compared with the case when only positive prices is assigned, there

is an $\frac{1}{3e}$-approximation for such a problem.

8.1.2 Problem definitions

The item pricing problem is also called the VERTEX-PRICING problem; it can be defined on a graph where each customer is corresponding to a hyperedge and each item to price is corresponding to a vertex. Let us start by formally define the following VERTEX-PRICING problem.

Definition 8.1.1. *(VERTEX-PRICING) A vertex pricing problem is specified by the tuple*

$$(G(V,E), \{b_e \mid e \in E\}))$$

Here $G(V,E)$ is a multigraph where each vertex $v_i \in V$ represents an item. Each hyperedge $e \in E$ represents a set of items (vertices) that a particular customer is interested with the budget b_e.

When the corresponding graph is k-hypergraph (i.e., each customer is interested in at most k items), we call the problem VERTEX-PRICING$_k$.

Definition 8.1.2. *Given a VERTEX-PRICING instance $\mathscr{I}$, and a price function $p : V \to \mathbb{R}$, the profit is defined as follows:*

$$\mathbf{profit}_{\mathscr{I}}(p) = \sum_{b_e \geq price(e)} \mathbf{price}(e)$$

where $\mathbf{price}(e) = \sum_{v \in e} p(v)$.

When we restrict the range of the price function p, we get the positive price model, as well as the discount model and B-bounded model that is introduced in [56]

Definition 8.1.3. *Given a instance $\mathscr{I}$ of VERTEX-PRICING:*

For the positive price model, the objective function is

$$\mathrm{Opt}_{pos} = \max_{p:V \to R^+} \mathbf{profit}_{\mathscr{I}}(p).$$

For the discount model, the objective function is

$$\mathrm{Opt}_{disc} = \max_{p:V \to R} \mathbf{profit}_{\mathscr{I}}(p)$$

For the B-bounded coupon model, the objective function is

$$\mathrm{Opt}_B = \max_{p:V \to [-B,\infty)} \mathbf{profit}_{\mathscr{I}}(p)$$

The B-bounded model applies to the case that each item has the same margin cost B and the seller could not price the profit margin below $-B$. The authors in [56] also defined the coupon model which assumes that the profit is at least 0 for each sale with the customer.

Definition 8.1.4. *Given a instance $\mathscr{I}$ of VERTEX-PRICING, the profit under coupon model is defined as*

$$\mathbf{profit}^+_{\mathscr{I}}(p) = \sum_{b_e \geq price(e)} \max(\mathbf{price}(e), 0)$$

159

and the objective function is the following:

$$\text{Opt}_{coup} = \max_{p:V \to \mathbb{R}} \textbf{profit}^+(p)$$

It is easy to see the following relationship among these models.

Fact 8.1.5. *For any $B > 0$ and a* VERTEX-PRICING *instance $\mathscr{I}$,*

$$\text{Opt}_{pos} \leq \text{Opt}_B \leq \text{Opt}_{disc} \leq \text{Opt}_{coup}.$$

weighted v.s. unweighted instance we can also define weighted version of the above vertex pricing problem. The difference is that every edge has a weight w_e and **profit**(p) is defined to be

$$\sum_{b_e \geq price(e)} w_e \cdot \textbf{price}(e).$$

Similar change is made to **profit**$^+(p)$.

As is shown in [93] (Lemma 2.2), the unweighted VERTEX-PRICING has the same approximability as the weighted VERTEX-PRICING.[1] In the rest of the thesis, we only prove the hardness results for weighted VERTEX-PRICING while the same hardness result also hold for unweighted VERTEX-PRICING.

8.1.3 Main result

Our main result is the following theorem:

Theorem 8.1.6. *Assuming the UGC, given a* VERTEX-PRICING$_3$ *instance. Then for any positive integer B, it is NP-hard to distinguish the following two cases:*

- *$\text{Opt}_B \geq \Omega(\log B)$;*
- *$\text{Opt}_{coup} \leq 25$.*

Using fact (8.1.5) and taking $B = 2^{\Omega(\alpha)}$, we get the following corollaries:

Corollary 8.1.7. *Assuming the Unique Games Conjecture, for any constant $\alpha > 0$,* VERTEX-PRICING$_3$ *under the coupon model is NP-hard to α-approximate.*

Corollary 8.1.8. *Assuming the Unique Games Conjecture, for any constant $\alpha > 0$,* VERTEX-PRICING$_3$ *under the discount model is NP-hard to α-approximate.*

Corollary 8.1.9. *Assuming the Unique Games Conjecture,* VERTEX-PRICING$_3$ *under the B-bounded model is NP-hard to $\Omega(\log B)$-approximate.*

8.2 Preliminaries

8.2.1 Dictator Test for vertex pricing

VERTEX-PRICING$_3$ as a 3-CSP The VERTEX-PRICING$_3$ problem can be viewed as a 3-CSP over a set of variables $p_1, p_2, \ldots p_n$ and a set of constraints specified by b_{ijk} with weight

---

[1]Although the original proof only applies to VERTEX-PRICING $_2$ with positive price, it is straightforward to adapt their proof for our problem: VERTEX-PRICING$_3$ with arbitrary price.

w_{ijk}[2]. Let us first think of the VERTEX-PRICING problem under the discount model, the payoff function on b_{ijk} is

$$\textbf{revenue}(p_i, p_j, p_k, w_{ijk}) = \mathbf{1}(p_i + p_j + p_k \le b_{ijk})(p_i + p_j + p_k).$$

The goal is to find $p : [n] \to \mathbb{R}$ to maximize the overall profit:

$$\sum_{i,j,k} w_{ijk} \cdot \textbf{revenue}(p_i, p_j, p_k, b_{ijk}).$$

By the rule of thumb, we need to design a Dictator Testof the following form. It is a test for functions $f : [p]^n \to \mathbb{R}$ with the following two properties: (i) *Dictator functions* — i.e., functions of the form $h(x_i)$ for a particular function $h : [p] \to \mathbb{R}$ and each $i \in [n]$ — pass the test with high $\textbf{profit}_{\mathcal{T}}(f) = c$;[3] (ii) Functions f that is of "low noisy influence" on each coordinate pass the test with low $\textbf{profit}_{\mathcal{T}}(f) = s$. Then roughly speaking, by the technique of [99], we can show that assuming the UGC, it is NP-hard to distinguish whether a VERTEX-PRICING$_3$ instance with profit above c or below s (which directly implies a hardness of approximation ratio s/c).

Above is the description of the Dictator Testfor the discount model. As for the coupon model, the Dictator Testis essentially of the same except the pay off function is defined as

$$\textbf{revenue}^+(p_i, p_j, p_k, w_{ijk}) = \mathbf{1}(p_i + p_j + p_k \le w_{ijk}) \cdot \max(p_i + p_j + p_k, 0).$$

and the profit of a function f is defined as

$$\textbf{profit}_{\mathcal{T}}^+(f) = \mathbf{E}_{x,y,z,w}[\textbf{revenue}^+(f(x) + f(y) + f(z), w)].$$

In the rest of the work, we first design and analyze a proper Dictator Testfor VERTEX-PRICING$_3$. Then we use the idea from [99] to construct a reduction from the UNIQUE-GAMES problem to the VERTEX-PRICING problem. We want to emphasize here that we can not directly use [99] as the variables in VERTEX-PRICING is unbounded. To circumvent that, we need to modify the definition of "low noisy influence" function correspondingly.

8.2.2 Mathematical tools

One major advanced tool we need in our analysis is the following theorem that is essentially similar to invariance principle stated in Chapter 2.

Theorem 8.2.1. *Let* $(\Omega = [p]^t, \mu)$ *be a finite probability spaces with the following properties:*
- $a = (a_1, a_2, \ldots, a_t) \sim \mu$ *are* pairwise independent.
- $\alpha = \min_{a \in \Omega} \mu(a) > 0$.

For $\eta > 0$ *and* $f = (f^1, \ldots f^t) : \Omega^n \to [0,1]^t$ *be function satisfying that for any* $i \in [n], j \in [k]$ *and some constant* $\tau > 0$,

$$\mathrm{Inf}_i^{1-\eta} f^j \le \tau$$

---

[2]strictly speaking, for each (i,j,k), there can be different b_{ijk} with different weights w_{ijk}.
[3]usually $h(t) = t$ for most of the other results in the thesis.

Then

$$\mathbf{E}[\prod_{i=1}^{t} T_{1-\eta}f^{(i)}] - \prod_{i=1}^{t} \mathbf{E}[f^{(i)}] \le \tau^{C_0\eta/\log(1/\alpha))}$$

Here C_0 is a constant that only dependent on t. The expectation is taken with respect to the product distribution $(\Omega,\mu)^n$.

Roughly speaking, above theorem states that for calculating the product of t different functions, if these functions do not have big noisy influence on each coordinate, then the product of them is the essentially the same under an pairwise independent distribution or the fully independent distribution.

8.3   Dictator Test for vertex pricing

8.3.1   Description of the Dictator Test

To introduce our Dictator Testas well as analyzing it, first let us define the following distributions $\mathscr{D}_0,\mathscr{D}_1,\mathscr{D}_2$ on $(x,y,z) \in [p]^n \times \prod_{i=1}^n \times \prod_{i=1}^n$.

Definition 8.3.1. *(Distribution $\mathscr{D}_0$) Choose x,y uniform randomly and independently from $\prod_{i=1}^n$; for each i, we have that*

- $z_i = p - (x_i + y_i)$ *if* $x_i + y_i < p$.
- $z_i = 2p - (x_i + y_i)$ *if* $p \le x_i + y_i \le 2p$.

By definition, we know that $x_i + y_i + z_i = 0 \mod p$ for each i. One important property of above distribution is that (x_i,y_i,z_i) for each i are *pairwise independent*.

Definition 8.3.2. *(Distribution $\mathscr{D}_1$) For $x,y,z \sim \mathscr{D}_0$„ Let x',y',z' be $1-\epsilon$ correlated with x,y,z. We call the corresponding distribution on x',y',z' as $\mathscr{D}_1$*

Definition 8.3.3. *(Distribution $\mathscr{D}_2$) Choose x,y,z uniform randomly and independently from $\prod_{i=1}^n$.*

Following is the Dictator Testfor vertex pricing. Here, we use $\mathbf{1}$ to indicate the all "1" vector: $(1,1,\ldots,1) \in \mathbb{R}^n$.

Definition 8.3.4. *(Dictator Test $\mathscr{T}$) For x',y',z' generated from $\mathscr{D}_1$, a k randomly chosen from $[\sqrt{p}]$, we generate a* VERTEX-PRICING *constraint among $f(x'),f(y'),f(z' \oplus_p \lfloor \sqrt{p}/k \rfloor \cdot \mathbf{1})$ with budget $\lfloor \sqrt{p}/k \rfloor$. We define*

$$\mathbf{profit}_{\mathscr{T}}(f) = \mathbf{E}_{x',y',z',k}[\mathbf{revenue}\big((f(x'),f(y'),f(z' \oplus_p \lfloor \sqrt{p}/k \rfloor \cdot \mathbf{1}),\lfloor \sqrt{p}/k \rfloor)\big].$$

and

$$\mathbf{profit}^+_{\mathscr{T}}(f) = \mathbf{E}_{x',y',z',k}[\mathbf{revenue}^+\big((f(x'),f(y'),f(z' \oplus_p \lfloor \sqrt{p}/k \rfloor \cdot \mathbf{1}),\lfloor \sqrt{p}/k \rfloor)\big].$$

For the purpose of analyzing $\mathscr{T}$, we also define the following Test $\mathscr{T}'$.

Definition 8.3.5. *(Test $\mathscr{T}'$) For x',y',z' generated from $\mathscr{D}_2$; randomly choose $k \in [\sqrt{p}]$. We generate a* VERTEX-PRICING *constraint among $f(x'),f(y'),f(z')$ with budget $\lfloor \sqrt{p}/k \rfloor$).*
We define

$$\mathbf{profit}_{\mathscr{T}'}(f) = \mathbf{E}_{x',y',z',k}[\mathbf{revenue}\big((f(x'),f(y'),f(z'),\lfloor \sqrt{p}/k \rfloor)\big].$$

162

and

$$\textbf{profit}^+_{\mathcal{T}'}(f) = \mathbf{E}_{x',y',z',k}[\textbf{revenue}^+((f(x'),f(y'),f(z'),\lfloor\sqrt{p}/k\rfloor)].$$

We claim that for the Dictator Test $\mathcal{T}'$, it has the following property:

Proposition 8.3.6. *For any function $f:[p]^n \to \mathbb{R}$, $\textbf{profit}^+_{\mathcal{T}'}(f) \le 1$.*

Proof. Notice that for each triple (x',y',z'), if there exists k' such that $\lfloor\sqrt{p}/(k'+1)\rfloor \le f(x')+f(y')+f(z') \le \lfloor\sqrt{p}/k'\rfloor$. Then the profit on (x',y',z') is at most

$$k'(f(x)+f(y)+f(z)) \le k'\sqrt{p}/k'/\sqrt{p} \le 1.$$

If $f(x')+f(y')+f(z') \le 0$ or $f(x')+f(y')+f(z') \ge \sqrt{p}$, then the profit on x',y',z' is 0.

Condition on every triple (x',y',z'), the expect profit associated with $f(x'),f(y'),f(z')$ is at most 1, therefore the overall profit is also at most 1. $\qquad\square$

8.3.2 Analysis of the Dictator Test $\mathcal{T}$

We prove the completeness (Theorem 10.3.5) and soundness (Theorem 10.3.6) for $\mathcal{T}$ in this section.

Theorem 8.3.7. *(Completeness of $\mathcal{T}$) For function $f(x) = x_i - p/3$ for $x \in \prod_{i=1}^n$, $\textbf{profit}_{\mathcal{T}}(f) \ge \Omega(\log p)$.*

Proof. Suppose $x',y',z' \sim \mathcal{D}_1$ is generated as $1-\epsilon$ copy of $x,y,z \sim \mathcal{D}_0$.

Since x_i,y_i are randomly generated from $[p]$, we know that $\sqrt{p} \le x_i + y_i \le p$ with probability at least $1/3$. When this happen, $x_i + y_i + z_i = p$ and $z_i \le p - \sqrt{p}$. Also as each of the x_i,y_i,z_i is reset to a random number with probability $\epsilon = 1/p$, we know that with probability $1/3 - 3/p$, $x'_i = x_i, y'_i = y_i, z'_i = z_i$ and we have that $x'_i + y'_i + z'_i = p$ and $z'_i \le p - \sqrt{p}$. We call these (x',y',z') "good".

Then for "good" (x',y',z'), if we choose $f(t) = x_i - p/3$, we know that $f(x')+f(y')+f(z' \oplus_p \lfloor\sqrt{p}/k\rfloor) = x_i + y_i + z_i + \lfloor\sqrt{p}/k\rfloor - p = \lfloor\sqrt{p}/k\rfloor$. Therefore,

$$\textbf{revenue}((f(x'),f(y'),f(z' \oplus_p \lfloor\sqrt{p}/k\rfloor \cdot \mathbf{1}),\lfloor\sqrt{p}/k\rfloor) = \lfloor\sqrt{p}/k\rfloor.$$

Therefore for "good" (x',y',z'), the associate is at least

$$(1/3 - 3/p) \cdot \frac{\sum_{k=1}^{\sqrt{p}}\lfloor\sqrt{p}/k\rfloor}{\sqrt{p}} \ge (1/3 - 3/p) \cdot \frac{\sum_{k=1}^{\sqrt{p}}(\sqrt{p}/k - 1)}{\sqrt{p}} \ge (1/3 - 3/p) \cdot (\log\sqrt{p} - 1) \ge 1/8\log p$$

for large enough p.

We also need to show bound the profit (loss) on those "bad" x',y',z' such that for some k

$$f(x')+f(y')+f(z' \oplus_p \lfloor\sqrt{p}/k\rfloor) < 0.$$

This could happen for x',y',z' generated from the following two cases:

1. At least one of the x'_i,y'_i,z'_i is reset, this happens with probability at most $3/p$.
2. None of the x'_i,y'_i,z'_i is not reset. Since $x_i + y_i + z_i = p, 2p$, to make $f(x'_i)+f(y'_i)+f(z' \oplus_p \sqrt{p}/k) < 0$, we know that we must have $x_i + y_i + z_i = p$ and $z_i > p - \sqrt{p}/k$. We must then have $x_i + y_i \le \sqrt{p}/k$. We know that $\mathbf{Pr}(x_i + y_i \le \sqrt{p}/k \le \mathbf{Pr}(x_i,y_i \le \sqrt{p}/k) = \frac{1}{pk^2}$.

Therefore, we can have negative profit on (x', y', z') occur with probability at most $4/p$. As we know that $f(x) = x_i - p/3 \geq -p/3$, therefore, $f(x') + f(y') + f(z' + \lfloor\sqrt{p}/k\rfloor) \geq -p$, overall, we lose at most $4/p \cdot p = -4$ on those "bad" (x', y', z').

Overall, for $f(x) = x_i - p/3$, we must have that $\mathbf{profit}_{\mathcal{T}}(f) \geq 1/8 \cdot \log p - 4 = \Omega(\log p)$ for sufficient large p. $\qquad\square$

Now we state the soundness statement. As $f : [p]^n \to R$ is not bounded, we define its influence on a transformation of f as follows. We define $\tilde{f}$ be the integral part of f, being $\lfloor f \rfloor$. We also define $f' \in [p]$ and is uniquely defined by $f' = \tilde{f} \mod p$. By abuse of the notation, we also write $f' : [p]^n \to \{-1, 1\}^p$ with $f'^{(i)}$ being the indicator function $\mathbf{1}(\tilde{f} = i \mod p)$. The influence of f' is defined with respect to its vector form.

Theorem 8.3.8. *(Soundness of $\mathcal{T}$) For $\tau^{C_0 1/p \log(p)} \leq 1/p^4$ and any function $f : [p]^n \to \mathbb{R}$ such that*

$$\max_i \mathrm{Inf}_i^{1-\epsilon} f' \leq \tau,$$

we have that $\mathbf{profit}_{\mathcal{T}}^+(f) < 7$.

Proof. Notice that the soundness statement is proved for the coupon model which automatically gives an upper bound for $\mathbf{profit}_{\mathcal{T}}(f)$.

First let us prove above statement under the assumption that $f \in [p]$. Then $f'^{(i)} = \mathbf{1}(f = i)$. We also use μ_a to denote $\mathbb{E}_{x \in [p]^n}[f'^a(x)]$. We can arithmetize and bound the objective function $\mathbf{profit}_{\mathcal{T}}^+(f)$ in terms of $f'^{(i)}$ as follows:

$$\mathbf{profit}_{\mathcal{T}}^+(f) = \sum_{0 \leq a+b+c \leq \lfloor\sqrt{p}/k\rfloor, a,b,c \in [p]} \mathop{\mathbb{E}}_{x',y',z' \sim \mathcal{D}_{2,k}} [f'^a(x') f'^b(y') f'^c(z' \oplus_p \lfloor\sqrt{p}/k\rfloor \cdot \mathbf{1})(a+b+c)]$$

$$= \mathop{\mathbb{E}}_{x,y,z \sim \mathcal{D}_{1,k}} [\sum_{0 \leq a+b+c \leq \lfloor\sqrt{p}/k\rfloor, a,b,c \in [p]} T_{1-\epsilon} f'^a(x) T_{1-\epsilon} f'^b(y) T_{1-\epsilon} f'^c(z \oplus_p \lfloor\sqrt{p}/k\rfloor \cdot \mathbf{1})(a+b+c)]$$

$$= \mathop{\mathbb{E}}_k [\sum_{0 \leq a+b+c \leq \lfloor\sqrt{p}/k\rfloor, a,b,c \in [p]} \mathop{\mathbb{E}}_{x,y,z \sim \mathcal{D}_1} T_{1-\epsilon} f'^a(x) T_{1-\epsilon} f'^b(y) T_{1-\epsilon} f'^c(z \oplus_p \lfloor\sqrt{p}/k\rfloor \cdot \mathbf{1})(a+b+c)].$$

Notice that $\mathrm{Inf}_i^{1-\epsilon} f'^a \leq \mathrm{Inf}_i^{1-\epsilon} f' \leq \tau$ for $i \in [n], a \in [p]$. and $x, y, z \sim \mathcal{D}_1$ are pairwise independent, by Theorem 8.2.1 (with minimum probability $\alpha = 1/p$), we can plug in independent $x, y, z \sim \mathcal{D}_2$ with additive error bounded by $\tau^{C_0 1/(p \log p)} \leq 1/p^4$. That is

$$\mathbf{profit}_{\mathcal{T}}^+(f) < \mathop{\mathbb{E}}_k [\sum_{0 \leq a+b+c \leq \lfloor\sqrt{p}/k\rfloor, a,b,c \in [p]} \mathop{\mathbb{E}}_{x,y,z \sim \mathcal{D}_2} \left(f'^a(x) f'^b(y) f'^c(z \oplus_p \lfloor\sqrt{p}/k\rfloor \cdot \mathbf{1}) + 1/p^4 \right)(a+b+c)]$$

$$\leq \mathbb{E}_{k \in [\sqrt{p}]} [\sum_{0 \leq a+b+c \leq \lfloor\sqrt{p}/k\rfloor, a,b,c \in [p]} \mu_a \mu_b f'^c(z \oplus_p \lfloor\sqrt{p}/k\rfloor \cdot \mathbf{1})(a+b+c)] + 1$$

The last inequality uses the fact that $a + b + c \leq \sqrt{p}$ and there are at most p^3 terms in the summation.

A important observation is that since z is independent of k, therefore the random vector variable $z \oplus_p \lfloor\sqrt{p}/k\rfloor$ is also independent of the random variable k. Also, the distribution on $z \oplus_p \lfloor\sqrt{p}/k\rfloor$ is uniformly random over $[p]^n$.

Therefore, we can further bound $\mathbf{profit}_{\mathcal{T}}^+(f)$ by

$$\mathbb{E}_k [\sum_{0 \leq a+b+c \leq \lfloor\sqrt{p}/k\rfloor, a,b,c \in [p]} \mu_a \mu_b \mu_c (a+b+c) + 1.$$

As for the term

$$\mathbf{E}_{k\in[\sqrt{p}]}\Big[\sum_{0\le a+b+c\le\lfloor\sqrt{p}/k\rfloor, a,b,c\in[p]}\mu_a\mu_b\mu_c(a+b+c).$$

It is just $\mathbf{profit}^+_{\mathcal{T}'}(f)$. By Proposition 8.3.6, we know that $\mathbf{profit}^+_{\mathcal{T}'}(f)\le 1$. Overall, we bound $\mathbf{profit}^+_{\mathcal{T}}(f)$ by 2 when $f\in[p]$.

Following two observation is useful in our analysis.

Observation 8.3.9. *Above proof also works even for randomized function $f(x)\in[p]$ specified by f' in the following way: for each x, with probability $f'^i(x)$, f outputs i. Here $\sum f'^i(x)=1$ for any $x\in[p]^n$.*

Observation 8.3.10. *For any $\theta\in\mathbb{R}, f\in[p]$, we can also bound the profit on function $f-\theta$. That is $\mathbf{profit}^+_{\mathcal{T}}(f-\theta)\le 2$.*

To see this, simply notice that

$$\mathbf{profit}^+_{\mathcal{T}}(f-\theta)=\sum_{3\theta\le a+b+c\le 3\theta+\lfloor\sqrt{p}/k\rfloor, a,b,c\in[p]}\underset{x',y',z'\sim\mathscr{D}_{2,k}}{\mathbf{E}}[f^a(x')f^b(y')f^c(z'\oplus_p\lfloor\sqrt{p}/k\rfloor\cdot\mathbf{1})(a+b+c-3\theta)]$$

and then we use the same proof and show that $\mathbf{profit}^+_{\mathcal{T}}(f-\theta)\le\mathbf{profit}^+_{\mathcal{T}'}(f-\theta)+1\le 2$.

Now we need to handle the case that f is not necessary a bounded integral function. Recall that $\tilde{f}=\lfloor f\rfloor$. First, we notice that $f\le\tilde{f}+1$, therefore, we have that

$$\mathbf{revenue}^+\big((f(x'),f(y'),f(z'\oplus_p\lfloor\sqrt{p}/k\rfloor\cdot\mathbf{1}),\lfloor\sqrt{p}/k\rfloor\big)$$
$$\le\mathbf{revenue}^+\big((\tilde{f}(x'),\tilde{f}(y'),\tilde{f}(z'\oplus_p\lfloor\sqrt{p}/k\rfloor\cdot\mathbf{1}),\lfloor\sqrt{p}/k\rfloor\big)+3. \quad(8.1)$$

We then have that

$$\mathbf{profit}^+_{\mathcal{T}}(f)=\mathbf{E}_{x',y',z',k}[\mathbf{revenue}^+\big((f(x'),f(y'),f(z'\oplus_p\lfloor\sqrt{p}/k\rfloor\cdot\mathbf{1}),\lfloor\sqrt{p}/k\rfloor\big)]$$
$$\le\mathbf{E}_{x',y',z',k}[\mathbf{revenue}^+\big((\tilde{f}(x'),\tilde{f}(y'),\tilde{f}(z'\oplus_p\lfloor\sqrt{p}/k\rfloor\cdot\mathbf{1}),\lfloor\sqrt{p}/k\rfloor\big)+3]\le\mathbf{profit}^+_{\mathcal{T}}(\tilde{f})+3.$$

The next step we show that

$$\mathbf{profit}^+_{\mathcal{T}}(\tilde{f})\le\mathbf{profit}^+_{\mathcal{T}}(f')+\mathbf{profit}^+_{\mathcal{T}}(f'-p/3)+\mathbf{profit}^+_{\mathcal{T}}(f'-2p/3) \quad(8.2)$$

By definition of f', we know that

$$\tilde{f}(x)+\tilde{f}(y)+\tilde{f}(z)=f'(x)+f'(y)+f'(z)\mod p.$$

Therefore, if $\tilde{f}(x)+\tilde{f}(y)+\tilde{f}(z)\le\lfloor\sqrt{p}/k\rfloor$ for some k, it must be the case that

$$f'(x)+f'(y)+f'(z)\in\big[0,\lfloor\sqrt{p}/k\rfloor\big]\text{ or }\big[p,p+\lfloor\sqrt{p}/k\rfloor\big]\text{ or }\big[2p,2p+\lfloor\sqrt{p}/k\rfloor\big].$$

Therefore,

$$\mathbf{revenue}^+(\tilde{f}(x),\tilde{f}(y),\tilde{f}(z),\lfloor\sqrt{p}/k\rfloor)\le\mathbf{revenue}^+(f'(x),f'(y),f'(z),\lfloor\sqrt{p}/k\rfloor))$$
$$+\mathbf{revenue}^+(f'(x)-p/3,f'(y)-p/3,f'(z)-p/3,\lfloor\sqrt{p}/k\rfloor))$$
$$+\mathbf{revenue}^+(f'(x)-2p/3,f'(y)-2p/3,f'(z)-2p/3,\lfloor\sqrt{p}/k\rfloor)).$$

This proves (8.2). And by Observation 8.3.10, we have that

$$\mathbf{profit}^+_{\mathcal{T}}(f)<\mathbf{profit}^+_{\mathcal{T}}(\tilde{f})+1\le 3\cdot 2+1\le 7.$$

$\square$

165

8.4 The reduction from the UNIQUE-GAMES

In this section we show how to use our Dictator Test $\mathcal{T}$ to obtain our main result, Theorem 9.1.1. First let us recall the definition of the UNIQUE-GAMES.

Definition 8.4.1. *For $L \in \mathbb{N}$, a* UNIQUE-GAMES$_L$ *instance consists of a bipartite graph having vertex sets U, V and edge set E, together with a bijective constraint $\pi^{v,u} : [L] \rightarrow [L]$ for each $(u,v) \in E$. In addition, each edge $e \in E$ has a nonnegative weight p_{uv}, with $\sum_{(u,v) \in E} p_{uv} = 1$. The algorithmic task is to find an assignment $A : (U \cup V) \rightarrow [L]$ such that the total weight of satisfied constraints is as large as possible. Here we say that A satisfies the constraint π^{uv} if $\pi^{uv}(A(u)) = A(v)$.*

The following equivalent version of the UGC due to Khot and Regev [103, Lemma 3.6]:

Theorem 8.4.2. *Assume the UGC. For all small $\zeta, \gamma > 0$, there exists $L \in \mathbb{N}$ such given an unweighted* UNIQUE-GAMES$_L$ *instance $\mathcal{G} = (U, V, E, (\pi^{u,v})_{(u,v) \in E})$ which is U-regular, it is* NP-*hard to distinguish the following two cases:*

1. *There is an assignment $A : (U \cup V) \rightarrow [L]$ and a subset $U' \subseteq U$ with $|U'|/|U| \geq 1 - \zeta$ such that A satisfies all constraints incident on U'.*

2. *There is no assignment A that satisfies more than γ fraction of the constraints.*

We make the following reduction from a UNIQUE-GAMES instance $\mathcal{G}$ to a VERTEX-PRICING$_3$ instance $\mathcal{I}$. The reduction is very similar to the one in [99]. Given the UNIQUE-GAMES$_L$ instance $\mathcal{G} = (U, V, E, \{\pi^{uv}\})$, the reduction produces a weighted VERTEX-PRICING instance $\mathcal{I}$ with variable set $V \times [p]^L$. We think of an price assignment F to these variables as a collection of functions $F = \{f_v : [p]^L \rightarrow \mathbb{R}\}$, one for each $v \in V$. We now define the instance according to the following procedures.

Reduction from UNIQUE-GAMES
1. Choose $u \in U$ randomly.
2. Choose 3 of u's neighbor v_1, v_2, v_3 randomly (with replacement).
3. Generate $(x, y, z) \sim \mathcal{D}_2$ and k randomly from $[\sqrt{p}]$.
4. Add a constraint among $f_{v_1}(\pi^{v_1,u}(x)), f_{v_2}(\pi^{v_2,u}(y)), f_{v_3}(\pi^{v_3,u}(z) + \lfloor \sqrt{p}/k \rfloor)$ with budget $\lfloor \sqrt{p}/k \rfloor$.

Here, for $x \in [p]^L$ and mapping $\pi : [L] \rightarrow [L]$, we denote $\pi(x) \in [p]^L$ as the permutation of x's coordinate according to i; i.e., $\pi(x)_i = x_{\pi(i)}$.

We claim that above reduction have the following property.

Theorem 8.4.3. *For $\zeta = 1/p$, τ satisfies that $\tau^{C_0 p \log p} \leq 1/p^4$ and $\gamma = \tau^2/p^4$, above reduction have the following property:*

- *(Completeness.) If statement 1 in Theorem 8.4.2 holds for $\mathcal{G}$, then there is a price assignment F such that $\mathbf{profit}_{\mathcal{I}}(F) = \Omega(\log p)$. In addition, the price assigned on each variable is p-bounded, i.e., with value $\geq -p$.*

- *(Soundness.) If there is non assignment for $\mathcal{G}$ that satisfies more than γ fraction of the edges, then for every price assignment F such that $\mathbf{profit}_{\mathcal{I}}^+(F) \leq 25$.*

By combining Theorem 8.4.3 with Theorem 8.4.2, and set $p = B$, we prove Theorem 9.1.1.

8.4.1 Proof of Theorem 8.4.3

It remains to prove Theorem 8.4.3.

Proof. (Completeness) To prove the completeness part of Theorem 8.4.3, suppose that assignment $A : V \to [L]$ and subset $U' \subseteq U$ are as in statement 1 of Theorem 8.4.2. Define an price assignment F for $\mathscr{I}$ by taking $f_v(x) = x_{A(v)} - p/3$. Then by definition and the property of A, for $u' \in U'$, $f_{v_i}(\pi^{v_i,u'}(x)) = x_{A(u')} - p/3$ for $i = 1,2,3$. Thus by the completeness of the Dictator Test(Theorem 10.3.5), assignment F will have profit at least $\Omega(\log p)$ conditioned on $u' \in U$ is picked. As for the case that $u \notin U'$ is picked, we lose a negative profit bounded by $-p$. Overall, we have that $\mathbf{profit}_{\mathscr{I}}(F) \geq (1 - \zeta)\Omega(\log P) + \zeta p$. Notice that we choose $\zeta = 1/p$, therefore, $\mathbf{profit}_{\mathscr{I}}(F) \geq \Omega(\log p)$. In addition, we know that the assignment on each f_v is above $-p/3$.

(Soundness) We prove the soundness statement by contradiction. Suppose that some assignment F have $\mathbf{profit}_{\mathscr{I}}^+(F) \geq 25$, we will exhibit a assignment to the Unique Games instance $\mathscr{G}$ that satisfies γ fraction of the edges. Notice that the maximum profit on each constraint is at most $\sqrt{p}$, then by an average argument, we must have for at least $1/\sqrt{p}$ of the vertex $u \in U$ picked in the first step, the expected profit on these u is above 24.

Let us call these u "good". Write $N(u)$ as the neighbor of u. By definition, for a fixed "good" u, we know that

$$\mathbf{E}_{v_1,v_2,v_3 \in N(u),x,y,z \sim \mathscr{D}_2}[\mathbf{revenue}^+ \left(f_{v_1}(\pi^{u,v_1}x), f_{v_2}(\pi^{u,v_2}(y)), f_{v_3}(\pi^{u,v_3}(z) + \lfloor \sqrt{p}/k \rfloor), \lfloor \sqrt{p}/k \rfloor)\right] \geq 24.$$

Similar to the analysis of Theorem 10.3.6 , we define $\tilde{f}_v = \lfloor f_v \rfloor$ and introduce $f'_v \in [p]$ such that $f'_v = \tilde{f}_v \mod p$, although we also write f'_v as $[p]^n \to \{0,1\}^p$ with its i-th coordinate indicate whether f'_v is i. We call the assignment corresponding to $\{\tilde{f}_v\}_{v \in V}$ as $\tilde{F}$ and the assignment corresponding to $\{f'_v\}_{v \in V}$ as F'.

By the proof of (8.1), we know that

$$\mathbf{profit}_{\mathscr{G}}^+(\tilde{F}) \geq \mathbf{profit}_{\mathscr{G}}^+(F) - 3 \geq 21$$

and by the proof of (8.2), we have that

$$\mathbf{profit}_{\mathscr{G}}^+(F') + \mathbf{profit}_{\mathscr{G}}^+(F' - p/3) + \mathbf{profit}_{\mathscr{G}}^+(F' - 2p/3) \geq \mathbf{profit}_{\mathscr{G}}^+(\tilde{F}) \geq 21.$$

Therefore, one of $\mathbf{profit}_{\mathscr{G}}^+(F'), \mathbf{profit}_{\mathscr{G}}^+(F' - p/3), \mathbf{profit}_{\mathscr{G}}^+(F' - 2p/3)$ should be above 7.

Assume that $\mathbf{profit}_{\mathscr{G}}^+(F' - p/3) \geq 7$. (The other 2 cases are similar)

We know then

$$\mathbf{profit}_{\mathscr{G}}^+(F' - p/3)$$
$$= \mathbf{E}_{x,y,z,k,,v_1,v_2,v_3}[\sum_{p < a+b+c \leq p + \lfloor \sqrt{p}/k \rfloor}[f_{v_1}^a(\pi^{v_1,u}(x))f_{v_2}^b(\pi^{v_2,u}(y))f_{v_3}^c(\pi^{v_3,u}(z) + \lfloor \sqrt{p}/k \rfloor \cdot \mathbf{1})(a+b+c-p)]$$
$$= \mathbf{E}_{x,y,z,k}[\sum_{p < a+b+c \leq p + \lfloor \sqrt{p}/k \rfloor} \mathbf{E}_{v_1 \in N(u)}[f_{v_1}^a(\pi^{v_1,u}(x))] \cdot \mathbf{E}_{v_2 \in N(u)}[f_{v_2}^b(\pi^{v_2,u}(y))] \cdot$$
$$\mathbf{E}_{v_3 \in N(u)}[f_{v_3}^c(\pi^{v_3,u}(z + \lfloor \sqrt{p}/k \rfloor \cdot \mathbf{1}))](a + b + c - p)] \quad (8.3)$$

167

If we define $f_u^i = \mathbf{E}_{v \in N(u)}[f_v^i(\pi^{v,u}(x))]$ for $i \in [p]$, then we have that

$$\mathbf{profit}_{\mathscr{I}}^+(F' - p/3) = \mathbf{E}_{x,y,z,k}[\sum_{p < a+b+c \leq p+\lfloor \sqrt{p}/k \rfloor} \mathbf{E}_{v_1,v_2,v_3}[f_u^i(\pi^{v_1,u}(x))$$

$$\cdot f_u^i(\pi^{v_2,u}(y))f_u^i(\pi^{v_3,u}(z)) + \lfloor \sqrt{p}/k \rfloor \cdot \mathbf{1})(a+b+c-p)] \geq 7. \quad (8.4)$$

Denote $f_u(x) = (f_u^1(x), f_u^2(x), \ldots, f_u^p(x))$. It is easy to check that $\sum f_u^i = 1$. Then f_u can be viewed as a randomized function that on a particular x, it output i with probability $f_u^i(x)$. Then (8.4) is equal to the profit of the Dictator Test $\mathscr{T}$ on $f_u - p/3$, being $\mathbf{profit}_{\mathscr{T}}^+(f_u - p/3) \geq 7$.

We know then by a contrapositive statement of Theorem 10.3.6 along with observation 8.3.10 and Observation 8.3.9, there must be some i such that $\mathrm{Inf}_i^{1-\epsilon} f_u \geq \tau$.

Then by Fact 7.3.3, we know that

$$\tau \leq \mathrm{Inf}_i^{1-\epsilon} f_u \leq \mathbf{E}_{v \in N(u)}[\mathrm{Inf}_i^{1-\epsilon} f_v(\pi^{v,u}(x))]$$

By an averaging argument, since $\mathrm{Inf}_i f_v(\pi^{v,u}(x)) = \sum_{j \in [p]} \mathrm{inf}_i f_v^j \leq p$, for $\frac{\tau}{2p}$ fraction of the $v \in N(u)$, we have that $\mathrm{Inf}_i f_v(\pi^{v,u}(x)) = \mathrm{Inf}_{j=(\pi^{v,u})^{-1}(i)}^{1-\epsilon} f_v \geq \tau/2$.

Now consider choosing a random assignment. Let $S_u = \{i \mid \mathrm{Inf}_i^{1-\epsilon} f_u \geq \tau\}$ and S_v be $\{i \mid \mathrm{Inf}_i^{1-\epsilon} f_u \geq \tau/2\}$. By fact 7.3.4, we know that $|S_v| \leq p^2/\tau$.

The assignment would be randomly set a label in S_u for u and a label in S_v for v. Then it is easy to see for good vertex u and any of its coordinate $i \in S_u$, $\tau/2p$ fraction of its neighbor will have a matching coordinate $j = (\pi^{v,u})^{-1}(i)$ in S_v. Therefore above assignment satisfy at least $1/|S_v| \cdot \tau/2p$ fraction of the edges for "good" u. We know that there is at least a $1/\sqrt{p}$ fraction of the u is good . By the regularity of the graph at the U side, we know that such a labeling strategy satisfies at least $1/\sqrt{p} \cdot (\tau/p^2)\tau/2p \geq \tau^2/p^4 = \gamma$ fraction of the edges.

$\square$

Part III

Hardness of Learning

Chapter 9

Hardness of Learning Monomials

9.1 Introduction

Monomials (conjunctions), decision lists, and halfspaces are among the most basic concept classes in learning theory. They are all long-known to be efficiently PAC learnable, when the given examples are guaranteed to be consistent with a function from any of these concept classes. However, in practice data is often noisy or too complex to be consistently explained by a simple concept. Dealing with noise and other inconsistencies is thus one of the most significant issues in learning theory. In this chapter, we prove a strong hardness result for agnostic learning of monomials using halfspaces, or equivalently the MON-HS-MA problem.

Theorem 9.1.1. *The problem of* MON-HS-MA $(1 - \epsilon, 1/2 + \epsilon)$ *is* NP-*hard.*

Note that this hardness result is essentially optimal since it is trivial to find a hypothesis with agreement rate $1/2$ — output either the function that is always 0 or the function that is always 1.

Since the class of monomials is a subset of the class of decision lists which in turn is a subset of the class of halfspaces, our result implies an optimal hardness result for proper agnostic learning of decision lists. In addition, a similar hardness result for proper agnostic learning of majority functions can be obtained via a simple reduction.

9.2 Proof Overview

By the rule of thumb, the first step is to construct a dictator test such that a dictator monomial passes with probability $1 - \epsilon$ while a non dictator test passes the test with probability $1/2 + \epsilon$.

We prove Theorem 9.1.1 by exhibiting a reduction from the k-LABEL-COVER problem, which is a particular variant of the LABEL-COVER problem. The k-LABEL-COVER problem is defined as follows:

Definition 9.2.1. *For $k \geqslant 2$, an instance of k-LABEL-COVER $\mathscr{L}(G(V,E), M, N, \{\pi^{v,e} | e \in E, v \in e\})$ consists of a k-uniform connected (multi-)hypergraph $G(V,E)$ with vertex set V and an edge set E; a set of functions $\{\pi^{v_i, e}\}_{i=1}^{k}$; and a set of labels $M = \{1, 2, \ldots, M\}$ for some positive integers M. Every hyperedge $e = (v_1, \ldots, v_k)$ is associated with a k-tuple of projection functions $\{\pi^{v_i, e}\}_{i=1}^{k}$ where $\pi^{v_i, e} : [M] \to [N]$.*

A vertex labeling $\mathscr{A}$ is an assignment of labels to vertices $\mathscr{A} : V \to [dR]$. A labeling $\mathscr{A}$ is said to strongly satisfy an edge e if $\pi^{v_i, e}(\mathscr{A}(v_i)) = \pi^{v_j, e}(\mathscr{A}(v_j))$ for every $v_i, v_j \in e$. A labeling L weakly satisfies edge e if $\pi^{v_i, e}(\mathscr{A}(v_i)) = \pi^{v_j, e}(\mathscr{A}(v_j))$ for some $v_i, v_j \in e$, $v_i \neq v_j$.

The goal in LABEL-COVER is to find a vertex labeling that satisfies as many edges (projection constraints) as possible.

For the sake of clarity, we first present the proof of Theorem 9.1.1 assuming the Unique Games Conjecture. Consequently, we will be interested in the k-UNIQUE LABEL-COVER problem which is a special case of k-LABEL-COVER where $M = N$, and all the projection functions $\{\pi^{v,e} | v \in e, e \in E\}$ are bijections. The following inapproximability result for k-UNIQUE LABEL-COVER is equivalent to the Unique Games Conjecture of Khot [98].

Conjecture 9.2.2. *For every constant $\eta > 0$ and a positive integer k, there exists R_0 such that for all positive integers $R > R_0$, given an instance $\mathscr{L}(G(V,E), 1, R, \{\pi^{v,e} | e \in E, v \in e\})$ it*

is NP-*hard to distinguish between,*

- *strongly satisfiable instances: there exists a labeling $\mathscr{A} : V \to [R]$ that strongly satisfies $1 - k\eta$ fraction of the edges E.*
- *almost unsatisfiable instances: there is no labeling that weakly satisfies $\frac{2k^2}{R^{\eta/4}}$ fraction of the edges.*

A proof of the equivalence between above conjecture and Unique Games Conjecture can be found in [103].

Given an instance $\mathscr{L}$ of k-UNIQUE LABEL-COVER, we will produce a distribution $\mathscr{D}$ over labeled examples such that the following holds: if $\mathscr{L}$ is a strongly satisfiable instance, then there is a disjunction (an OR function) that agrees with a randomly chosen example with probability at least $1 - \epsilon$, while if $\mathscr{L}$ is an almost unsatisfiable instance then no halfspace agrees with a random example from $\mathscr{D}$ with probability more than $\frac{1}{2} + \epsilon$. Clearly, such a reduction implies Theorem 9.1.1 assuming the Unique Games Conjecture but with disjunctions in place of conjunctions. De Morgan's law and the fact that a negation of a halfspace is a halfspace then imply that the statement is also true for monomials (we use disjunctions only for convenience).

Let $\mathscr{L}$ be an instance of k-UNIQUE LABEL-COVER on hypergraph $G = (V, E)$ and a set of labels $[R]$. The examples we generate will have $|V| \times R$ coordinates, i.e., belong to $\{0, 1\}^{|V| \times R}$. These coordinates are to be thought of as one block of R coordinates for every vertex $v \in V$. We will index the coordinates of $\boldsymbol{x} \in \{0, 1\}^{|V| \times R}$ as $\boldsymbol{x} = (x_v^{(\ell)})_{v \in V, \ell \in [R]}$.

For every labeling $\mathscr{A} : V \to [R]$ of the instance, there is a corresponding disjunction (OR function) over $\{0, 1\}^{|V| \times R}$ given by,

$$h(\boldsymbol{x}) = \bigvee_v x_v^{(\mathscr{A}(v))}.$$

Thus, using a label ℓ for a vertex v is encoded as including the literal $x_v^{(\ell)}$ in the disjunction. Notice that an arbitrary halfspace over $\{0, 1\}^{|V| \times R}$ need not correspond to any labeling at all. The idea would be to construct a distribution on examples which ensures that any halfspace agreeing with at least $\frac{1}{2} + \epsilon$ fraction of random examples somehow corresponds to a labeling of $\mathscr{L}$ weakly satisfying a constant fraction of the edges in $\mathscr{L}$.

Fix an edge $e = (v_1, \ldots, v_k)$. For the sake of exposition, let us assume $\pi^{v_i, e}$ is the identity permutation for every $i \in [k]$. The general case is not anymore complicated. For the edge e, we require a distribution on examples $\mathscr{D}_e$ with the following properties:

- All coordinates $x_v^{(\ell)}$ for a vertex $v \notin e$ are fixed to be zero. Restricted to these examples, the halfspace h can be written as $h(\boldsymbol{x}) = \text{sgn}(\sum_{i \in [k]} \langle \boldsymbol{w}_{v_i}, \boldsymbol{x}_{v_i} \rangle - \theta)$.
- For any label $\ell \in [R]$, the labeling $\mathscr{A}(v_1) = \ldots = \mathscr{A}(v_k) = \ell$ *strongly* satisfies the edge e. Hence, the corresponding disjunction $\vee_{i \in [k]} x_{v_i}^{(\ell)}$ needs to have agreement $\geqslant 1 - \epsilon$ with the examples from $\mathscr{D}_e$.
- There exists a decoding procedure that given a halfspace h outputs a labeling L_h for $\mathscr{L}$ such that, if h has agreement $\geqslant \frac{1}{2} + \epsilon$ with the examples from $\mathscr{D}_e$, then L_h *weakly* satisfies the edge e with non-negligible probability.

For conceptual clarity, let us rephrase the above requirement as a testing problem. Given a halfspace h, consider a randomized procedure that samples an example $(\boldsymbol{x}, b)$ from the distribution $\mathscr{D}_e$, and accepts if $h(\boldsymbol{x}) = b$. This amounts to a test that checks if

the function h corresponds to a consistent labeling. Further, let us suppose the halfspace h is given by $h(\boldsymbol{x}) = \mathrm{sgn}(\sum_{v \in V} \langle \boldsymbol{w}_v, \boldsymbol{x}_v \rangle - \theta)$. Define the linear function $l_v : \{0,1\}^R \to \mathbb{R}$ as $l_v(\boldsymbol{x}_v) = \langle \boldsymbol{w}_v, \boldsymbol{x}_v \rangle$. Then, we have $h(\boldsymbol{x}) = \mathrm{sgn}(\sum_{v \in V} l_v(\boldsymbol{x}_v) - \theta)$.

For a halfspace h corresponding to a labelling L, we will have $l_v(\boldsymbol{x}_v) = x_v^{(A(v))}$ — a *dictator* function. Formally, the ℓ'th dictator function on $\{0,1\}^R$ is given by $F(\boldsymbol{x}) = x^{(\ell)}$. Thus, in the intended solution every linear function l_v associated with the halfspace h is a *dictator* function.

Now, let us again restate the above testing problem in terms of these linear functions. For succinctness, we write l_i for the linear function l_{v_i}. We need a randomized procedure that does the following:

> Given k linear functions $l_1, \ldots, l_k : \{0,1\}^R \to \mathbb{R}$, queries the functions at one point each (say $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ respectively), and accepts if $\mathrm{sgn}(\sum_{i=1}^k l_i(\boldsymbol{x}_i) - \theta) = b$.

The procedure must satisfy,

- (Completeness) If each of the linear functions l_i is the ℓ'th dictator function for some $\ell \in [R]$, then the test accepts with probability $1 - \epsilon$.

- (Soundness) If the test accepts with probability $\frac{1}{2} + \epsilon$, then at least *two* of the linear functions are *close* to the same dictator function.

A testing problem of the above nature is referred to as a *Dictatorship Testing* and is a recurring theme in hardness of approximation.

Notice that the notion of a linear function being *close* to a dictator function is not formally defined yet. In most applications, a function is said to be close to a dictator if it has *influential* coordinates. It is easy to see that this notion is not sufficient by itself here. For example, in the linear function $\mathrm{sgn}(10^{100} x_1 + x_2 - 0.5)$, although the coordinate x_2 has little influence on the linear function, it has the significant influence on the halfspace.

We resolve this problem by using the notion of *critical index* (Definition 9.3.1) introduced in [133] and has found numerous applications in the analysis of halfspaces [41, 113, 119]. Roughly speaking, given a linear function l, the idea is to recursively delete its influential coordinates until there are none left. The total number of coordinates so deleted is referred to as the critical index of l. Let $c_\tau(\boldsymbol{w}_i)$ denote the critical index of $\boldsymbol{w}_i$, and let $C_\tau(\boldsymbol{w}_i)$ denote the set of $c_\tau(\boldsymbol{w}_i)$ largest coordinates of $\boldsymbol{w}_i$. The linear function l is said to be *close* to the i'th dictator function for every i in $C_\tau(\boldsymbol{w}_i)$. A function is *far* from every dictator if it has critical index 0.

An important issue is that the critical index of a linear function can be much larger than the number of influential coordinates and cannot be appropriately bounded. In other words, a linear function can be close to a large number of dictator functions, as per the definition above. To counter this, we employ a structural lemma about halfspaces that was used in the recent work on fooling halfspaces with limited independence [41]. Using this lemma, we are able to prove that if the critical index is large, then one can in fact zero out the coordinates of $\boldsymbol{w}_i$ outside the t largest coordinates for some large enough t, and the agreement of the halfspace h only changes by a negligible amount! Thus, we first carry out the zeroing operation for all linear functions with large critical index.

We now describe the above construction and analysis of the dictatorship test in some more detail. It is convenient to think of the k queries $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ as the rows of a $k \times R$ matrix with $\{0,1\}$ entries. Henceforth, we will refer to matrices $\{0,1\}^{k \times R}$ and their rows

and columns.

We construct two distributions $\mathscr{D}_0, \mathscr{D}_1$ on $\{0,1\}^k$ such that for $s = 0, 1$, we have $\mathbf{Pr}_{x \in \mathscr{D}_s}\left[\vee_{i=1}^k x_i = s\right] \geqslant 1 - \epsilon$ for $\epsilon = o_k(1)$ (this will ensure the completeness of the reduction, i.e., certain disjunctions pass with high probability). Further, the distributions will be carefully chosen to have matching first four moments. This will be used in the soundness analysis where we will use an *invariance principle* to infer structural properties of halfspaces that pass the test with probability noticeably greater than 1/2.

We define the distribution $\tilde{\mathscr{D}}_s^R$ on matrices $\{0,1\}^{k \times R}$ by sampling R columns independently according to $\mathscr{D}_s$, and then perturbing each bit with a small random noise. We define the following test (or equivalently, distribution on examples): given a halfspace h on $\{0,1\}^{k \times R}$, with probability 1/2 we check $h(\boldsymbol{x}) = 0$ for a sample $\boldsymbol{x} \in \tilde{\mathscr{D}}_0^R$, and with probability 1/2 we check $h(\boldsymbol{x}) = 1$ for a sample $\boldsymbol{x} \in \tilde{\mathscr{D}}_1^R$.

Completeness By construction, each of the R disjunctions $\mathsf{OR}_j(\boldsymbol{x}) = \vee_{i=1}^k x_i^{(j)}$ passes the test with probability at least $1 - \epsilon$ (here $x_i^{(j)}$ denotes the entry in the i'th row and j'th column of $\boldsymbol{x}$).

Soundness For the soundness analysis, suppose $h(\boldsymbol{x}) = \mathrm{sgn}(\langle \boldsymbol{w}, \boldsymbol{x} \rangle - \theta)$ is a halfspace that passes the test with probability at least $1/2 + \epsilon$. The halfspace h can be written in two ways by expanding the inner product $\langle \boldsymbol{w}, \boldsymbol{x} \rangle$ along rows and columns, i.e., $h(\boldsymbol{x}) = \mathrm{sgn}(\sum_{i=1}^k \langle \boldsymbol{w}_i, \boldsymbol{x}_i \rangle - \theta) = \mathrm{sgn}(\sum_{i=1}^R \langle \boldsymbol{w}^{(i)}, \boldsymbol{x}^{(i)} \rangle - \theta)$. Let us denote $l_i(\boldsymbol{x}) = \langle \boldsymbol{w}_i, \boldsymbol{x}_i \rangle$.

First, let us see why the linear functions $\langle \boldsymbol{w}_i, \boldsymbol{x}_i \rangle$ must be close to *some* dictator. Note that we need to show that two of the linear functions are close to the *same* dictator.

Suppose each of the linear functions l_i is not *close* to any dictator. In other words, for each i, no single coordinate of the vector $\boldsymbol{w}_i$ is too large (contains more than τ-fraction of the ℓ_2 mass $\|\boldsymbol{w}_i\|_2$ of vector $\boldsymbol{w}_i$). Clearly, this implies that no single column of the matrix $\boldsymbol{w}$ is too *large*.

Recall that the halfspace is given by, $h(\boldsymbol{x}) = \mathrm{sgn}(\sum_{j \in [R]} \langle \boldsymbol{w}^{(j)}, \boldsymbol{x}^{(j)} \rangle - \theta)$. Here $l(\boldsymbol{x}) = \sum_{j \in [R]} \langle \boldsymbol{w}^{(j)}, \boldsymbol{x}^{(j)} \rangle - \theta$ is a degree 1 polynomial into which we are substituting values from two product distributions $\mathscr{D}_0^R$ and $\mathscr{D}_1^R$. Further, the distributions $\mathscr{D}_0$ and $\mathscr{D}_1$ have matching moments up to order 4 by design. Using the invariance principle, the distribution of $l(\boldsymbol{x})$ is roughly the same, whether $\boldsymbol{x}$ is from $\mathscr{D}_0^R$ or $\mathscr{D}_1^R$. Thus, by the invariance principle, the halfspace h is unable to distinguish between the distributions $\mathscr{D}_0^R$ and $\mathscr{D}_1^R$ with a noticeable advantage.

Suppose no two linear functions l_i are *close* to the same dictator, i.e., $C_\tau(\boldsymbol{w}_i) \cap C_\tau(\boldsymbol{w}_j) = \emptyset$. In this case, we condition on the values of $x_i^{(j)}$ for $j \in C_\tau(\boldsymbol{w}_i)$ (note that we condition on at most *one* value in each column so the conditional distribution on each column still has matching first three moments), and then apply the invariance principle using the fact that after deleting the coordinates in $C_\tau(\boldsymbol{w}_i)$, all the remaining coefficients of the weight vector $\boldsymbol{w}$ are small (by definition of critical index). This implies that $C_\tau(\boldsymbol{w}_i) \cap C_\tau(\boldsymbol{w}_j) \neq \emptyset$ for some two rows i, j. This finishes the proof of the soundness claim.

The above consistency-enforcing test almost immediately yields the Unique Games hardness of weak learning disjunctions by halfspaces. To prove NP-hardness, we reduce a version of Label Cover to our problem. This requires a more complicated consistency check, and we have to overcome several additional technical obstacles in the proof.

The main obstacle encountered in transferring the dictatorship test to a Label Cover-based hardness is one that commonly arises for several other problems. Specifically, the

projection constraint on an edge $e = (u, v)$ maps a large set of labels $L = \{\ell_1, \dots, \ell_d\}$ corresponding to a vertex u to a single label ℓ for the vertex v. While composing the Label Cover constraint (u, v) with the dictatorship test, all labels in L have to be necessarily *equivalent*. In several settings including this work, this requires the coordinates corresponding to labels in L to be mostly identical! However, on making the coordinates corresponding to L identical, the prover corresponding to u can determine the identity of edge (u, v), thus completely destroying the soundness of the composition. In fact, the natural extension of the Unique Games-based reduction for MAXCUT [100] to a corresponding Label Cover hardness fails primarily for this reason.

Unlike MAXCUT or other Unique Games-based reductions, in our case, the soundness of the dictatorship test is required to hold against a specific class of functions, i.e, halfspaces. Harnessing this fact, we execute the reduction starting from a Label Cover instance whose projections are *unique on average*. More precisely, a *smooth* Label Cover (introduced in [95]) is one in which for every vertex u, and a pair of labels ℓ, ℓ', the labels $\{\ell, \ell'\}$ project to the same label with a tiny probability over the choice of the edge $e = (u, v)$. Technically, we express the error term in the invariance principle as a certain fourth moment of halfspace, and use the smoothness to bound this error term for most edges of the Label Cover instance. It is of great interest to find other applications where a *weak uniqueness* property like the smoothness condition can be used to convert a Unique Games hardness result to an unconditional NP-hardness result.

9.3   Preliminaries

In this section, we define two important tools in our analysis: i) critical index, ii) invariance principle.

9.3.1   Critical Index

The notion of critical index was first introduced by Servedio [133] and plays an important role in the analysis of halfspaces in [41, 113, 119].

Definition 9.3.1. *Given any real vector $\boldsymbol{w} = (w^{(1)}, w^{(2)}, \dots, w^{(n)}) \in \mathbb{R}^n$. Reorder the coordinates by decreasing absolute value, i.e., $|w^{(i_1)}| \geqslant |w^{(i_2)}| \geqslant \dots \geqslant |w^{(i_n)}|$ and denote $\sigma_t^2 = \sum_{j=t}^n |w^{(i_j)}|^2$. For $0 \leqslant \tau \leqslant 1$. The τ-critical index of the vector $\boldsymbol{w}$ is defined to be the smallest index k such $|w^{(i_k)}| \leqslant \tau \sigma_k$. If no such k exists ($\forall k, |w^{(i_k)}| > \tau \sigma_k$), the τ-critical index is defined to be $+\infty$. The vector $\boldsymbol{w}$ is said to be τ-regular if the τ-critical index is 1.*

A simple observation from [41] is that if the critical index of a sequence is large the sequence must contain a geometrically decreasing subsequence.

Lemma 9.3.2. *(Lemma 5.5 in [41]) Given a vector $\boldsymbol{w} = (w^{(i)})_{i=1}^n$ such that $|w^{(1)}| \geqslant |w^{(2)}| \geqslant \dots \geqslant |w^{(n)}|$, if the τ-critical index of the vector $\boldsymbol{w}$ is larger than l, then for any $1 \leqslant i \leqslant j \leqslant l+1$,*

$$|w^{(j)}| \leqslant \sigma_j \leqslant (\sqrt{1-\tau^2})^{j-i} \sigma_i \leqslant (\sqrt{1-\tau^2})^{j-i} |w^{(i)}|/\tau.$$

In particular, if $j > i + (4/\tau^2)\ln(1/\tau)$ then $|w^{(j)}| \leqslant |w^{(i)}|/3$.

For a τ-regular weight vector, the following lemma bounds the probability that its weighted sum falls into a small interval under certain distributions on the points. The proof is in Appendix 9.8.

Lemma 9.3.3. *Let $\boldsymbol{w} \in \mathbb{R}^n$ be a τ-regular vector $\boldsymbol{w}$, and $\sum |w^{(i)}|^2 = 1$. $\mathscr{D}$ is a distribution over $\{-1,1\}^n$. Define a distribution $\tilde{\mathscr{D}}$ on $\{-1,1\}^n$ as follows: to generate $\boldsymbol{y}$ from $\tilde{\mathscr{D}}$, first sample $\boldsymbol{x}$ from $\mathscr{D}$ and then define,*

$$y^{(i)} = \begin{cases} x^{(i)} & \text{with probability } 1-\gamma \\ \text{random bit} & \text{with probability } \gamma. \end{cases}$$

Then for any interval $[a,b]$, we have

$$\mathbf{Pr}\Big[\langle \boldsymbol{w}, \boldsymbol{y} \rangle \in [a,b]\Big] \leqslant \frac{4|b-a|}{\sqrt{\gamma}} + \frac{4\tau}{\sqrt{\gamma}} + 2e^{-\frac{\gamma^2}{2\tau^2}}.$$

Intuitively, $\langle \boldsymbol{w}, \boldsymbol{y} \rangle$ is τ close to the Gaussian distribution if each $y^{(i)}$ is a random bit and therefore we can bound the probability that $\langle \boldsymbol{w}, \boldsymbol{y} \rangle$ falls into the interval $[a,b]$. In above lemma, each $y^{(i)}$ has probability γ to be a random bit, then γ fraction of $y^{(i)}$ is set to be a random bit and we can therefore bound the probability that $\langle \boldsymbol{w}, \boldsymbol{y} \rangle$ falls into the interval $[a,b]$.

Definition 9.3.4. *For a vector $\boldsymbol{w} \in \mathbb{R}^n$, define set of indices $S_t(\boldsymbol{w}) \subseteq [n]$ as the set of indices containing the t largest coordinates of $\boldsymbol{w}$ by absolute value. Suppose its τ-critical index is c_τ, define set of indices $C_\tau(\boldsymbol{w}) = S_{c_\tau}(\boldsymbol{w})$. In other words, $C_\tau(\boldsymbol{w})$ is the set of indices whose deletion makes the vector w to be τ-regular.*

Definition 9.3.5. *For a vector $\boldsymbol{w} \in \mathbb{R}^n$ and a subset of indices $S \subseteq [n]$, define the vector $\mathsf{Truncate}(\boldsymbol{w}, S) \in \mathbb{R}^n$ as:*

$$(\mathsf{Truncate}(\boldsymbol{w}, S))^{(i)} = \begin{cases} w^{(i)} \text{ if } i \in S \\ 0 \text{ otherwise} \end{cases}$$

As suggested by Lemma 9.3.2, a weight vector with a large critical index has a geometrically decreasing subsequence. The following two lemmas use this fact to bound the probability that the weighted sum of a geometrically decreasing sequence of weights falls into a small interval. First, we restate Claim 5.7 from [41] here.

Lemma 9.3.6. *[Claim 5.7, [41]] Let $|w^{(1)}| \geqslant |w^{(2)}| \ldots \geqslant |w^{(T)}| \geqslant 0$ be a sequence of numbers so that $|w^{(i+1)}| \leqslant |\frac{w^{(i)}}{3}|$ for $1 \leqslant i \leqslant T-1$. Then for any interval $I = [\alpha - \frac{w^{(T)}}{6}, \alpha + \frac{w^{(T)}}{6}]$ of length $\frac{|w^{(T)}|}{3}$, there is at most one point $\boldsymbol{x} \in \{0,1\}^T$ such that $\langle \boldsymbol{w}, \boldsymbol{x} \rangle \in I$.*

Lemma 9.3.7. *Let $|w^{(1)}| \geqslant |w^{(2)}| \ldots \geqslant |w^{(T)}| \geqslant 0$ be a sequence of numbers so that $|w^{(i+1)}| \leqslant |\frac{w^{(i)}}{3}|$ for $1 \leqslant i \leqslant T-1$. $\mathscr{D}$ is a distribution over $\{-1,1\}^T$. Define a distribution $\tilde{\mathscr{D}}$ on $\{-1,1\}^T$ as follows: To generate $\boldsymbol{y}$ from $\tilde{\mathscr{D}}$, sample $\boldsymbol{x}$ from $\mathscr{D}$ and set*

$$y^{(i)} = \begin{cases} x^{(i)} & \text{with probability } 1-\gamma \\ \text{random bit} & \text{with probability } \gamma. \end{cases}$$

Then for any $\theta \in \mathbb{R}$ we have

$$\mathbf{Pr}\Big[\langle \boldsymbol{w}, \boldsymbol{y} \rangle \in [\theta - \frac{w^{(T)}}{6}, \theta + \frac{w^{(T)}}{6}]\Big] \leqslant \Big(1 - \frac{\gamma}{2}\Big)^T.$$

Proof. By Lemma 9.3.6, we know that for the interval $J = \left[\theta - \frac{|w^T|}{6}, \theta + \frac{|w^T|}{6}\right]$, there is at most one point $\boldsymbol{r} \in \{-1,1\}^T$ such that $\langle \boldsymbol{w}, \boldsymbol{r} \rangle \in J$. If no such $\boldsymbol{r}$ exists then clearly the probability is zero. On the other hand, suppose there exists such an $\boldsymbol{r}$, then $\langle \boldsymbol{w}, \boldsymbol{y} \rangle \in J$ only if $(y_1^{(1)}, y_1^{(2)}, \ldots, y_1^{(T)}) = (r^{(1)}, \ldots, r^{(T)})$ holds.

Conditioned on any fixing of the bits $\boldsymbol{x}$, every bit $y^{(j)}$ is an independent random bit with probability γ. Therefore, for every fixing of $\boldsymbol{x}$, for each $i \in [T]$, with probability at least $\gamma/2$, $y^{(i)}$ is not equal to $r^{(i)}$. Therefore, $\mathbf{Pr}[y^{(1)} = r^{(1)}, y^{(2)} = r^{(2)}, \ldots, y^{(T)} = r^{(T)}] \leqslant \left(1 - \frac{\gamma}{2}\right)^T$. $\qquad \square$

9.3.2 Invariance Principle

While invariance principles have been shown in various settings by [31, 114, 117], we re-state a version of the principle well suited for our application. We present a self-contained proof for it in Appendix 9.9.

Definition 9.3.8. *A C^4-function $\Psi(x): \mathbb{R} \to \mathbb{R}$ is said to be B-nice if $|\Psi''''(t)| \leqslant B$ for all $t \in \mathbb{R}$.*

Definition 9.3.9. *Two ensembles of random variables $\mathscr{P} = (p_1, \ldots, p_k)$ and $\mathscr{Q} = (q_1, \ldots, q_k)$ are said to have matching moments up to degree d if for every multi-set S of elements from $[k]$, $|S| \leqslant d$, we have $\mathbf{E}[\prod_{i \in S} p_i] = \mathbf{E}[\prod_{i \in S} q_i]$.*

Theorem 9.3.10. *(Invariance Principle) Let $\mathscr{A} = \{\boldsymbol{A}^{\{1\}}, \ldots, \boldsymbol{A}^{\{R\}}\}, \mathscr{B} = \{\boldsymbol{B}^{\{1\}}, \ldots, \boldsymbol{B}^{\{R\}}\}$ be families of ensembles of random variables with $\boldsymbol{A}^{\{i\}} = \{a_1^{(i)}, \ldots, a_{k_i}^{(i)}\}$ and $\boldsymbol{B}^{\{i\}} = \{b_1^{(i)}, \ldots, b_{k_i}^{(i)}\}$, satisfying the following properties:*

- *For each $i \in [R]$, the random variables in ensembles $(\boldsymbol{A}^{\{i\}}, \boldsymbol{B}^{\{i\}})$ have matching moments up to degree 3. Further all the random variables in $\mathscr{A}$ and $\mathscr{B}$ are bounded by 1.*

- *The ensembles $\boldsymbol{A}^{\{i\}}$ are all independent of each other, similarly the ensembles $\boldsymbol{B}^{\{i\}}$ are independent of each other.*

Given a set of vectors $\boldsymbol{l} = \{\boldsymbol{l}^{\{1\}}, \ldots, \boldsymbol{l}^{\{R\}}\}(\boldsymbol{l}^{\{i\}} \in \mathbb{R}^{k_i})$, define the linear function $\boldsymbol{l}: \mathbb{R}^{k_1} \times \cdots \times \mathbb{R}^{k_R} \to \mathbb{R}$ as

$$\boldsymbol{l}(\boldsymbol{x}) = \sum_{i \in [R]} \langle \boldsymbol{l}^{\{i\}}, \boldsymbol{x}^{\{i\}} \rangle$$

Then for a B-nice function $\Psi: \mathbb{R} \to \mathbb{R}$ we have

$$\left| \mathbf{E}_{\mathscr{A}}\left[\Psi\left(\boldsymbol{l}(\mathscr{A}) - \theta\right)\right] - \mathbf{E}_{\mathscr{B}}\left[\Psi\left(\boldsymbol{l}(\mathscr{B}) - \theta\right)\right] \right| \leqslant B \sum_{i \in [R]} \|\boldsymbol{l}^{\{i\}}\|_1^4$$

for all $\theta > 0$. Further, define the spread function $c(\alpha)$ corresponding to the ensembles $\mathscr{A}, \mathscr{B}$ and the linear function $\boldsymbol{l}$ as follows,

(***Spread Function:***)*For $1/2 > \alpha > 0$, let*

$$c(\alpha) = \max\left(\sup_\theta \mathbf{Pr}_{\mathscr{A}}\left[\boldsymbol{l}(\mathscr{A}) \in [\theta - \alpha, \theta + \alpha]\right], \quad \sup_\theta \mathbf{Pr}_{\mathscr{B}}\left[\boldsymbol{l}(\mathscr{B}) \in [\theta - \alpha, \theta + \alpha]\right]\right)$$

then for all θ,

$$\left| \mathbf{E}_{\mathscr{A}}[\operatorname{sgn}(\boldsymbol{l}(\mathscr{A}) - \theta)] - \mathbf{E}_{\mathscr{B}}[\operatorname{sgn}(\boldsymbol{l}(\mathscr{B}) - \theta)] \right| \leqslant O\left(\frac{1}{\alpha^4}\right) \sum_{i \in [R]} \|\boldsymbol{l}^{\{i\}}\|_1^4 + 2c(\alpha).$$

178

9.4 Construction of the Dictatorship Test

In this section we describe the construction of the dictatorship test which will be the key ingredient in the hardness reduction from k-UNIQUE LABEL-COVER.

9.4.1 Distributions $\mathscr{D}_0$ and $\mathscr{D}_1$

The dictatorship test is based on following two distributions $\mathscr{D}_0$ and $\mathscr{D}_1$ defined on $\{-1,1\}^k$.

Lemma 9.4.1. *For $k \in \mathbb{N}$, there exists two probability distributions $\mathscr{D}_0$, $\mathscr{D}_1$ on $\{-1,1\}^k$ such that $\mathbf{Pr}_{x \sim \mathscr{D}_0}\{every\ x_i\ is\ 0\} \geqslant 1 - \frac{2}{\sqrt{k}}$, $\mathbf{Pr}_{x \sim \mathscr{D}_1}\{every\ x_i\ is\ 0\} \leqslant \frac{1}{\sqrt{k}}$, while matching moments up to degree 4, i.e., $\forall a,b,c,d \in [k]$*

$$\mathop{\mathbf{E}}_{\mathscr{D}_0}[x_a] = \mathop{\mathbf{E}}_{\mathscr{D}_1}[x_a] \qquad\qquad \mathop{\mathbf{E}}_{\mathscr{D}_0}[x_a x_b x_c x_d] = \mathop{\mathbf{E}}_{\mathscr{D}_1}[x_a x_b x_c x_d]$$

$$\mathop{\mathbf{E}}_{\mathscr{D}_0}[x_a x_b] = \mathop{\mathbf{E}}_{\mathscr{D}_1}[x_a x_b] \qquad\qquad \mathop{\mathbf{E}}_{\mathscr{D}_0}[x_a x_b x_c] = \mathop{\mathbf{E}}_{\mathscr{D}_1}[x_a x_b x_c]$$

Proof. For $\epsilon = \frac{1}{\sqrt{k}}$, take $\mathscr{D}_1$ to be the following distribution:

1. with probability $(1-\epsilon)$, randomly set exactly one of the bit to be 1 and all the other to be 0;

2. with probability $\frac{\epsilon}{4}$, independently set every bit to be 1 with probability $\frac{1}{k^{1/3}}$;

3. with probability $\frac{\epsilon}{4}$, independently set every bit to be 1 with probability $\frac{2}{k^{1/3}}$;

4. with probability $\frac{\epsilon}{4}$, independently set every bit to be 1 with probability $\frac{3}{k^{1/3}}$;

5. with probability $\frac{\epsilon}{4}$, independently set every bit to be 1 with probability $\frac{4}{k^{1/3}}$.

The distribution $\mathscr{D}_0$ is defined to be the following distribution with parameter $\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4$ to be specified later:

1. with probability $1 - (\epsilon_1 + \epsilon_2 + \epsilon_3 + \epsilon_4)$, set every bit to be zero;

2. with probability ϵ_1, independently set every bit to be 1 with probability $\frac{1}{k^{1/3}}$;

3. with probability ϵ_2, independently set every bit to be 1 with probability $\frac{2}{k^{1/3}}$;

4. with probability ϵ_3, independently set every bit to be 1 with probability $\frac{3}{k^{1/3}}$;

5. with probability ϵ_4, independently set every bit to be 1 with probability $\frac{4}{k^{1/3}}$.

From the definition of $\mathscr{D}_0, \mathscr{D}_1$, we know that $\mathbf{Pr}_{x \sim \mathscr{D}_0}[every\ x_i\ is\ 0] \geqslant 1 - (\epsilon_1 + \epsilon_2 + \epsilon_3 + \epsilon_4)$ and $\mathbf{Pr}_{x \sim \mathscr{D}_1}[every\ x_i\ is\ 0] \leqslant \epsilon = \frac{1}{\sqrt{k}}$.

It remains to determine each ϵ_i. Notice that the moment matching conditions can be

expressed as a linear system over the parameters $\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4$ as follows:

$$\sum_{i=1}^{4} \epsilon_i \left(\frac{i}{k^{1/3}}\right) = (1-\epsilon)/k + \sum_{i=1}^{4} \frac{\epsilon}{4}\left(\frac{i}{k^{\frac{1}{3}}}\right)$$

$$\sum_{i=1}^{4} \epsilon_i \left(\frac{i}{k^{1/3}}\right)^2 = \sum_{i=1}^{4} \frac{\epsilon}{4}\left(\frac{i}{k^{\frac{1}{3}}}\right)^2$$

$$\sum_{i=1}^{4} \epsilon_i \left(\frac{i}{k^{\frac{1}{3}}}\right)^3 = \sum_{i=1}^{4} \frac{\epsilon}{4}\left(\frac{i}{k^{\frac{1}{3}}}\right)^3$$

$$\sum_{i=1}^{4} \epsilon_i \left(\frac{i}{k^{\frac{1}{3}}}\right)^4 = \sum_{i=1}^{4} \frac{\epsilon}{4}\left(\frac{i}{k^{\frac{1}{3}}}\right)^4.$$

We then show that such a linear system has a feasible solution $\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4 > 0$ and $\sum_{i=1}^{4} \epsilon_i \leqslant 2/\sqrt{k}$.

To prove this, by applying Cramer's rule,

$$\epsilon_1 = \frac{\begin{vmatrix} (1-\epsilon)/k + \sum_{i=1}^{4}\frac{\epsilon}{4}(\frac{i}{k^{\frac{1}{3}}}) & \frac{2}{k^{\frac{1}{3}}} & \frac{3}{k^{\frac{1}{3}}} & \frac{4}{k^{\frac{1}{3}}} \\ \sum_{i=1}^{4}\frac{\epsilon}{4}(\frac{i}{k^{\frac{1}{3}}})^2 & \frac{4}{k^{\frac{2}{3}}} & \frac{9}{k^{\frac{2}{3}}} & \frac{16}{k^{\frac{2}{3}}} \\ \sum_{i=1}^{4}\frac{\epsilon}{4}(\frac{i}{k^{\frac{1}{3}}})^3 & \frac{8}{k^{\frac{3}{3}}} & \frac{27}{k^{\frac{3}{3}}} & \frac{64}{k^{\frac{3}{3}}} \\ \sum_{i=1}^{4}\frac{\epsilon}{4}(\frac{i}{k^{\frac{1}{3}}})^4 & \frac{16}{k^{\frac{4}{3}}} & \frac{81}{k^{\frac{4}{3}}} & \frac{256}{k^{\frac{4}{3}}} \end{vmatrix}}{\begin{vmatrix} \frac{1}{k^{\frac{1}{3}}} & \frac{2}{k^{\frac{1}{3}}} & \frac{3}{k^{\frac{1}{3}}} & \frac{4}{k^{\frac{1}{3}}} \\ \frac{1}{k^{\frac{2}{3}}} & \frac{4}{k^{\frac{2}{3}}} & \frac{9}{k^{\frac{2}{3}}} & \frac{16}{k^{\frac{2}{3}}} \\ \frac{1}{k^{\frac{3}{3}}} & \frac{8}{k^{\frac{3}{3}}} & \frac{27}{k^{\frac{3}{3}}} & \frac{64}{k^{\frac{3}{3}}} \\ \frac{1}{k^{\frac{4}{3}}} & \frac{16}{k^{\frac{4}{3}}} & \frac{81}{k^{\frac{4}{3}}} & \frac{256}{k^{\frac{4}{3}}} \end{vmatrix}}$$

With some calculation using basic linear algebra, we get

$$\epsilon_1 = \epsilon/4 + \frac{\begin{vmatrix} (1-\epsilon)/k & \frac{2}{k^{\frac{1}{3}}} & \frac{3}{k^{\frac{1}{3}}} & \frac{4}{k^{\frac{1}{3}}} \\ 0 & \frac{4}{k^{\frac{2}{3}}} & \frac{9}{k^{\frac{2}{3}}} & \frac{16}{k^{\frac{2}{3}}} \\ 0 & \frac{8}{k^{\frac{3}{3}}} & \frac{3}{k^{\frac{3}{3}}} & \frac{64}{k^{\frac{3}{3}}} \\ 0 & \frac{16}{k^{\frac{4}{3}}} & \frac{3}{k^{\frac{4}{3}}} & \frac{256}{k^{\frac{4}{3}}} \end{vmatrix}}{\begin{vmatrix} \frac{1}{k^{\frac{1}{3}}} & \frac{2}{k^{\frac{1}{3}}} & \frac{3}{k^{\frac{1}{3}}} & \frac{4}{k^{\frac{1}{3}}} \\ \frac{1}{k^{\frac{2}{3}}} & \frac{4}{k^{\frac{2}{3}}} & \frac{9}{k^{\frac{2}{3}}} & \frac{16}{k^{\frac{2}{3}}} \\ \frac{1}{k^{\frac{3}{3}}} & \frac{8}{k^{\frac{3}{3}}} & \frac{27}{k^{\frac{3}{3}}} & \frac{64}{k^{\frac{3}{3}}} \\ \frac{1}{k^{\frac{4}{3}}} & \frac{16}{k^{\frac{4}{3}}} & \frac{81}{k^{\frac{4}{3}}} & \frac{256}{k^{\frac{4}{3}}} \end{vmatrix}} = \frac{1}{4\sqrt{k}} + O\left(\frac{1}{k^{\frac{2}{3}}}\right).$$

For big enough k, we have $0 \leqslant \epsilon_1 \leqslant \frac{1}{2\sqrt{k}}$. By similar calculation, we can bound $\epsilon_2, \epsilon_3, \epsilon_4$ by $\frac{1}{2\sqrt{k}}$. Overall, we have $\epsilon_1 + \epsilon_2 + \epsilon_3 + \epsilon_4 \leqslant 2/\sqrt{k}$

□

We define a "noisy" version of $\mathscr{D}_b$ ($b \in \{0,1\}$) below.

Definition 9.4.2. *For $b \in \{-1,1\}$, define the distribution $\tilde{D}_b$ on $\{-1,1\}^k$ as follows:*

- *First generate $x \in \{-1,1\}^k$ according to D_b.*

- *For each $i \in [k]$,*

$$y_i = \begin{cases} x_i & \text{with probability } 1 - \frac{1}{k^2} \\ \text{uniform random bit } u_i & \text{with probability } \frac{1}{k^2} \end{cases}$$

Observation 9.4.3. *Since the noise is defined to be an independent uniform random bit, when calculating moments of y, such as $\mathbf{E}_{\tilde{D}_b}[y_{i_1} y_{i_2} \cdots y_{i_d}]$, we can substitute y_i by $(1-\gamma)x_i + \frac{1}{2}\gamma$. Therefore, a degree d moment of y can be expressed as a weighted sum of moments of x of degree up to d. Since $\mathscr{D}_0$ and $\mathscr{D}_1$ have matching moments up to degree 4, $\tilde{\mathscr{D}}_0$ and $\tilde{\mathscr{D}}_1$ also have matching moments up to degree 4.*

The following simple lemma asserts that conditioning the two distributions $\tilde{D}_0$ and $\tilde{D}_1$ on the same coordinate x_j being fixed to value b results in conditional distributions that still have matching moments up to degree 3.

Lemma 9.4.4. *Given two distributions $\mathscr{P}_0, \mathscr{P}_1$ on $\{-1,1\}^k$ with matching moments up to degree d, for any multi-set S of elements from $[k]$, $|S| \leqslant d-1$, $j \in [k]$ and $c \in \{-1,1\}$.*

$$\mathbf{E}_{\mathscr{P}_0}[\prod_{i \in S} x_i \mid x_j = c] = \mathbf{E}_{\mathscr{P}_1}[\prod_{i \in S} x_i \mid x_j = c].$$

Proof. For the case $c = 1$ and any $b \in \{-1,1\}$,

$$\mathbf{E}_{\mathscr{P}_b}[x_j \prod_{i \in S} x_i] = \mathbf{E}_{\mathscr{P}_b}[\prod_{i \in S} x_i \mid x_j = 1]\mathbf{Pr}_{\mathscr{P}_0}[x_j = 1] = \mathbf{E}_{\mathscr{P}_b}[\prod_{i \in S} x_i \mid x_j = 1] \mathbf{E}_{\mathscr{P}_0}[x_j].$$

Therefore,

$$\mathbf{E}_{\mathscr{P}_0}[\prod_{i \in S} x_i \mid x_j = 1] = \frac{\mathbf{E}_{\mathscr{P}_0}[x_j \prod_{i \in S} x_i]}{\mathbf{E}_{\mathscr{P}_0}[x_j]} = \frac{\mathbf{E}_{\mathscr{P}_1}[x_j \prod_{i \in S} x_i]}{\mathbf{E}_{\mathscr{P}_1}[x_j]} = \mathbf{E}_{\mathscr{P}_1}[\prod_{i \in S} x_i \mid x_j = 1].$$

For the case $c = 0$, replace x_j with $x_j' = 1 - x_j$. It is easy to see that $\mathscr{P}_0$ $\mathscr{P}_1$ still have matching moments and conditioning on $x_j = 0$ is the same as conditioning on $x_j' = 1$. Hence we can reduce to the case $c = 1$. $\qquad \square$

9.4.2 The Dictatorship Test

Let R be a positive integer. Based on the distribution $\mathscr{D}_0$ and $\mathscr{D}_1$, we define the dictatorship test as follows:

1. Generate a random bit $b \in \{0,1\}$.
2. Generate $\boldsymbol{x} \in \{-1,1\}^{kR}$ from $\mathscr{D}_b^R$.
3. For each $i \in [k], j \in [R]$,

$$y_i^{(j)} = \begin{cases} x_i^{(j)} & \text{with probability } 1 - \frac{1}{k^2}; \\ \text{random bit} & \text{with probability } \frac{1}{k^2}. \end{cases}$$

4. Output the pair $(\boldsymbol{y}, b)$. Equivalently, ACCEPT if $h(\boldsymbol{y}) = b$.

We can also view y as being generated as follows: i) With probability $\frac{1}{2}$, generate a negative sample from distribution $\tilde{\mathscr{D}}_0^R$; ii) With probability $\frac{1}{2}$, generate a positive sample from distribution $\tilde{\mathscr{D}}_1^R$.

Theorem 9.4.5. *(Completeness) For any $j \in [R]$, $h(\boldsymbol{y}) = \vee_{i=1}^k y_i^{(j)}$ passes with probability $\geqslant 1 - \frac{3}{\sqrt{k}}$.*

Proof. If x is generated from $\mathscr{D}_0^R$, we know that with probability at least $1 - \frac{2}{\sqrt{k}}$, all the bits in $\{x_1^{(j)}, x_2^{(j)}, \ldots, x_k^{(j)}\}$ are set to 0. By union bound, with probability at least $1 - \frac{2}{\sqrt{k}} - \frac{1}{k}$, $\{y_1^{(j)}, y_2^{(j)}, \ldots, y_k^{(j)}\}$ are all set to 0, in which case the test passes as $\vee_{i=1}^k y_i^{(j)} = 0$. If x is generated from $\mathscr{D}_1^R$, we know that with probability at least $1 - \frac{1}{\sqrt{k}}$, one of the bits in $\{x_1^{(j)}, x_2^{(j)}, \ldots, x_k^{(j)}\}$ is set to 1 and by union bound one of $\{y_1^{(j)}, y_2^{(j)}, \ldots, y_k^{(j)}\}$ is set to 1 with probability at least $1 - \frac{1}{\sqrt{k}} - \frac{1}{k}$, in which case the test passes since $\vee_{i=1}^k y_i^{(j)} = 1$. Overall, the test passes with probability at least $1 - \frac{3}{\sqrt{k}}$. $\qquad\square$

9.4.3 Soundness Analysis

The soundness property of the test (formally stated in Theorem 9.4.8) is that if some $h(\boldsymbol{y})$ passes the above dictatorship test with high probability, then we can decode each $\boldsymbol{w}_i$ ($i \in [k]$) in to a small list and at least two of the list will intersect. The proof of the soundness property is based on two key lemmas (Lemma 9.4.6, 9.4.7). Roughly speaking, the first lemma states that if a halfspace passes the test with good probability, then two of its critical index sets $C_\tau(\boldsymbol{w}_i), C_\tau(\boldsymbol{w}_j)$ (see Definition 9.3.1) must intersect; the second lemma states that every halfspace can be approximated by another halfspace with a small critical index.

Let $h(\boldsymbol{y})$ be a halfspace function on $\{-1, 1\}^{kR}$ given by $h(\boldsymbol{y}) = \text{sgn}(\langle \boldsymbol{w}, \boldsymbol{y} \rangle - \theta)$. Equivalently, $h(\boldsymbol{y})$ can be written as

$$h(\boldsymbol{y}) = \text{sgn}\left(\sum_{j \in [R]} \langle \boldsymbol{w}^{(j)}, \boldsymbol{y}^{(j)} \rangle - \theta \right) = \text{sgn}\left(\sum_{i \in [k]} \langle \boldsymbol{w}_i, \boldsymbol{y}_i \rangle - \theta \right)$$

where $\boldsymbol{w}^{(j)} \in \mathbb{R}^k$ and $\boldsymbol{w}_i \in \mathbb{R}^R$.

Lemma 9.4.6. *(Common Influential Coordinate) For $\tau = \frac{1}{k^7}$, let $h(\boldsymbol{y})$ be a halfspace such that for all $i \neq j \in [k]$, we have $C_\tau(\boldsymbol{w}_i) \cap C_\tau(\boldsymbol{w}_j) = \emptyset$. Then*

$$\left| \mathop{\mathbf{E}}_{\tilde{\mathscr{D}}_0^R}[h(\boldsymbol{y})] - \mathop{\mathbf{E}}_{\tilde{\mathscr{D}}_1^R}[h(\boldsymbol{y})] \right| \leqslant O\left(\frac{1}{k}\right)$$

Proof. Fix the following notation,

$$\boldsymbol{s}_i = \text{Truncate}(\boldsymbol{w}_i, C_\tau(\boldsymbol{w}_i)) \qquad\qquad \boldsymbol{l}_i = \boldsymbol{w}_i - \boldsymbol{s}_i$$
$$\boldsymbol{y}_i^L = \text{Truncate}(\boldsymbol{y}_i, C_\tau(\boldsymbol{w}_i)) \qquad\qquad \boldsymbol{y}^L = \boldsymbol{y}_1^L, \boldsymbol{y}_2^L, \ldots, \boldsymbol{y}_k^L$$

We can rewrite the halfspace $h(\boldsymbol{y})$ as $h(\boldsymbol{y}) = \text{sgn}\left(\langle \boldsymbol{s}, \boldsymbol{y}^L \rangle + \langle \boldsymbol{l}, \boldsymbol{y} \rangle - \theta\right)$. Let us first normalize the halfspace $h(\boldsymbol{y})$ so that $\sum_{i \in [k]} \|\boldsymbol{l}_i\|^2 = 1$. We now condition on a possible fixing of the

vector $\boldsymbol{y}^L$. Under this conditioning and for $\boldsymbol{y}$ chosen randomly from the distribution $\tilde{\mathcal{D}}_0^R$, define the family of ensembles $\mathcal{A} = \boldsymbol{A}^{\{1\}}, \ldots, \boldsymbol{A}^{\{R\}}$ as follows:

$$\boldsymbol{A}^{\{j\}} = \{y_i^{(j)} \mid \forall i \in [k] \text{ such that } j \notin C_\tau(\boldsymbol{w}_i)\}$$

Similarly define the ensemble $\mathcal{B} = \boldsymbol{B}^{\{1\}}, \ldots, \boldsymbol{B}^{\{R\}}$ using $\boldsymbol{y}$ chosen randomly from the distribution $\tilde{\mathcal{D}}_1^R$. Further let us denote $\boldsymbol{l}^{\{j\}} = (l_1^{(j)}, \ldots, l_k^{(j)})$. Now we apply the invariance principle (Theorem 9.3.10) to the ensembles $\mathcal{A}, \mathcal{B}$ and the linear function $\boldsymbol{l}$. For each $j \in [R]$, there is at most one coordinate $i \in [k]$ such that $j \in C_\tau(\boldsymbol{w}_i)$. Thus, conditioning on $\boldsymbol{y}^L$ amounts to fixing of at most one variable $y_i^{(j)}$ in each $\{y_i^{(j)}\}_{i \in [k]}$. By Lemma 9.4.4, since $\tilde{\mathcal{D}}_0$ and $\tilde{\mathcal{D}}_1$ have matching moments up to degree 4, we get that $\boldsymbol{A}^{\{j\}}$ and $\boldsymbol{B}^{\{j\}}$ have matching moments up to degree 3. Also notice that $\max_{j \in [R], i \in [k]} |l_i^{(j)}| \leqslant \tau \|\boldsymbol{l}_i\|_2 \leqslant \tau \|\boldsymbol{l}\|_2$ (as $\boldsymbol{l}_i$ is a τ-regular) and each $y_i^{(j)}$ is set to be a random unbiased bit with probability $\frac{1}{k^2}$; by Lemma 9.3.3, the linear function $\boldsymbol{l}$ and the ensembles $\mathcal{A}, \mathcal{B}$ satisfy the following spread property for every $\theta' \in \mathbb{R}$:

$$\mathbf{Pr}_{\mathcal{A}}\Big[\boldsymbol{l}(\mathcal{A}) \in [\theta' - \alpha, \theta' + \alpha]\Big] \leqslant c(\alpha)$$

$$\mathbf{Pr}_{\mathcal{B}}\Big[\boldsymbol{l}(\mathcal{B}) \in [\theta' - \alpha, \theta' + \alpha]\Big] \leqslant c(\alpha),$$

where $c(\alpha) \leqslant 8\alpha k + 4\tau k + 2e^{-\frac{1}{2\tau^2 k^4}}$ (by setting $\gamma = \frac{1}{k^2}$ and $|b - a| = 2\alpha$ in Lemma 9.3.3). Using the invariance principle (Theorem 9.3.10) this implies:

$$\Big| \mathbf{E}_{\mathcal{A}}\Big[\mathrm{sgn}\Big(\langle \boldsymbol{s}, \boldsymbol{y}^L \rangle + \sum_{j \in [R]} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{A}^{\{j\}} \rangle - \theta \Big) \Big| \boldsymbol{y}^L \Big] -$$

$$\mathbf{E}_{\mathcal{B}}\Big[\mathrm{sgn}\Big(\langle \boldsymbol{s}, \boldsymbol{y}^L \rangle + \sum_{j \in [R]} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{B}^{\{j\}} \rangle - \theta \Big) \Big| \boldsymbol{y}^L \Big] \Big|$$

$$\leqslant O\Big(\frac{1}{\alpha^4}\Big) \sum_{i \in [R]} \|\boldsymbol{l}^{\{i\}}\|_1^4 + 2c(\alpha) \quad (9.1)$$

By definition of the critical index, we have $\max_{j \in [R]} l_i^j \leqslant \tau \|\boldsymbol{l}_i\|_2$. Using this, we can bound $\sum_{i \in [R]} \|\boldsymbol{l}^{\{i\}}\|_1^4$ as follows:

$$\sum_{j \in [R]} \|\boldsymbol{l}^{\{j\}}\|_1^4 \leqslant k^4 \sum_{i \in [k]} \sum_{j \in [R]} \|l_i^{(j)}\|^4 \leqslant k^4 \sum_{i \in [k]} \Big(\max_{j \in [R]} |l_i^{(j)}|^2 \Big) \|\boldsymbol{l}_i\|_2^2$$

$$\leqslant k^4 \tau^2 \sum_{i \in [k]} \|\boldsymbol{l}_i\|_2^2 \leqslant k^4 \tau^2 \|\boldsymbol{l}\|_2^2 \leqslant \frac{1}{k^8}.$$

In the final step, we used the fact that $\tau = \frac{1}{k^7}$ and $\|\boldsymbol{l}\|_2 = 1$ by normalization. Let us fix $\alpha = \frac{1}{k^2}$. The inequality (9.1) holds for all settings of $\boldsymbol{y}^L$. Averaging over all settings of $\boldsymbol{y}^L$ we get that (9.1) can be bounded by $O(\frac{1}{k})$. $\qquad \square$

The set $C_\tau(\boldsymbol{w}_i)$ can be thought of as the set of *influential* coordinates of $\boldsymbol{w}_i$. In this light, the above lemma asserts that unless some two vectors $\boldsymbol{w}_i, \boldsymbol{w}_j$ have a *common influential coordinate*, the halfspace $h(\boldsymbol{y})$ cannot distinguish between $\tilde{\mathcal{D}}_0^R$ and $\tilde{\mathcal{D}}_1^R$.

Unlike with the traditional notion of influence, it is unclear whether the number of coordinates in $C_\tau(\boldsymbol{w}_i)$ is small. The following lemma yields a way to get around this.

Lemma 9.4.7. *(Bounding the number of influential coordinates) Fix*

$$t = \frac{4}{\tau^2}(\log(1/\tau) + \log R) + 4k^2 \log(1/k)\frac{4}{\tau^2}(\log(1/\tau)).$$

Given a halfspace $h(\boldsymbol{y})$ and $\ell \in [k]$ such that $|C_\tau(\boldsymbol{w}_\ell)| > t$, define $\tilde{h}(\boldsymbol{y}) = \text{sgn}(\sum_{i \in [k]}\langle \tilde{\boldsymbol{w}}_i, \boldsymbol{y}_i\rangle - \theta)$ as follows: $\tilde{\boldsymbol{w}}_\ell = \text{Truncate}(\boldsymbol{w}_\ell, S_t(\boldsymbol{w}_\ell))$ and $\tilde{\boldsymbol{w}}_i = \boldsymbol{w}_i$ for all $i \neq \ell$. Then,

$$\left| \mathop{\mathbf{E}}_{\tilde{\mathcal{D}}_0^R}[\tilde{h}(\boldsymbol{y})] - \mathop{\mathbf{E}}_{\tilde{\mathcal{D}}_0^R}[h(\boldsymbol{y})]\right| \leqslant \frac{1}{k^2} \; and \; \left| \mathop{\mathbf{E}}_{\tilde{\mathcal{D}}_1^R}[\tilde{h}(\boldsymbol{y})] - \mathop{\mathbf{E}}_{\tilde{\mathcal{D}}_1^R}[h(\boldsymbol{y})]\right| \leqslant \frac{1}{k^2}.$$

Proof. Without loss of generality, we assume $\ell = 1$ and $|w_1^{(1)}| \geqslant |w_1^{(2)}| \geqslant \cdots \geqslant |w_1^{(R)}|$. In particular, this implies $S_t(\boldsymbol{w}_1) = \{1,\ldots,t\}$. Set $T = 4k^2\log(1/k)$. Define the subset G of $S_t(\boldsymbol{w}_1)$ as

$$G = \{g_i \mid g_i = 1 + i\lceil(4/\tau^2)\ln(1/\tau)\rceil, 0 \leqslant i \leqslant T\}.$$

Therefore, by Lemma 9.3.2, $|w_1^{(g_i)}|$ is a geometrically decreasing sequence such that $|w_1^{(g_{i+1})}| \leqslant |w_1^{(g_i)}|/3$. Let $H = S_t(\boldsymbol{w}_1) \setminus G$. Fix the following notation:

$$\boldsymbol{w}_1^G = \text{Truncate}(\boldsymbol{w}_1, G), \quad \boldsymbol{w}_1^H = \text{Truncate}(\boldsymbol{w}_1, H), \quad \boldsymbol{w}_1^{>t} = \text{Truncate}(\boldsymbol{w}_1, \{t+1,\ldots,n\}).$$

Similarly, define the vectors $\boldsymbol{y}_1^G, \boldsymbol{y}_1^H, \boldsymbol{y}_1^{>t}$. We now rewrite the halfspace functions $h(\boldsymbol{y})$ and $\tilde{h}(\boldsymbol{y})$ as:

$$h(\boldsymbol{y}) = \text{sgn}\Big(\sum_{i=2}^k \langle \boldsymbol{w}_i, \boldsymbol{y}_i\rangle + \langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G\rangle + \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H\rangle + \langle \boldsymbol{w}_1^{>t}, \boldsymbol{y}_1^{>t}\rangle - \theta\Big)$$

$$\tilde{h}(\boldsymbol{y}) = \text{sgn}\Big(\sum_{i=2}^k \langle \boldsymbol{w}_i, \boldsymbol{y}_i\rangle + \langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G\rangle + \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H\rangle - \theta\Big).$$

Notice that for any $\boldsymbol{y}$, $h(\boldsymbol{y}) \neq \tilde{h}(\boldsymbol{y})$ implies

$$\Big|\sum_{i=2}^k \langle \boldsymbol{w}_i, \boldsymbol{y}_i\rangle + \langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G\rangle + \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H\rangle - \theta\Big| \leqslant |\langle \boldsymbol{w}_1^{>t}, \boldsymbol{y}_1^{>t}\rangle|. \tag{9.2}$$

By Lemma 9.3.2, we know that

$$|w_1^{(g_T)}|^2 \quad \geqslant \quad \frac{\tau^2}{(1-\tau^2)^{t-g_T}}\|\boldsymbol{w}_1^{>t}\|_2^2 \quad \geqslant \quad \frac{\tau^2}{(1-\tau^2)^{\frac{4}{\tau^2}(\log(1/\tau)+\log R)}}\|\boldsymbol{w}_1^{>t}\|_2^2 \quad \geqslant \quad \frac{R^2}{\tau}\|\boldsymbol{w}_1^{>t}\|_2^2.$$

Using the fact that $R\|\boldsymbol{w}_1^{>t}\|_2^2 \geqslant \|\boldsymbol{w}_1^{>t}\|_1^2$, we can get that $\|\boldsymbol{w}_1^{>t}\|_1 \leqslant \sqrt{\tau}|w_1^{(g_T)}| \leqslant \frac{1}{6}|w_1^{(g_T)}|$. Combining the above inequality with (9.2) we see that,

$$\mathop{\mathbf{Pr}}_{\tilde{\mathcal{D}}_0^R}\Big[h(\boldsymbol{y}) \neq \tilde{h}(\boldsymbol{y})\Big] \leqslant \mathop{\mathbf{Pr}}_{\tilde{\mathcal{D}}_0^R}\Big[|\sum_{i=2}^k \langle \boldsymbol{w}_i, \boldsymbol{y}_i\rangle + \langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G\rangle + \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H\rangle - \theta| \leqslant |\langle \boldsymbol{w}_1^{>t}, \boldsymbol{y}_1^{>t}\rangle|\Big]$$

$$\leqslant \mathop{\mathbf{Pr}}_{\tilde{\mathcal{D}}_0^R}\Big[|\sum_{i=2}^k \langle \boldsymbol{w}_i, \boldsymbol{y}_i\rangle + \langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G\rangle + \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H\rangle - \theta| \leqslant \frac{|w_1^{(g_T)}|}{6}\Big]$$

$$= \mathop{\mathbf{Pr}}_{\tilde{\mathcal{D}}_0^R}\Big[\langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G\rangle \in [\theta' - \frac{1}{6}|w_1^{(g_T)}|, \theta' + \frac{1}{6}|w_1^{(g_T)}|]\Big]$$

184

where $\theta' = -\sum_{i=2}^{k}\langle \boldsymbol{w}_i, \boldsymbol{y}_i\rangle - \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H\rangle + \theta$. For any fixing of the value of $\theta' \in \mathbb{R}$, induces a certain distribution on $\boldsymbol{y}_1^G$. However, the $\frac{1}{k^2}$ noise introduced in $\boldsymbol{y}_1^G$ is completely independent. This corresponds to the setting of Lemma 9.3.7, and hence we can bound the above probability by $\left(1 - \frac{1}{2k^2}\right)^T \leqslant \frac{1}{k^2}$. The result follows from averaging over all values of θ'. $\qquad\square$

Theorem 9.4.8. *(Soundness) Fix $\tau = \frac{1}{k^7}$ and t to be set as the same as in Lemma 9.4.7. Let $h(\boldsymbol{x}) = \mathrm{sgn}(\langle \boldsymbol{w}, \boldsymbol{y}\rangle - \theta)$ be a halfspace such that $S_t(\boldsymbol{w}_i) \cap S_t(\boldsymbol{w}_j) = \emptyset$ for all $i, j \in [k]$. Then the halfspace $h(\boldsymbol{y})$ passes the dictatorship test with probability at most $\frac{1}{2} + O(\frac{1}{k})$.*

Proof. The probability of success of $h(\boldsymbol{y})$ is given by $\frac{1}{2} + \frac{1}{2}\left(\mathbf{E}_{\tilde{\mathcal{D}}_1^R}[h(\boldsymbol{y})] - \mathbf{E}_{\tilde{\mathcal{D}}_0^R}[h(\boldsymbol{y})]\right)$. Therefore, it suffices to show that $\left|\mathbf{E}_{\tilde{\mathcal{D}}_0^R}[h(\boldsymbol{y})] - \mathbf{E}_{\tilde{\mathcal{D}}_1^R}[h(\boldsymbol{y})]\right| \leqslant O(\frac{1}{k})$.

Define $K = \{l \mid C_\tau(w_l) \geqslant t\}$. We discuss the following two cases.

1. $K = \emptyset$; i.e., $\forall i \in [k]$, $C_\tau(w_i) \leqslant t$. Then for all i, j, $S_t(\boldsymbol{w}_i) \cap S_t(\boldsymbol{w}_j) = \emptyset$ implies $C_\tau(\boldsymbol{w}_i) \cap C_\tau(\boldsymbol{w}_j) = \emptyset$. By Lemma 9.4.6, we thus have $\left|\mathbf{E}_{\tilde{\mathcal{D}}_0^R}[h(\boldsymbol{y})] - \mathbf{E}_{\tilde{\mathcal{D}}_1^R}[h(\boldsymbol{y})]\right| \leqslant O(\frac{1}{k})$.

2. $K \neq \emptyset$. Then for all $l \in K$, we set $\tilde{\boldsymbol{w}}_\ell = \mathsf{Truncate}(\boldsymbol{w}_\ell, S_t(\boldsymbol{w}_\ell))$ and replace $\boldsymbol{w}_\ell$ with $\tilde{\boldsymbol{w}}_\ell$ in h to get a new halfspace h'. Since such replacements occur at most k times and by Lemma 9.4.7 every replacement changes the output of the halfspace on at most $\frac{1}{k^2}$ fraction of examples, we can bound the overall change by $k \times \frac{1}{k^2} = \frac{1}{k}$. That is

$$\left|\mathbf{E}_{\tilde{\mathcal{D}}_0^R}[h'(\boldsymbol{y})] - \mathbf{E}_{\tilde{\mathcal{D}}_0^R}[h(\boldsymbol{y})]\right| \leqslant \frac{1}{k}, \qquad \left|\mathbf{E}_{\tilde{\mathcal{D}}_1^R}[h'(\boldsymbol{y})] - \mathbf{E}_{\tilde{\mathcal{D}}_1^R}[h(\boldsymbol{y})]\right| \leqslant \frac{1}{k}. \tag{9.3}$$

Also notice that for h' and all $\ell \in [k]$, the critical index of $|C_\tau(\tilde{w}_\ell)|$ is less than t. This reduces the problem to Case 1, and we conclude $\left|\mathbf{E}_{\tilde{\mathcal{D}}_0^R}[h'(\boldsymbol{y})] - \mathbf{E}_{\tilde{\mathcal{D}}_1^R}[h'(\boldsymbol{y})]\right| = O(1/k)$. Along with (9.3) this finishes the proof.

$\qquad\square$

9.5 Reduction from k-UNIQUE LABEL-COVER

In this section, we describe briefly a reduction from k-UNIQUE LABEL-COVER problem to agnostic learning of monomials, thus showing Theorem 9.1.1 under the Unique Games Conjecture (Conjecture 9.2.2). Although our final hardness result only assumes $\mathbf{P} \neq \mathrm{NP}$, we describe the reduction to k-UNIQUE LABEL-COVER for the purpose of illustrating the main idea of our proof.

Let $\mathcal{L}(G(V, E), 1, R, \{\pi^{v,e} \mid v \in V, e \in E\})$ be an instance of k-UNIQUE LABEL-COVER. The reduction will produce a distribution over labeled examples: (y, b) where y lies in $\{0, 1\}^{|V| \times R}$ and label $b \in \{0, 1\}$. We will index the coordinates of $y \in \{0, 1\}^{|V| \times R}$ by $y_w^{(i)}$ (for $w \in V, i \in R$) and denote y_w (for $w \in V$) to be the vector $(y_w^{(1)}, y_w^{(2)}, \ldots, y_w^{(R)})$.

1. Sample an edge $e = (v_1, \ldots, v_k) \in E$
2. Generate a random bit $b \in \{-1, 1\}$.
3. Sample $\boldsymbol{x} \in \{-1, 1\}^{kR}$ from $\tilde{\mathcal{D}}_b^R$.
4. Define $\boldsymbol{y} \in \{-1, 1\}^{|V| \times R}$ as follows:
   (a) For each $v \notin \{v_1, \ldots, v_k\}$, $\boldsymbol{y}_v = \boldsymbol{0}$.
   (b) For each $i \in [k]$ and $j \in [R]$, $y_{v_i}^{(j)} = x_i^{(\pi^{v_i, e}(j))}$.
5. Output the example $(\boldsymbol{y}, b)$.

Proof of Theorem 9.1.1 assuming Unique Games Conjecture Fix $k = \frac{10}{\epsilon^2}$, $\eta = \frac{\epsilon^3}{100}$ and a positive integer $R > \lceil (2k^{29})^{\frac{1}{\eta^2}} \rceil$ for which Conjecture 9.2.2 holds.

Completeness: Suppose that $\mathscr{A} : V \to [R]$ is a labeling that *strongly* satisfies $1 - k\eta$ fraction of the edges. Consider disjunction $h(\boldsymbol{y}) = \bigvee_{v \in V} y_v^{(\mathscr{A}(v))}$. For at least $1 - k\eta$ fraction of edges $e = (v_1, v_2, \ldots, v_k) \in E$, $\pi^{v_1, e}(\mathscr{A}(v_1)) = \cdots = \pi^{v_k, e}(\mathscr{A}(v_k)) = \ell$. As all coordinates of $\boldsymbol{y}$ outside of $\{\boldsymbol{y}_{v_1}, \ldots, \boldsymbol{y}_{v_k}\}$ are set to 0 in step 4(a), the disjunction reduces to $\bigvee_{i \in [k]} y_{v_i}^{(\mathscr{A}(v_i))} = \bigvee_{i \in [k]} x_i^{(\ell)}$. By Theorem 9.4.5, such a disjunction agrees with every $(\boldsymbol{y}, b)$ with probability at least $1 - \frac{3}{\sqrt{k}}$. Therefore $h(\boldsymbol{y})$ agrees with a random example with probability at least $(1 - \frac{3}{\sqrt{k}})(1 - k\eta) \geqslant 1 - \frac{3}{\sqrt{k}} - k\eta \geqslant 1 - \epsilon$.

Soundness: Suppose there exists a halfspace $h(\boldsymbol{y}) = \sum_{v \in V} \langle w_v, y_v \rangle$ that agrees with more than $\frac{1}{2} + \epsilon \geqslant \frac{1}{2} + \frac{1}{\sqrt{k}}$ fraction of the examples. Set $t = k^{12}(3\log(k^6) + \log R) + 4k^2 \log(1/k)) = O(k^{13} \log(R))$ (same as in Theorem 9.4.8). Define the labeling $\mathscr{A}$ using the following strategy : for each vertex $v \in V$ randomly pick a label from $S_t(w_v)$.

By an averaging argument, for at least $\frac{\epsilon}{2}$ fraction of the edges $e \in E$ generated in step 1 of the reduction, $h(\boldsymbol{y})$ agrees with the examples corresponding to e with probability at least $\frac{1}{2} + \frac{\epsilon}{2}$. We will refer to such edges as *good*. By Theorem 9.4.8 for each *good* edge $e \in E$, there exists $i, j \in [k]$, such that $\pi^{v_i, e}(S_t(\boldsymbol{w}_{v_i})) \cap \pi^{v_j, e}(S_t(\boldsymbol{w}_{v_j})) \neq \emptyset$. Therefore the edge $e \in E$ is *weakly* satisfied by the labeling $\mathscr{A}$ with probability at least $\frac{1}{t^2}$. Hence, in expectation the labelling $\mathscr{A}$ *weakly* satisfies at least $\frac{\epsilon}{2} \cdot \frac{1}{t^2} = \Omega(\frac{1}{k^{27} \log^2 R}) \geqslant \frac{2k^2}{R^{\eta/4}}$ fraction of the edges (by the choice of R and t).

9.6 Reduction from Label Cover

In this section, we describe a reduction from the bipartite Label Cover problem to a k-LABEL-COVER with an additional *smoothness* property. We then reduce the smooth k-LABEL-COVER problem to agnostic learning of disjunctions by halfspaces. This will give us Theorem 9.1.1 without assuming the Unique Games Conjecture.

9.6.1 Smooth k-LABEL-COVER

Our reduction use the following hardness result for k-LABEL-COVER (Definition 9.2.1) with the additional smoothness property.

Theorem 9.6.1. *There exists a constant $\gamma > 0$ such that for any integer parameter $J, u \geqslant 1$, it is NP-hard to distinguish between the following two types of k-LABEL-COVER $\mathcal{L}(G(V, E), N, M, \{\pi^{v,e} | e \in E, v \in e\})$ instances with $M = 7^{(J+1)u}$ and $N = 2^u 7^{Ju}$:*

1. *(Strongly satisfiable instances) There is some labeling that strongly satisfies every hyperedge.*

2. *(Instances that are not $2k^2 2^{-\gamma u}$-weakly satisfiable) There is no labeling that weakly satisfies at least $2k^2 2^{-\gamma u}$ fraction of the hyperedges.*

In addition, the k-LABEL-COVER instances have the following properties:

- *(Smoothness) for a fixed vertex w and a randomly picked hyperedge containing w,*

$$\forall i, j \in [M], \mathbf{Pr}[\pi^{w,e}(i) = \pi^{w,e}(j)] \leqslant 1/J.$$

- *For any mapping $\pi^{e,v}$ and any number $i \in [N]$, we have $|(\pi^{e,v})^{-1}(i)| \leqslant d = 4^u$; i.e., there are at most $d = 4^u$ elements in $[M]$ that are mapped to the same number in $[N]$.*

The proof of the above theorem can be found in Appendix 9.10.

In the rest of the thesis, we will set $u = k$ and therefore $d = 4^k$. Also we set the smoothness parameter $J = d^{17} = 4^{17k}$.

9.6.2 Reduction from Smooth k-LABEL-COVER

The starting point is a smooth k-LABEL-COVER $\mathcal{L}(G(V, E), N, M, \{\pi^{v,e} | e \in E, v \in e\})$ with $M = 7^{(J+1)u}$ and $N = 2^u 7^{Ju}$ as described in Theorem 9.6.1. Following below is the reduction from k-LABEL-COVER $\mathcal{L}(G(V, E), N, M, \{\pi^{v,e} | e \in E, v \in e\})$ that given an instance of k-LABEL-COVER $\mathcal{L}$ produces a random labeled example. We refer to the obtained distribution on examples as $\mathcal{E}$.

- Pick a hyperedge $e = (v_1, v_2, \ldots, v_k) \in E$ with corresponding projections $\pi_1, \ldots, \pi_k : [M] \to [N]$.

- Generate a random bit $b \in \{-1, 1\}$.

- Sample $\boldsymbol{x} \in \{-1, 1\}^{kR}$ from $\mathscr{D}_b^N$.

- Generate $\boldsymbol{y} \in \{-1, 1\}^{|V| \times M}$ as follows:

  1. For each $v \notin e$, $\boldsymbol{y}_v = \boldsymbol{0}$.
  2. For each $i \in [k]$, set $\boldsymbol{y}_{v_i} \in \{-1, 1\}^M$ as follows:

$$y_{v_i}^{(j)} = \begin{cases} x_i^{(\pi_i(j))} & \text{with probability } 1 - \frac{1}{k^2} \\ \text{random bit} & \text{with probability } \frac{1}{k^2} \end{cases}$$

- Output the example $(\boldsymbol{y}, b)$ or equivalently ACCEPT if $h(\boldsymbol{y}) = b$.

9.6.3   Proof of Theorem 9.1.1

We claim that our reduction has the following completeness and soundness properties.

Theorem 9.6.2.   • COMPLETENESS: *If $\mathscr{L}$ is a strongly-satisfiable instance of smooth k-LABEL-COVER, then there is a disjunction that agrees with a random example from $\mathscr{E}$ with probability at least $1 - O(\frac{1}{\sqrt{k}})$.*

   • SOUNDNESS: *If $\mathscr{L}$ is not $2k^2 2^{-\gamma k}$-weakly satisfiable, then there is no halfspace that agrees with a random example from $\mathscr{E}$ with probability more than $\frac{1}{2} + O(\frac{1}{\sqrt{k}})$.*

Combining the above theorem with Theorem 9.6.1 we get that for $k = O(1/\epsilon^2)$, we obtain our main result: Theorem 9.1.1.

It remains to check the correctness of the completeness and soundness claims in Theorem 9.6.2. First let us prove the completeness property.

Proof. (Proof of Completeness) Let L be the labeling that strongly satisfies $\mathscr{L}$. Consider disjunction $h(\boldsymbol{y}) = \bigvee_{v \in V} y_v^{(L(v))}$. Let $e = (v_1, v_2, \ldots, v_k)$ be any hyperedge and let $\mathscr{E}_e$ be the distribution $\mathscr{E}$ restricted to the examples generated for e. With probability at least $1 - 1/k$, $y_{v_i}^{L(v_i)} = x_i^{\pi^{e,v_i}(L(v_i))}$ for every $i \in [k]$. As e is strongly satisfied by L, for all $i, j \in [k]$, $\pi^{e,v_i}(L(v_i)) = \pi^{e,v_j}(L(v_j))$. Therefore, as in the proof of Theorem 9.4.5, we obtain that $h(\boldsymbol{y})$ agrees with a random example from $\mathscr{E}_e$ with probability at least $1 - O(1/\sqrt{k})$. Labeling L strongly satisfies all edges and therefore we obtain that $h(\boldsymbol{y})$ agrees with a random example from $\mathscr{E}$ with probability at least $1 - O(1/\sqrt{k})$. $\square$

The more complicated part is the soundness property which we prove in Section 9.6.4.

9.6.4   Soundness Analysis

Let $h(\boldsymbol{y})$ be a halfspace that agrees with more than $\frac{1}{2} + \frac{1}{\sqrt{k}}$-fraction of the examples. Suppose,

$$h(\boldsymbol{y}) = \text{sgn}\Big(\sum_{v \in V} \langle \boldsymbol{w}_v, \boldsymbol{y}_v \rangle - \theta \Big).$$

Let $\tau = \frac{1}{k^{13}}$ and let

$$\boldsymbol{s}_v = \text{Truncate}(\boldsymbol{w}_v, C_\tau(\boldsymbol{w}_v)), \quad \boldsymbol{l}_v = \boldsymbol{w}_v - \boldsymbol{s}_v.$$

Definition 9.6.3. *A vertex $v \in V$ is said to be β-nice with respect to a hyperedge $e \in E$ containing it if*

$$\sum_{i \in [N]} \Big(\sum_{j \in \pi^{-1}(i)} |l_v^{(j)}| \Big)^4 \leqslant \beta \|\boldsymbol{l}_v\|_2^4$$

where $\pi : [M] \to [N]$ is the projection associated with vertex v and hyperedge e. In other words,

$$\sum_{i \in [N]} \Big(\|\boldsymbol{l}_v^{\{e,i\}}\|_1 \Big)^4 \leqslant \beta \|\boldsymbol{l}_v\|_2^4.$$

188

An hyperedge $e = (v_1, v_2, \ldots, v_k)$ is β-nice, if for every $i \in [k]$, the vertex v_i is β-nice with respect to e.

Lemma 9.6.4. *The fraction of 2τ-nice hyperedges in E is at least $1 - O(1/k)$.*

Proof. By definition, we know that $\boldsymbol{l}_v$ is τ-regular vector. Denote $I_v = \{i \mid \frac{(l_v^{(i)})^2}{\|\boldsymbol{l}_v\|_2^2} \geqslant \frac{1}{d^8}\}$. Therefore, $|I| \leqslant d^8$. Notice there are at most d^{16} pairs of values in $I \times I$. By the smoothness property of the k-LABEL-COVER instance, for any vertex v, at leat $1 - \frac{d^{16}}{J}$ fraction of the hyperedges incident on v have the following property: for any $i, j \in I$, $\pi^{e,v}(i) \neq \pi^{e,v}(j)$. If all the vertices in an hyperedge have this property we call it a *good* hyperedge. By an averaging argument, we know that among all hyperedges at least $1 - \frac{kd^{16}}{J} = 1 - \frac{k}{4^k} \geqslant 1 - O(\frac{1}{k})$ fraction is *good*.

We will show all these *good* hyperedges are also 2τ-nice. For a given *good* hyperedge e, a vertex $v \in e$, $\pi = \pi^{e,v}$ and $i \in [N]$, there is at most one $j \in \pi^{-1}(i)$ such that $\frac{(l_v^{(i)})^2}{\|\boldsymbol{l}_v\|_2^2} \geqslant \frac{1}{d^8}$.

Based on the above property, we will show

$$\sum_{i \in [N]} \left(\sum_{j \in \pi^{-1}(i)} |l_v^{(j)}| \right)^4 \leqslant 2\tau \|\boldsymbol{l}_v\|_2^4$$

Notice that

$$\sum_{i \in N} \left(\sum_{j \in \pi^{-1}(i)} |l_v^{(j)}| \right)^4 = \sum_{i \in N} \sum_{j_1, j_2, j_3, j_4 \in \pi^{-1}(i)} \left| l_v^{(j_1)} l_v^{(j_2)} l_v^{(j_3)} l_v^{(j_4)} \right| \tag{9.4}$$

and the sum of all the terms with $j_1 = j_2 = j_3 = j_4$ is $\|\boldsymbol{l}_v\|_4^4$.

For other term $|l_v^{(j_1)} l_v^{(j_2)} l_v^{(j_3)} l_v^{(j_4)}|$ such that j_1, j_2, j_3, j_4 are not all equal, there is at least one $|l_v^{(j_r)}|$ ($r \in [4]$) smaller than $\frac{\|\boldsymbol{l}_v\|_2}{d^4}$. Therefore, $|l_v^{(j_1)} l_v^{(j_2)} l_v^{(j_3)} l_v^{(j_4)}|$ can be bounded by

$$\frac{\|\boldsymbol{l}_v\|_2}{d^4} \left(\sum_{j_1, j_2, j_3, j_4} (l_v^{(j_1)})^3 + (l_v^{(j_2)})^3 + (l_v^{(j_3)})^3 + (l_v^{(j_4)})^3 \right).$$

Overall, expression (9.4) can be bounded by

$$\|\boldsymbol{l}_v\|_4^4 + \frac{\|\boldsymbol{l}_v\|_2}{d^4} \sum_{j_1, j_2, j_3, j_4} (l_v^{(j_1)})^3 + (l_v^{(j_2)})^3 + (l_v^{(j_3)})^3 + (l_v^{(j_4)})^3$$

$$\leqslant \tau^2 \|\boldsymbol{l}_v\|_2^4 + \frac{\|\boldsymbol{l}_v\|_2}{d^4} 4d^3 \|\boldsymbol{l}_v\|_3^3 \qquad \text{(each term is counted at most $4d^3$ times)}$$

$$\leqslant (\tau^2 + 4\frac{\tau}{d}) \|\boldsymbol{l}_v\|_2^4 \qquad \text{($\boldsymbol{l}_v$ is τ-regular vector)}$$

$$\leqslant 2\tau \|\boldsymbol{l}_v\|_2^4.$$

$\square$

Let us fix a 2τ-nice hyperedge $e = (v_1, \ldots, v_k)$. As before let $\mathscr{E}_e$ denote the distribution on examples restricted to those generated for hyperedge e. We will analyze the probability that the halfspace $h(\boldsymbol{y})$ agrees with a random example from $\mathscr{E}_e$.

189

Let $\pi_1, \pi_2, \ldots, \pi_k : [M] \to [N]$ denote the projections associated with the hyperedge e. For the sake of brevity, we shall write $\boldsymbol{w}_i, \boldsymbol{y}_i, \boldsymbol{l}_i$ instead of $\boldsymbol{w}_{v_i}, \boldsymbol{y}_{v_i}, \boldsymbol{l}_{v_i}$. For all $j \in [N]$ and $i \in [k]$, define

$$\boldsymbol{y}_i^{\{j\}} = \mathsf{Truncate}(\boldsymbol{y}_i, \pi_i^{-1}(j)).$$

Similarly, define vectors $\boldsymbol{w}_i^{\{j\}}, \boldsymbol{l}_i^{\{j\}}$ and $\boldsymbol{s}_i^{\{j\}}$.

Notice that for every example $(\boldsymbol{y}, b)$ in the support of $\mathcal{E}_e$, $\boldsymbol{y}_v = \boldsymbol{0}$ for every vertex $v \notin e$. Therefore, on restricting to examples from $\mathcal{E}_e$ we can write:

$$h(\boldsymbol{y}) = \mathsf{sgn}\Big(\sum_{i \in [k]} \langle \boldsymbol{w}_i, \boldsymbol{y}_i \rangle - \theta \Big).$$

Common Influential Variables

Lemma 9.6.5. *(Common Influential Coordinate) Let $h(\boldsymbol{y})$ be a halfspace such that for all $i \neq j \in [k]$, we have $\pi_i(C_\tau(\boldsymbol{w}_i)) \cap \pi_j(C_\tau(\boldsymbol{w}_j)) = \emptyset$. Then*

$$\Big| \mathbf{E}_{\mathcal{E}_e}[h(\boldsymbol{y})|b = 0] - \mathbf{E}_{\mathcal{E}_e}[h(\boldsymbol{y})|b = 1] \Big| \leqslant O\Big(\frac{1}{k}\Big). \tag{9.5}$$

Proof. Fix the following notation:

$$\boldsymbol{y}_i^L = \mathsf{Truncate}(\boldsymbol{y}_i, C_\tau(\boldsymbol{w}_i)) \qquad\qquad \boldsymbol{y}^L = \boldsymbol{y}_1^L, \boldsymbol{y}_2^L, \ldots, \boldsymbol{y}_k^L$$
$$\boldsymbol{s} = \boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_k \qquad\qquad \boldsymbol{l} = \boldsymbol{l}_1, \boldsymbol{l}_2, \ldots, \boldsymbol{l}_k.$$

We can rewrite the halfspace $h(\boldsymbol{y})$ as $h(\boldsymbol{y}) = \mathsf{sgn}\Big(\langle \boldsymbol{s}, \boldsymbol{y}^L \rangle + \langle \boldsymbol{l}, \boldsymbol{y} \rangle - \theta \Big)$. Let us first normalize the weights of $h(\boldsymbol{y})$ so that $\sum_{i \in [k]} \|\boldsymbol{l}_i\|_2^2 = 1$. Let us condition on a possible fixing of the vector $\boldsymbol{y}^L$. Under this conditioning and also for $b = 0$, define the family of ensembles $\mathscr{A} = \boldsymbol{A}^{\{1\}}, \ldots, \boldsymbol{A}^{\{N\}}$ as follows:

$$\boldsymbol{A}^{\{j\}} = \Big\{ y_i^{(\ell)} \mid i \in [k], \ell \in [M] \text{ such that } \pi_i(\ell) = j \text{ and } \ell \notin C_\tau(\boldsymbol{w}_i) \Big\}$$

Similarly define the ensemble $\mathscr{B} = \boldsymbol{B}^{\{1\}}, \ldots, \boldsymbol{B}^{\{N\}}$ for the conditioning $b = 1$. Now we shall apply the invariance principle (Theorem 9.3.10) to the ensembles $\mathscr{A}, \mathscr{B}$ and the linear function $\boldsymbol{l}(\boldsymbol{y})$:

$$\boldsymbol{l}(\boldsymbol{y}) = \sum_{j \in [N]} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{y}^{\{j\}} \rangle.$$

As we prove in Claim 9.6.6 below, the ensembles $\mathscr{A}, \mathscr{B}$ have matching moments up to degree 3. Furthermore, by Lemma 9.3.3, the linear function $\boldsymbol{l}$ and the ensembles $\mathscr{A}, \mathscr{B}$ satisfy the following spread property:

$$\mathbf{Pr}_{\mathscr{A}}\Big[\ell(\mathscr{A}) \in [\theta' - \alpha, \theta' + \alpha] \Big] \leqslant c(\alpha) \qquad\qquad \mathbf{Pr}_{\mathscr{B}}\Big[\ell(\mathscr{B}) \in [\theta' - \alpha, \theta' + \alpha] \Big] \leqslant c(\alpha)$$

for all $\theta' \in \mathbb{R}$, where $c(\alpha) = 8\alpha k + 4\tau k + 2e^{-\frac{1}{2k^4\tau^2}}$ (by setting $\gamma = \frac{1}{k^2}$ and $|b - a| = 2\alpha$ in Lemma 9.3.3).

Using the invariance principle (Th. 9.3.10), this implies:

$$\left| \underset{\mathscr{A}}{\mathbf{E}}\left[\mathrm{sgn}\left(\langle \boldsymbol{s}, \boldsymbol{y}^L \rangle + \sum_{j\in[N]} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{A}^{\{j\}} \rangle - \theta \right) | \boldsymbol{y}^L \right] - \underset{\mathscr{B}}{\mathbf{E}}\left[\mathrm{sgn}\left(\langle \boldsymbol{s}, \boldsymbol{y}^L \rangle + \sum_{j\in[N]} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{B}^{\{j\}} \rangle - \theta \right) | \boldsymbol{y}^L \right] \right|$$

$$\leqslant O(\frac{1}{\alpha^4}) \sum_{j\in[N]} \|\boldsymbol{l}^{\{j\}}\|_1^4 + 2c(\alpha). \quad (9.6)$$

Take α to be $\frac{1}{k^2}$ and recall that $\tau = \frac{1}{k^{13}}$. In Claim 9.6.7 below we show that

$$\sum_{j\in[N]} \|\boldsymbol{l}^{\{j\}}\|_1^4 \leqslant 2\tau k^4.$$

The above inequality holds for an arbitrary conditioning of the values of $\boldsymbol{y}^L$. Hence, by averaging over all settings of $\boldsymbol{y}^L$ we get that expression (9.6) is bounded by $O(1/k)$. $\qquad \square$

Claim 9.6.6. *The ensembles $\mathscr{A}$ and $\mathscr{B}$ have matching moments up to degree 3.*

Let us suppose for a moment that $\boldsymbol{y}$ was generated by setting $y_{v_i}^{(j)} = x_i^{(\pi_i(j))}$, that is without adding any noise. By Lemma 9.4.1, the first moments of random variable $\boldsymbol{y}$ conditioned on $b = 0$ agree with the first moments of random variable $\boldsymbol{y}$ conditioned on $b = 1$. As we showed in Observation 9.4.3, even with noise, the first four moments of $\boldsymbol{y}$ remain the same when conditioned on $b = 0$ and $b = 1$. Finally, $\pi_i(C_\tau(\boldsymbol{w}_i)) \cap \pi_j(C_\tau(\boldsymbol{w}_j)) = \varnothing$ for all $i \neq j \in [k]$. Hence for each $j \in [N]$, conditioning on $\boldsymbol{y}^L$ fixes bits in at most one row of $\boldsymbol{A}^{\{j\}}$. Formally, for every $j \in [N]$, there exists at most one $i \in [k]$ such that $\boldsymbol{y}_i^{\{j\}}$ and $\boldsymbol{y}^L \neq \phi$ have shared variables. Therefore, by Lemma 9.4.4, $\mathscr{A}$ and $\mathscr{B}$ have matching moments up to degree 3.

Claim 9.6.7.

$$\sum_{j\in[N]} \|\boldsymbol{l}^{\{j\}}\|_1^4 \leqslant 2\tau k^4 .$$

Proof. Since $\|\boldsymbol{l}^{\{j\}}\|_1 = \sum_{i\in[k]} \|\boldsymbol{l}_i^{\{j\}}\|_1$, we can write

$$\sum_{j\in[N]} \|\boldsymbol{l}^{\{j\}}\|_1^4 \leqslant \sum_{j\in N} k^4 \left(\sum_{i\in[k]} \|\boldsymbol{l}_i^{\{j\}}\|_1^4 \right) = k^4 \sum_{i\in[k]} \left(\sum_{j\in[N]} \|\boldsymbol{l}_i^{\{j\}}\|_1^4 \right). \quad (9.7)$$

As $e = (v_1,\dots,v_k)$ is a 2τ-nice hyperedge, we have $\sum_{j\in[N]} \|\boldsymbol{l}_i^{\{j\}}\|_1^4 \leqslant 2\tau \|\boldsymbol{l}_i\|_2^4$. By normalization of $\boldsymbol{l}$, we know $\sum_{i\in[k]} \|\boldsymbol{l}_i\|_2^2 = 1$. Substituting this into inequality (9.7) we get the claimed bound. $\qquad \square$

Bounding the Number of Influential Coordinates

Lemma 9.6.8. *Given a halfspace $h(\boldsymbol{y}) = \mathrm{sgn}(\sum_{i\in[k]} \langle \boldsymbol{w}_i, \boldsymbol{y}_i \rangle - \theta)$ and $\ell \in [k]$ such that $|C_\tau(\boldsymbol{w}_\ell)| \geqslant t$ for $t = \frac{1}{\tau^2}(16k^2 \log(1/k)\log(1/\tau) + \ln(1/\tau) + 10\ln d) = O(k^{30})$, define $\tilde{h}(\boldsymbol{y}) = \mathrm{sgn}(\sum_{i\in[k]} \langle \tilde{\boldsymbol{w}}_i, \boldsymbol{y}_i \rangle - \tilde{\theta})$ as follows:*

- $\tilde{\boldsymbol{w}}_\ell = \mathsf{Truncate}(\boldsymbol{w}_\ell, S_t(\boldsymbol{w}_\ell))$ *and* $\tilde{\boldsymbol{w}}_i = \boldsymbol{w}_i$ *for all* $i \neq \ell$.
- $\tilde{\theta} = \theta - \mathbf{E}[\langle \boldsymbol{a}_\ell, \boldsymbol{y}_\ell \rangle | b = 0]$, *for* $\boldsymbol{a} = \boldsymbol{w} - \tilde{\boldsymbol{w}}$.

Then,

$$\left| \mathbf{E}_{\mathscr{E}_e}[\tilde{h}(\boldsymbol{y})|b=0] - \mathbf{E}_{\mathscr{E}_e}[h(\boldsymbol{y})|b=0] \right| \leqslant \frac{1}{k^2}, \qquad \left| \mathbf{E}_{\mathscr{E}_e}[\tilde{h}(\boldsymbol{y})|b=1] - \mathbf{E}_{\mathscr{E}_e}[h(\boldsymbol{y})|b=1] \right| \leqslant \frac{1}{k^2}.$$

Proof. It is easy to see that the matching moments condition implies that

$$\mathbf{E}_{\mathscr{E}_e}[\langle \boldsymbol{a}_\ell, \boldsymbol{y}_\ell \rangle | b=0] = \mathbf{E}_{\mathscr{E}_e}[\langle \boldsymbol{a}_\ell, \boldsymbol{y}_\ell \rangle | b=1].$$

Let us show the inequality for the case $b = 0$, the other inequality can be derived in an identical way. Let $\mathscr{E}_{e,0}$ denote distribution $\mathscr{E}_e$ conditioned on $b = 0$. Without loss of generality, we may assume that $\ell = 1$ and $|w_1^{(1)}| \geqslant |w_1^{(2)}| \ldots \geqslant |w_1^{(M)}|$. In particular, this implies $S_t(\boldsymbol{w}_1) = \{1, \ldots, t\}$. Define

$$\mu_\ell = \mathbf{E}_{\mathscr{E}_{e,0}}[\langle \boldsymbol{a}_\ell, \boldsymbol{y}_\ell \rangle], \qquad \mu_\ell^{\{i\}} = \mathbf{E}_{\mathscr{E}_{e,0}}[\langle \boldsymbol{a}_\ell^{\{i\}}, \boldsymbol{y}_\ell^{\{i\}} \rangle].$$

Let us set $T = \lceil 4k^2 \log(1/k) \rceil$ and define the subset $G = \{g_1, \ldots, g_T\}$ of $S_t(\boldsymbol{w}_1)$ as follows:

$$G = \{g_i \mid g_i = 1 + i \lceil (4/\tau^2) \ln(1/\tau) \rceil, 0 \leqslant i \leqslant T\}.$$

Therefore, by Lemma 9.3.2, $|w_1^{(g_i)}|$ is a geometrically decreasing sequence such that $|w_1^{(g_{i+1})}| \leqslant |w_1^{(g_i)}|/3$. Let $H = S_t(\boldsymbol{w}_1) \setminus G$. Fix the following notation:

$$\boldsymbol{w}_1^G = \mathsf{Truncate}(\boldsymbol{w}_1, G), \qquad \boldsymbol{w}_1^H = \mathsf{Truncate}(\boldsymbol{w}_1, H), \qquad \boldsymbol{w}_1^{>t} = \mathsf{Truncate}(\boldsymbol{w}_1, \{t+1, \ldots, n\}).$$

Similarly, define the vectors $\boldsymbol{y}_1^G, \boldsymbol{y}_1^H, \boldsymbol{y}_1^{>t}$. By definition, we have $\boldsymbol{a}_1 = \boldsymbol{w}_1^{>t}$. Rewriting the halfspace functions $h(\boldsymbol{y}), \tilde{h}(\boldsymbol{y})$:

$$h(\boldsymbol{y}) = \mathsf{sgn}\Big(\sum_{i=2}^{k} \langle \boldsymbol{w}_i, \boldsymbol{y}_i \rangle + \langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G \rangle + \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H \rangle + \langle \boldsymbol{a}_1, \boldsymbol{y}_1^{>t} \rangle - \theta \Big),$$

$$\tilde{h}(\boldsymbol{y}) = \mathsf{sgn}\Big(\sum_{i=2}^{k} \langle \boldsymbol{w}_i, \boldsymbol{y}_i \rangle + \langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G \rangle + \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H \rangle + \mu_1 - \theta \Big).$$

By Claim 9.6.9 below, with probability at most $\frac{1}{d} = \frac{1}{4^k}$, we have $|\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle - \mu_1| \geqslant d^4 \|\boldsymbol{a}_1\|_2$. Suppose $|\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle - \mu_1| < d^4 \|\boldsymbol{a}_1\|_2$, then Claim 9.6.10 below gives $|\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle - \mu_1| < 1/d^6 |w_1^{(g_T)}| < \frac{1}{3} |w_1^{(g_T)}|$. Thus, we can write

$$\mathbf{Pr}_{\mathscr{E}_{e,0}}\Big[h(\boldsymbol{y}) \neq \tilde{h}(\boldsymbol{y}) \Big] \leqslant \mathbf{Pr}_{\mathscr{E}_{e,0}}\Big[\langle \boldsymbol{w}_1^G, \boldsymbol{y}_1^G \rangle \in [\theta' - \frac{1}{3}|w_1^{(g_T)}|, \theta' + \frac{1}{3}|w_1^{(g_T)}|] \Big] + \frac{1}{4^k}.$$

where $\theta' = -\sum_{i=2}^{k} \langle \boldsymbol{w}_i, \boldsymbol{y}_i \rangle - \langle \boldsymbol{w}_1^H, \boldsymbol{y}_1^H \rangle - \mu_1 + \theta$. For any fixing of the value of $\theta' \in \mathbb{R}$, induces a certain distribution on $\boldsymbol{y}_1^G$. However, the $\frac{1}{k^2}$ noise introduced in $\boldsymbol{y}_1^G$ is completely independent. This corresponds to the setting of Lemma 9.3.7, and hence we can bound the above probability by $(1 - 1/(2k^2))^T + 1/4^k \leqslant (1 - 1/(2k^2))^{4k^2 \log(1/k)} + 1/4^k \leqslant 1/k^2$. $\qquad \square$

Claim 9.6.9.
$$\mathbf{Pr}_{\mathscr{E}_{e,0}}\Big[|\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle - \mu_1| \geqslant d^4 \|\boldsymbol{a}_1\|_2 \Big] \leqslant \frac{1}{d}.$$

192

Proof. We claim:

$$\mathbf{Var}_{\mathscr{E}_{e,0}}\big(\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle\big) \leqslant td\|\boldsymbol{a}_1\|_2^2 + d\|\boldsymbol{a}_1\|_2^2 \leqslant 2td\|\boldsymbol{a}_1\|_2^2.$$

Notice that $[M]$ can be seen as the union of disjoint sets $R_1 \cup R_2 \cup \cdots \cup R_N$ where $R_i = \pi_1^{-1}(i)$. There are at most t sets such that $R_i \cap S_t(\boldsymbol{w}_1) \neq \emptyset$ and, by the property of our k-LABEL-COVER instance, there are at most td indices in those sets. Let $U_1 = \cup_{R_i \cap S_t(\boldsymbol{w}_1) \neq \emptyset} R_i$ and let $U_2 = S \setminus U_1$. It is easy to see that $\boldsymbol{y}_1^{U_1}$ is independent of $\boldsymbol{y}_1^{U_2}$ and therefore

$$\mathbf{Var}_{\mathscr{E}_{e,0}}\big(\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle - \mu_1\big) = \mathbf{Var}_{\mathscr{E}_{e,0}}\big(\langle \boldsymbol{a}_1^{U_1}, \boldsymbol{y}_1^{U_1} \rangle\big) + \mathbf{Var}_{\mathscr{E}_{e,0}}\big(\langle \boldsymbol{a}_1^{U_2}, \boldsymbol{y}_1^{U_2} \rangle\big).$$

The variance of $\langle \boldsymbol{a}_1^{U_1}, \boldsymbol{y}_1^{U_1} \rangle$ is at most

$$\|\boldsymbol{a}_1^{U_1}\|_1^2 \leqslant td\|\boldsymbol{a}_1^{U_1}\|_2^2 \leqslant td\|\boldsymbol{a}_1\|_2^2.$$

Notice that U_2 is the union of all the R_i's that do not intersect with $S_t(\boldsymbol{w}_1)$. Further for any $i, j \in [N]$ such that $R_i \cap S_t(\boldsymbol{w}_1) = \emptyset$ and $R_j \cap S_t(\boldsymbol{w}_1) = \emptyset$, $\boldsymbol{y}_1^{R_i}$ is independent of $\boldsymbol{y}_1^{R_j}$. Also since every R_i has size at most d,

$$\mathbf{Var}_{\mathscr{E}_{e,0}}\big(\langle \boldsymbol{a}_1^{U_2}, \boldsymbol{y}_1^{U_2} \rangle\big) = \sum_{R_i \cap S_t(\boldsymbol{w}_1) = \emptyset} \mathbf{Var}_{\mathscr{E}_{e,0}}\big(\langle \boldsymbol{a}_1^{R_i}, \boldsymbol{y}_1^{R_i} \rangle\big) \leqslant \sum_{R_i \cap S_t(\boldsymbol{w}_1) = \emptyset} d\|\boldsymbol{a}_1^{R_i}\|_2^2 = d\|\boldsymbol{a}_1^{U_2}\|_2^2 \leqslant d\|\boldsymbol{a}_1\|_2^2.$$

Overall, we have

$$\mathbf{Var}_{\mathscr{E}_{e,0}}\big(\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle\big) \leqslant td\|\boldsymbol{a}_1\|_2^2 + d\|\boldsymbol{a}_1\|_2^2 \leqslant 2td\|\boldsymbol{a}_1\|_2^2.$$

Notice that $t = \mathrm{poly}(\log d)$ and by applying Chebyshev's inequality (Th. 9.7.3), we have

$$\mathbf{Pr}_{\mathscr{E}_{e,0}}\big[|\langle \boldsymbol{a}_1, \boldsymbol{y}_1 \rangle - \mu_1| \geqslant d^4\|\boldsymbol{a}_1\|_2\big] \leqslant \frac{2td}{d^8} \leqslant \frac{1}{d}.$$

$\square$

Claim 9.6.10. *By the choice of the parameters T and t,*

$$\|\boldsymbol{a}_1\|_2 \leqslant \frac{1}{d^{10}} |w_1^{(g_T)}|.$$

Proof. By Lemma 9.3.2,

$$|w_1^{(g_T)}|^2 \geqslant \frac{\tau}{(1-\tau^2)^{t-g_T}}\|a_1\|_2^2 \geqslant \frac{\tau}{(1-\tau^2)^{\frac{1}{\tau^2}(\ln(1/\tau)+10\ln d)}}\|a_1\|_2^2 \geqslant d^{10}\|a_1\|_2^2.$$

$\square$

193

Soundness Theorem

Recall that we chose $\tau = 1/k^{13}$ and $t = O(k^{30})$.

Lemma 9.6.11. *Fix a hyperedge e which is 2τ-nice. If for all $i \neq j \in [k]$, $\pi_i\big(S_t(\boldsymbol{w}_i)\big) \cap \pi_j\big(S_t(\boldsymbol{w}_j)\big) = \emptyset$ then the probability that halfspace $h(\boldsymbol{y})$ agrees with a random example from $\mathscr{E}_e$ is at most $\frac{1}{2} + O(\frac{1}{k})$.*

Proof. The proof is similar to the proof of Theorem 9.4.8. Define $K = \{\ell \mid C_\tau(w_\ell) > t\}$. We divide the problem into the following two cases.

1. $K = \emptyset$; i.e., for all $i \in [k]$, $C_\tau(w_i) \leqslant t$. Then for any $i \neq j \in [k]$, $S_t(\boldsymbol{w}_i) \cap S_t(\boldsymbol{w}_j) = \emptyset$ implies $C_\tau(\boldsymbol{w}_i) \cap C_\tau(\boldsymbol{w}_j) = \emptyset$. By Lemma 9.6.5, we have

$$\left| \mathop{\mathbf{E}}_{\mathscr{E}_e}[h(\boldsymbol{y})|b = 0] - \mathop{\mathbf{E}}_{\mathscr{E}_e}[h(\boldsymbol{y})|b = 1] \right| \leqslant O\left(\frac{1}{k}\right).$$

2. $K \neq \emptyset$. Then for all $\ell \in K$, we set $\tilde{\boldsymbol{w}}_\ell = \mathsf{Truncate}(\boldsymbol{w}_\ell, S_t(\boldsymbol{w}_\ell))$ and define a new halfspace h' by replacing $\boldsymbol{w}_\ell$ with $\tilde{\boldsymbol{w}}_\ell$ in h. Since such replacements occur at most k times and, by Lemma 9.6.8, every replacement changes the output of the halfspace on at most $\frac{1}{k^2}$ fraction of examples from $\mathscr{E}_e$, we can bound the overall change by $k \times \frac{1}{k^2} = \frac{1}{k}$. That is

$$\left| \mathop{\mathbf{E}}_{\mathscr{E}_{e,0}}[h'(\boldsymbol{y})] - \mathop{\mathbf{E}}_{\mathscr{E}_{e,0}}[h(\boldsymbol{y})] \right| \leqslant \frac{1}{k}, \qquad \left| \mathop{\mathbf{E}}_{\mathscr{E}_{e,1}}[h'(\boldsymbol{y})] - \mathop{\mathbf{E}}_{\mathscr{E}_{e,1}}[h(\boldsymbol{y})] \right| \leqslant \frac{1}{k}. \tag{9.8}$$

For the halfspace h' and for all $\ell \in [k]$, we have $|C_\tau(\tilde{\boldsymbol{w}}_\ell)| \leqslant t$, thus reducing to Case 1. Therefore

$$\left| \mathop{\mathbf{E}}_{\mathscr{E}_{e,o}}[h'(\boldsymbol{y})] - \mathop{\mathbf{E}}_{\mathscr{E}_{e,1}}[h'(\boldsymbol{y})] \right| \leqslant O\left(\frac{1}{k}\right). \tag{9.9}$$

Combining (9.8) and (9.9), we get

$$\left| \mathop{\mathbf{E}}_{\mathscr{E}_{e,0}}[h(\boldsymbol{y})] - \mathop{\mathbf{E}}_{\mathscr{E}_{e,1}}[h(\boldsymbol{y})] \right| \leqslant O\left(\frac{1}{k}\right).$$

In other words, the probability that halfspace $h(\boldsymbol{y})$ agrees with a random example from $\mathscr{E}_e$ is at most $\frac{1}{2} + O(\frac{1}{k})$. $\qquad \square$

We first recall the soundness statement:

Proposition 9.6.12. *If $\mathscr{L}$ is not a $2k^2 2^{-\gamma k}$-weakly satisfiable instance of smooth k-LABEL-COVER, then there is no halfspace that agrees with a random example from $\mathscr{E}$ with probability more than $\frac{1}{2} + \frac{1}{\sqrt{k}}$.*

Proof. The proof is by contradiction. We can define the following labeling strategy: for each vertex v, uniformly randomly pick a label from $S_t(w_v)$. We know the size of $S_t(w_{v_i})$ is $t = O(k^{30})$.

Suppose there exists a halfspace that agrees with a random example from $\mathscr{E}$ with probability more than $\frac{1}{2} + \frac{1}{\sqrt{k}}$. Then by an averaging argument, for at least $\frac{1}{2\sqrt{k}}$-fraction of the

hyperedges e, $h(y)$ agrees with a random example from $\mathcal{E}_e$ with probability at least $\frac{1}{2} + \frac{1}{2\sqrt{k}}$. We refer to these edges as *good*.

Since there is at most $O(1/k)$-fraction of the hyperedges that are not 2τ-nice we know that at least $\frac{1}{4\sqrt{k}}$-fraction of the hyperedges are 2τ-nice and *good*. By Lemma 9.6.11, for each 2τ-nice and *good* hyperedge e there exist two vertices $v_i, v_j \in e$ such that $\pi^{e,v_i}(S_t(\boldsymbol{w}_i))$ and $\pi^{e,v_j}(S_t(\boldsymbol{w}_j))$ intersect. Then there is a $\frac{1}{t^2}$ probability that the labeling strategy we defined will weakly satisfy hyperedge e.

Overall this strategy is expected to weakly satisfy at least $\frac{1}{4\sqrt{k}}\frac{1}{t^2} = \Omega(\frac{1}{k^{61}})$ fraction of the hyperedges. This is a contradiction since $\mathcal{L}$ is not $\frac{2k^2}{2^{\gamma k}}$-weakly satisfiable. $\qquad\square$

9.7 Probabilistic Inequalities

In the discussion below we will make use of the following well-known inequalities.

Theorem 9.7.1. *(Hoeffding's Inequality) Let $x_1, \ldots, x_n$ be independent real random variables such that $x_i \in [a_i, b_i]$. Then the sum of these variables $S = \sum_{i=1}^{n} x_i$ satisfies*

$$\mathbf{Pr}[|S - \mathbf{E}[S]| \geqslant nt] \leqslant 2e^{-\frac{n^2 t^2}{\sum_{i=1}^{n}(b^{(i)} - a^{(i)})^2}}.$$

Theorem 9.7.2. *(Berry-Esseen Theorem) Let $x_1, x_2, \ldots, x_n$ be i.i.d. random unbiased $\{-1, 1\}$ variables. Also assume that $\sum_{i=1}^{n} c_i^2 = 1$ and $\max_i\{|c_i|\} \leqslant \alpha$. Let g denote a unit Gaussian variable $N(0, 1)$. Then for any $t \in \mathbb{R}$,*

$$\left| \mathbf{Pr}\left[\sum c_i x_i \leqslant t \right] - \mathbf{Pr}[g \leqslant t] \right| \leqslant \alpha.$$

Theorem 9.7.3. *(Chebyshev's Inequality) Let X be a random variable with expected value u and variance σ^2. Then for any real number $t > 0$,*

$$\mathbf{Pr}[|X - \mu| \geqslant t \cdot \sigma] \leqslant 1/t^2.$$

9.8 Proof of Lemma 9.3.3

Recall that each $y^{(i)}$ is generated by the following manner:

$$y^{(i)} = \begin{cases} x^{(i)} & \text{with probability } 1 - \gamma \\ \text{random bit} & \text{with probability } \gamma. \end{cases} \tag{9.10}$$

Let us define a random vector $z \in \{-1, 1\}^n$ based on y. For y generated, if $y^{(i)}$ is generated as a copy of $x^{(i)}$ in (9.10), then $z^{(i)} = 0$; if $y^{(i)}$ is generated as a random bit in (9.10), then $z^{(i)} = 1$. Let us write $S = \sum_{i=1}^{n} w^{(i)} y^{(i)}$. Our proof is based on two claims.

Claim 9.8.1. $\mathbf{Pr}[\sum_{i=1}^{n} |w^{(i)}|^2 z^{(i)} \geqslant \gamma/2] \geqslant 1 - 2e^{-\frac{\gamma^2}{2\tau^2}}$.

Claim 9.8.2. *For any $a' < b' \in \mathbb{R}$ and any fixing of $z^{(1)}, z^{(2)}, \ldots, z^{(n)}$, if $\sum_{i=1}^{n}(w^{(i)})^2 z^{(i)} = \sigma^2 > 0$, then $\mathbf{Pr}[S \in [a', b']] \leqslant \frac{2|b' - a'|}{\sigma} + \frac{2\tau}{\sigma}$.*

Given the above two claims are correct, define event V to be $\{\sum_{i=1}^{n}(w^{(i)})^2 z^{(i)} \geqslant \frac{\gamma}{2}\}$ and use $\mathbf{1}_{[a,b]}(x) : \mathbb{R} \to \{0,1\}$ to denote the indicator function of whether x falls into interval $[a,b]$.

$$\mathbf{Pr}[S \in [a,b]] = \mathbf{E}[\mathbf{1}_{[a,b]}(S)] = \mathbf{Pr}[V]\mathbf{E}[\mathbf{1}_{[a,b]}(S) \mid V] + \mathbf{Pr}[\neg V]\mathbf{E}[\mathbf{1}_{[a,b]}(S) \mid \neg V]$$

By Claim 9.8.1,

$$\mathbf{Pr}[\neg V]\mathbf{E}[\mathbf{1}_{[a,b]}(S) \mid \neg V] \leqslant \mathbf{Pr}[\neg V] \leqslant 2e^{-\frac{\gamma^2}{2\tau^2}}.$$

By Claim 10.4.1,

$$\mathbf{Pr}[V]\mathbf{E}[\mathbf{1}_{[a,b]}(S) \mid V] \leqslant \frac{4(b-a)}{\sqrt{\gamma}} + \frac{4\tau}{\sqrt{\gamma}}.$$

Overall,

$$\mathbf{Pr}[S \in [a,b]] \leqslant \frac{4(b-a)}{\sqrt{\gamma}} + \frac{4\tau}{\sqrt{\gamma}} + 2e^{-\frac{\gamma^2}{2\tau^2}}.$$

It remains to verify Claim 9.8.1 and Claim 10.4.1.

To prove Claim 9.8.1, we need to apply the Hoeffding's inequality (see Theorem 9.7.1). Notice that $(w^{(i)})^2 z^{(i)} \in [0,(w^{(i)})^2]$ and applying Hoeffding's Inequality, we know

$$\mathbf{Pr}\left[\left|\sum_{i=1}^{n}(w^{(i)})^2 z^{(i)} - \mathbf{E}\left[\sum_{i=1}^{n}(w^{(i)})^2 z^{(i)}\right]\right| \geqslant nt\right] \leqslant 2e^{\frac{-2n^2 t^2}{\sum_{i=1}^{n}(w^{(i)})^4}}.$$

We know $\mathbf{E}[\sum_{i=1}^{n}(w^{(i)})^2 z^{(i)}] = \gamma$ and $\sum_{i=1}^{n}((w^{(i)})^2)^2 \leqslant \max_i\{(w^{(i)})^2\}\sum_{i=1}^{n}(w^{(i)})^2 \leqslant \tau^2$. If we take $nt = \gamma/2$, we have

$$\mathbf{Pr}\left[\left|\sum_{i=1}^{n}(w^{(i)})^2 z^{(i)} - \gamma\right| \geqslant \frac{\gamma}{2}\right] \leqslant 2e^{-\frac{\gamma^2}{2\tau^2}}.$$

Therefore, with probability at least $1 - 2e^{-\frac{\gamma^2}{2\tau^2}}$, $\sum_{i=1}^{n}(w^{(i)})^2 z^{(i)} \geqslant \frac{\gamma}{2}$.

To prove Claim 10.4.1, we need use Berry-Esseen Theorem (See Theorem 9.7.2). Let us split S into two parts: $S' = \sum_{z_i=1} w_i y_i$ and $S'' = \sum_{z_i=0} w_i y_i$. Since $S = S' + S''$ and S' is independent of S'', it suffices to show that $\mathbf{Pr}\left[S' \in [a',b']\right] \leqslant \frac{2|b'-a'|}{\sqrt{\sigma}} + \frac{2\tau}{\sigma}$ for any $a',b' \in \mathbb{R}$. Define $y'^{(i)} = 2y^{(i)} - 1$ and note that $y'^{(i)}$ a $\{-1,1\}$ variable. By rewriting S' using this definition, we have

$$S' = \sum_{z^{(i)}=1} w^{(i)} y^{(i)} = \sum_{z^{(i)}=1} w^{(i)}\frac{1 + y'^{(i)}}{2}.$$

Then

$$\mathbf{Pr}\left[S' \in [a',b']\right] = \mathbf{Pr}\left[\sum_{z^{(i)}=1} w^{(i)} y'^{(i)} \in [a'',b'']\right], \tag{9.11}$$

196

where $a'' = 2a' - \sum_{z^{(i)}=1} w^{(i)}$ and $b'' = 2b' - \sum_{z^{(i)}=1} w^{(i)}$. We can further rewrite the above term as

$$\mathbf{Pr}\left[\sum_{z^{(i)}=1} w^{(i)} y'^{(i)} \leqslant b''\right] - \mathbf{Pr}\left[\sum_{z^{(i)}=1} w^{(i)} y'^{(i)} \leqslant a''\right]$$

$$= \mathbf{Pr}\left[\sum_{z^{(i)}=1} \frac{w^{(i)} y'^{(i)}}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}} \leqslant \frac{b''}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}}\right] - \mathbf{Pr}\left[\sum_{z^{(i)}=1} \frac{w^{(i)} y'^{(i)}}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}} \leqslant \frac{a''}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}}\right].$$

We can now apply Berry-Esseen's theorem. Notice that for all the i such that $z^{(i)} = 1$, $y'^{(i)}$ is distributed as an independent unbiased random $\{-1, 1\}$ variable. Also $\max_{z^{(i)}=1} \frac{|w^{(i)}|}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}} \leqslant \frac{\tau}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}}$.

By Berry-Esseen's theorem, we know that expression (9.11) is bounded by

$$\mathbf{Pr}\left[N(0,1) \leqslant \frac{b''}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}}\right] - \mathbf{Pr}\left[N(0,1) \leqslant \frac{a''}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}}\right] + \frac{2\tau}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}}.$$

Using the fact that a unit Gaussian variable falls in any interval of length λ with probability at most λ and noticing that $b'' - a'' = 2(b' - a')$, we can bound the above quantity by

$$\frac{2|b' - a'|}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}} + \frac{2\tau}{\sqrt{\sum_{z^{(i)}=1}(w^{(i)})^2}} = \frac{2|b - a|}{\sigma} + \frac{2\tau}{\sigma}.$$

9.9 Proof of Invariance Principle (Theorem 9.3.10)

We restate our version of the invariance principle here for convenience.

Theorem 9.3.10 restated (Invariance Principle) Let $\mathscr{A} = \{\boldsymbol{A}^{\{1\}}, \ldots, \boldsymbol{A}^{\{R\}}\}, \mathscr{B} = \{\boldsymbol{B}^{\{1\}}, \ldots, \boldsymbol{B}^{\{R\}}\}$ be families of ensembles of random variables with $\boldsymbol{A}^{\{i\}} = \{a_1^{(i)}, \ldots, a_{k_i}^{(i)}\}$ and $\boldsymbol{B}^{\{i\}} = \{b_1^{(i)}, \ldots, b_{k_i}^{(i)}\}$, satisfying the following properties:

- For each $i \in [R]$, the random variables in ensembles $(\boldsymbol{A}^{\{i\}}, \boldsymbol{B}^{\{i\}})$ have matching moments up to degree 3. Further all the random variables in $\mathscr{A}$ and $\mathscr{B}$ are bounded by 1.

- The ensembles $\boldsymbol{A}^{\{i\}}$ are all independent of each other, similarly the ensembles $\boldsymbol{B}^{\{i\}}$ are independent of each other.

Given a set of vectors $\boldsymbol{l} = \{\boldsymbol{l}^{\{1\}}, \ldots, \boldsymbol{l}^{\{R\}}\}(\boldsymbol{l}^{\{i\}} \in \mathbb{R}^{k_i})$, define the linear function $\boldsymbol{l} : \mathbb{R}^{k_1} \times \cdots \times \mathbb{R}^{k_R} \to \mathbb{R}$ as

$$\boldsymbol{l}(\boldsymbol{x}) = \sum_{i \in [R]} \langle \boldsymbol{l}^{\{i\}}, \boldsymbol{x}^{\{i\}} \rangle$$

197

Then for a *B*-nice function $\Psi : \mathbb{R} \to \mathbb{R}$ we have

$$\left| \underset{\mathscr{A}}{\mathbf{E}} \left[\Psi \big(\boldsymbol{l}(\mathscr{A}) - \theta \big) \right] - \underset{\mathscr{B}}{\mathbf{E}} \left[\Psi \big(\boldsymbol{l}(\mathscr{B}) - \theta \big) \right] \right| \leqslant B \sum_{i \in [R]} \| \boldsymbol{l}^{\{i\}} \|_1^4. \qquad (9.12)$$

for all $\theta > 0$. Further, define the spread function $c(\alpha)$ corresponding to the ensembles $\mathscr{A}, \mathscr{B}$ and the linear function $\boldsymbol{l}$ as follows,

(**Spread Function:**)For $1/2 > \alpha > 0$, let

$$c(\alpha) = \max \left(\sup_\theta \mathbf{Pr}_\mathscr{A} \left[\boldsymbol{l}(\mathscr{A}) \in [\theta - \alpha, \theta + \alpha] \right], \quad \sup_\theta \mathbf{Pr}_\mathscr{B} \left[\boldsymbol{l}(\mathscr{B}) \in [\theta - \alpha, \theta + \alpha] \right] \right)$$

then for all $\tilde{\theta}$,

$$\left| \underset{\mathscr{A}}{\mathbf{E}} \left[\mathrm{sgn} \big(\boldsymbol{l}(\mathscr{A}) - \tilde{\theta} \big) \right] - \underset{\mathscr{B}}{\mathbf{E}} \left[\mathrm{sgn} \big(\boldsymbol{l}(\mathscr{B}) - \tilde{\theta} \big) \right] \right| \leqslant O\left(\tfrac{1}{\alpha^4} \right) \sum_{i \in [R]} \| \boldsymbol{l}^{\{i\}} \|_1^4 + 2c(\alpha). \qquad (9.13)$$

Proof. Let us prove equation (9.12) first. Let $\mathscr{X}_i = \{ \boldsymbol{B}^{\{1\}}, \dots, \boldsymbol{B}^{\{i-1\}}, \boldsymbol{B}^{\{i\}}, \boldsymbol{A}^{\{i+1\}}, \dots, \boldsymbol{A}^{\{R\}} \}$.
We know that

$$\underset{\mathscr{A}}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{A}) - \theta)] - \underset{\mathscr{B}}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{B}) - \theta)] = \underset{\mathscr{X}_0}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_R) - \theta)] - \underset{\mathscr{X}_R}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_0) - \theta)]$$

$$= \sum_{i=1}^R \underset{\mathscr{X}_{i-1}}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_{i-1}) - \theta)] - \underset{\mathscr{X}_i}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_i) - \theta)].$$

Therefore, it suffices to prove

$$\left| \underset{\mathscr{X}_{i-1}}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_{i-1}) - \theta)] - \underset{\mathscr{X}_i}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_i) - \theta)] \right| \leqslant B \| \boldsymbol{l}^{\{i\}} \|_1^4.$$

Let $\mathscr{Y}_i = \{ \boldsymbol{B}^{\{1\}}, \dots, \boldsymbol{B}^{\{i-1\}}, \boldsymbol{A}^{\{i+1\}}, \dots, \boldsymbol{A}^{\{R\}} \}$ and we have $\mathscr{X}_i = \{ \mathscr{Y}_i, \boldsymbol{B}^{\{i\}} \}$ and $\mathscr{X}_{i-1} = \{ \mathscr{Y}_i, \boldsymbol{A}^{\{i\}} \}$.
Then

$$\underset{\mathscr{X}_{i-1}}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_{i-1}) - \theta)] - \underset{\mathscr{X}_i}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_i) - \theta)] = \underset{\mathscr{Y}_i}{\mathbf{E}} \left[\underset{\boldsymbol{A}^{\{i\}}}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_{i-1}) - \theta)] - \underset{\boldsymbol{B}^{\{i\}}}{\mathbf{E}}[\Psi(\boldsymbol{l}(\mathscr{X}_i) - \theta)] \right]. \quad (9.14)$$

Notice that

$$\boldsymbol{l}(\mathscr{X}_{i-1}) - \theta = \langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}} \rangle + \sum_{1 \leqslant j \leqslant i-1} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{B}^{\{j\}} \rangle + \sum_{i+1 \leqslant j \leqslant R} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{A}^{\{j\}} \rangle - \theta$$

and

$$\boldsymbol{l}(\mathscr{X}_i) - \theta = \langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}} \rangle + \sum_{1 \leqslant j \leqslant i-1} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{B}^{\{j\}} \rangle + \sum_{i+1 \leqslant j \leqslant R} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{A}^{\{j\}} \rangle - \theta.$$

Take $\theta' = \sum_{1 \leqslant j \leqslant i-1} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{B}^{\{j\}} \rangle + \sum_{i+1 \leqslant j \leqslant R} \langle \boldsymbol{l}^{\{j\}}, \boldsymbol{A}^{\{j\}} \rangle - \theta$, We can further rewrite equation (9.14) as

$$\underset{\mathscr{Y}_i}{\mathbf{E}} \left[\underset{\boldsymbol{A}^{\{i\}}}{\mathbf{E}}[\Psi(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}} \rangle + \theta')] - \underset{\boldsymbol{B}^{\{i\}}}{\mathbf{E}}[\Psi(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}} \rangle + \theta')] \right]. \qquad (9.15)$$

198

Using the Taylor expansion of Ψ, we have that the inner expectation of equation (9.15) is equal to

$$\Big|\, \mathop{\mathbf{E}}_{\boldsymbol{A}^{\{i\}}}[\Psi(\theta') + \Psi'(\theta')\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}}\rangle + \frac{\Psi''(\theta')}{2}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}}\rangle)^2 + \frac{\Psi'''(\theta')}{6}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}}\rangle)^3 + \frac{\Psi''''(\delta_1)}{24}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}}\rangle)^4]$$

$$- \mathop{\mathbf{E}}_{\boldsymbol{B}^{\{i\}}}[\Psi(\theta') + \Psi'(\theta')\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}}\rangle + \frac{\Psi''(\theta')}{2}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}}\rangle)^2 + \frac{\Psi'''(\theta')}{6}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}}\rangle)^3 + \frac{\Psi''''(\delta_2)}{24}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}}\rangle)^4]\,\Big|.$$

$$(9.16)$$

Using the fact that $\boldsymbol{A}^{\{i\}}$ and $\boldsymbol{B}^{\{i\}}$ have matching moments up to degree 3, we can upper bound equation (9.16) by

$$\mathop{\mathbf{E}}_{\boldsymbol{A}^{\{i\}}}[\frac{\Psi''''(\delta_1)}{24}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}}\rangle)^4] - \mathop{\mathbf{E}}_{\boldsymbol{B}^{\{i\}}}[\frac{\Psi''''(\delta_2)}{24}(\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}}\rangle)^4] \leqslant \frac{B}{12}|\boldsymbol{l}^{\{i\}}|_1^4.$$

In the last inequality, we use the fact that Ψ is B-nice and $\langle \boldsymbol{l}^{\{i\}}, \boldsymbol{A}^{\{i\}}\rangle \leqslant \|\boldsymbol{l}^{\{i\}}\|_1, \langle \boldsymbol{l}^{\{i\}}, \boldsymbol{B}^{\{i\}}\rangle \leqslant \|\boldsymbol{l}^{\{i\}}\|_1$.

Overall, we bound the inner expectation of equation (9.15) by $\frac{B}{12}\|\boldsymbol{l}^{\{i\}}\|_1^4$. This implies equation (9.15) and therefore equation (9.9) is bounded by $\frac{B}{12}\|\boldsymbol{l}^{\{i\}}\|_1^4$, establishing equation (9.12).

To prove equation (9.13), we need to use the following lemma.

Lemma 9.9.1. *([117], Lemma 3.21) There exists some constant C such that $\forall 0 < \lambda < \frac{1}{2}$, there exists $\frac{C}{\lambda^4}$-nice function $\Phi_\lambda : \mathbb{R} \to [0,1]$ which approximates the $\mathrm{sgn}(x)$ function in the following sense: $\Phi_\lambda(t) = 1$ for all $t > \lambda$; $\Phi_\lambda(t) = 0$ for $t < -\lambda$.*

By the above lemma, we can find a $\frac{C}{\alpha^4}$-nice function Φ_α such that $\Phi_\alpha(\boldsymbol{l}(\mathscr{A}) - \theta)$ is equal to $\mathrm{sgn}(\boldsymbol{l}(\mathscr{A}) - \theta)$ except when $\boldsymbol{l}(\mathscr{A}) \in [\theta - \alpha, \theta + \alpha]$ and $\Phi_\alpha(\boldsymbol{l}(\mathscr{B}) - \theta)$ is equal to $\mathrm{sgn}(\boldsymbol{l}(\mathscr{B}) - \theta)$ except when $\boldsymbol{l}(\mathscr{B}) \in [\theta - \alpha, \theta + \alpha]$. Also for any $x \in R$, $|\mathrm{sgn}(x) - \Phi_\alpha(x)| \leqslant 1$ as $\mathrm{sgn}(x)$ and $\Phi_\alpha(x)$ are both in $[0,1]$.

Overall, we have

$$\Big|\mathop{\mathbf{E}}_{\mathscr{A}}\Big[\mathrm{sgn}\Big(\boldsymbol{l}(\mathscr{A}) - \theta\Big)\Big] - \mathop{\mathbf{E}}_{\mathscr{B}}\Big[\mathrm{sgn}\Big(\boldsymbol{l}(\mathscr{B}) - \theta\Big)\Big]\Big| \leqslant \Big|\mathop{\mathbf{E}}_{\mathscr{A}}\Big[\mathrm{sgn}\Big(\boldsymbol{l}(\mathscr{A}) - \theta\Big)\Big] - \mathop{\mathbf{E}}_{\mathscr{A}}\Big[\Phi_\alpha\Big(\boldsymbol{l}(\mathscr{A}) - \theta\Big)\Big]\Big|$$

$$+ \Big|\mathop{\mathbf{E}}_{\mathscr{A}}\Big[\Phi_\alpha\Big(\boldsymbol{l}(\mathscr{A}) - \theta\Big)\Big] - \mathop{\mathbf{E}}_{\mathscr{B}}\Big[\Phi_\alpha\Big(\boldsymbol{l}(\mathscr{B}) - \theta\Big)\Big]\Big| + \Big|\mathop{\mathbf{E}}_{\mathscr{B}}\Big[\Phi_\alpha\Big(\boldsymbol{l}(\mathscr{B}) - \theta\Big)\Big] - \mathop{\mathbf{E}}_{\mathscr{B}}\Big[\mathrm{sgn}\Big(\boldsymbol{l}(\mathscr{B}) - \theta\Big)\Big]\Big|$$

$$\leqslant \frac{C}{\alpha^4}\sum_{i \in [R]}\|\boldsymbol{l}^{\{i\}}\|_1^4 + 2c(\alpha).$$

$\square$

9.10 Hardness of Smooth k-LABEL-COVER

First we state the bipartite smooth Label Cover given by Khot [95]. Our reduction is similar to the one in [61] but in addition requires proving the smoothness property.

Definition 9.10.1. *A Label Cover problem $\mathscr{L}(G(V,W,E),N,M,\{\pi^{w,v}|(w,v)\in E\})$ consists of a bipartite graph $G(V,W,E)$ with bipartition V and W, projection functions $\pi^{w,v} : [M] \to [N]$ associated with each edge $(w,v) \in E$. We will only consider instances where all vertices in W have the same degree. For any labeling $L : V \to [M]$ and $L : W \to [N]$, an edge is said to be satisfied if $\pi^{w,v}(L(v)) = L(w)$. We define $Opt(\mathscr{L})$ to be the maximum fraction of edges satisfied by any labeling.*

Theorem 9.10.2. *There is an constant $\gamma > 0$ such that for all integer parameters u and J, it is NP-hard to distinguish the following two cases: A Label Cover problem $\mathscr{L}(G(V,W,E),N,M,\{\pi^{w,v}|(w,$ $E\})$ with $M = 7^{(J+1)u}$ and $N = 2^u 7^{Ju}$ having*

- $Opt(\mathscr{L}) = 1$ *or*
- $Opt(\mathscr{L}) \leqslant 2^{-2\gamma u}$.

In addition, the Label Cover has the following properties:

- *for each $\pi^{w,v}$ and any $i \in [N]$, we have $|(\pi^{w,v})^{-1}(i)| \leqslant 4^u$;*
- *for a fixed vertex w and a randomly picked neighbor of w called v,*

$$\forall i,j \in [M], \mathbf{Pr}[\pi^{w,v}(i) = \pi^{w,v}(j)] \leqslant 1/J.$$

Now we are ready to prove Theorem 9.6.1.

Proof. Given an instance of bipartite Label Cover $\mathscr{L}(G(V,W,E),N,M,\{\pi^{w,v}|(w,v)\in E\})$, we can convert it to a smooth k-LABEL-COVER instance $\mathscr{L}'$ as follows. The vertex set of $\mathscr{L}'$ is V and we generate the hyperedge set E' and projections associated with the hyperedges in the following way:

1. pick a vertex $w \in W$;
2. pick all k-tuple of v's neighbors $v_1,\ldots,v_k$ and add them as an hyperedge e to E';
3. for each $v_i \in e$, define $\pi^{e,v_i} = \pi^{w,v_i}$.

Completeness: If $Opt(\mathscr{L}) = 1$, then there exists a labeling L such that for every edge $(w,v) \in E$, $\pi^{w,v}(L(v)) = L(w)$. We can simply take the restriction of labeling L on W for the smooth k-LABEL-COVER instance $\mathscr{L}'$. For any hyperedge $e = (v_1,v_2,\ldots,v_k)$ generated by $w \in W$, we know $\pi^{e,v_i}(L(v_i)) = L(w) = \pi^{e,v_j}(L(v_j))$ for any $i,j \in [k]$. Therefore, we know that there exists a labeling strongly satisfying all hyperedges in $\mathscr{L}'$.

Soundness: If $Opt(\mathscr{L}) \leqslant 2^{-2\gamma u}$, then we can weakly satisfy at most $2k^2 2^{-\gamma u}$-fraction of the hyperedges in $\mathscr{L}'$. This can be proved via contrapositive argument. Suppose there is a labeling strategy L (defined on V) for the smooth k-LABEL-COVER that weakly satisfies $\alpha \geqslant 2k^2 2^{-\gamma u}$ fraction of the hyperedges. Using the regularity of the graph in $\mathscr{L}'$, we know that if we randomly pick a vertex w and randomly pick two of its neighbors v_1, v_2 then

$$\mathbf{Pr}\left[\pi^{w,v_1}(L(v_1)) = \pi^{w,v_2}(L(v_2))\right] \geqslant \frac{\alpha}{\binom{k}{2}} \geqslant \frac{2\alpha}{k^2}.$$

By an averaging argument, for at least $\frac{\alpha}{k^2}$-fraction of the vertices $w \in W$, have the following property: for all the possible pairs of w's neighbors, at least $\frac{\alpha}{k^2}$-fraction have the

same labels in L. For every w with this property, by an averaging argument again, one of w's neighbors, say v_0, must have the same label with at least $\frac{\alpha}{k^2}$-fraction of w's other neighbors. We can simply assign w label $\pi^{e,v_0}(L(v_0))$. Using such a labeling strategy (only on vertices with the above property) we will satisfy at least $\frac{\alpha^2}{k^4} = 4 \cdot 2^{-2\gamma u}$-fraction the edges of $\mathscr{L}$, leading to a contradiction.

Smoothness of $\mathscr{L}'$:   For any given vertex v in $\mathscr{L}'$, we want so show that if we randomly pick an hyperedge e' containing v, then for the projection $\pi^{e,v}$ as defined in $\mathscr{L}'$,

$$\forall i, j \in [M], \mathbf{Pr}[\pi^{e',v}(i) = \pi^{e',v}(j)] \leqslant \frac{1}{J}.$$

To see this, notice that all vertices in W have the same degree; picking a projection $\pi^{e',v}$ using the above procedure is the same as randomly picking a neighbor w of v and using the projection $\pi^{w,v}$ defined in $\mathscr{L}$. Therefore,

$$\forall i, j \in [M], \mathbf{Pr}[\pi^{e',v}(i) = \pi^{e',v}(j) = \mathbf{Pr}[\pi^{w,v}(i) = \pi^{w,v}(j)] \leqslant \frac{1}{J}.$$

$\square$

Chapter 10

Hardness of Learning Low degree PTFs

10.1 Introduction

10.1.1 Motivation

The last few years have witnessed a surge of research interest and results in theoretical computer science on halfspaces and low-degree PTFs, see e.g. [42, 54, 55, 70, 85, 124, 132]. One reason for this interest is the central role played by low-degree PTFs (and halfspaces in particular) in both practical and theoretical aspects of *machine learning*, where many learning algorithms either implicitly or explicitly use low-degree PTFs as their hypotheses. More specifically, several widely used linear separator learning algorithms such as the Perceptron algorithm and the "maximum margin" algorithm at the heart of Support Vector Machines output halfspaces as their hypotheses. These and other halfspace-based learning methods are commonly augmented in practice with the "kernel trick," which makes it possible to efficiently run these algorithms over an expanded feature space and thus potentially learn from labeled data that is not linearly separable in $\mathbb{R}^n$. The "polynomial kernel" is a popular kernel to use in this way; when, as is usually the case, the degree parameter in the polynomial kernel is set to be a small constant, these algorithms output hypotheses that are equivalent to low-degree PTFs. Low-degree PTFs are also used as hypotheses in several important learning algorithms with a more complexity-theoretic flavor, such as the low-degree algorithm of Linial *et al.* [111] and its variants [81, 118], including some algorithms for distribution-specific agnostic learning [21, 42, 84, 108].

Given the importance of learning algorithms that construct low-degree PTF hypotheses, it is a natural goal to study the limitations of learning algorithms that work in this way. We study the problem of learning low degree PTFs under the agnostic learning model, or equivalently the PTF_d-MA problem.

10.1.2 Our Main results

Recall the definition of PTFs as follows:

Definition 10.1.1. *For positive integer d, we call a function $f(x)\colon \mathbb{R}^n \to \mathbb{R}$ degree d polynomial function if it is of the following polynomial expansion form:*

$$\sum_{multiset\ S\subseteq[n],|S|\leqslant d} c_S \prod_{i\in S} x_i.$$

A degree d polynomial threshold function is of the form $\text{sgn}(f(x))$ *where $f(x)$ is a degree d polynomial function.*

Our main results are the following two theorems. Our first result is obtained assuming the UGC..

Theorem 10.1.2. *Assuming the UGC, for any constant d, PTF_d-MA $(1-\epsilon, 1/2+\epsilon)$ is NP hard for any constant $\epsilon > 0$.*

Remark 10.1.3. *In fact, Our hardness results also hold for d being $o(\log\log n)$.*

Theorem 10.1.4. PTF_1-PTF_2-MA $(1-\epsilon, 1/2+\epsilon)$ *is* NP-*hard.*

Note that the parameters in these hardness result are essentially optimal since it is

trivial to find a hypothesis with agreement rate $\frac{1}{2}$ as we can randomly choose to output function that is always -1 or the function that is always 1.

Our results immediately implies the following hardness of agnostic learning results: i) Assuming the UGC, even there exists a good degree d PTF that is consistent with $1-\epsilon$ fraction the examples, there is no efficient proper agnostic learning algorithm that can output a degree d PTFs correctly labelling more than $\frac{1}{2}+\epsilon$ fraction of the examples; ii) Assuming $P \neq NP$, even there exists a good halfspace that is consistent with $1-\epsilon$ fraction the examples, there is no efficient agnostic learning algorithm that can find a degree 2 PTFs that correctly label more than $\frac{1}{2}+\epsilon$ fraction of the examples.

Admittedly, our results do not rule out the possibility of efficient learning algorithm when ϵ is sub-constant or unrestricted hypothesis may be used.

10.1.3 Overview of the Proof

Based on the idea of constructing Dictator Test for PTF, we now overview the idea to prove Theorem 10.1.2 and 10.1.4. In comparison with the Dictator Test constructed in Section 2.6.2, to prove Theorem 10.1.2 which address the hardness of proper learning degree d PTFs, the additional complication is to handle the cross terms (such as $x_u^i x_v^j$) in degree d PTFs. It is easy to see that for the Dictator Test $\mathcal{T}_1$ there exists a degree 3 polynomial: $f_e = (x_u^i - x_v^i)\sum(x_u^i)^2$ that would pass the test with high probability. However, $f_v = 0$ which gives no clue for deciding the label of v. The main innovation of our proof is to design a proper Dictator Test that let $f_e = x_u^i - (x_v^i)^d$ passes with high probability. More specifically, we modify the test $\mathcal{T}_1$ by setting $y = (a_1 h_1 + g_1^d + b\delta, a_2 h_2 + g_1^d + b\delta, \ldots, a_n h_n + g_n^d, g_1, \ldots, g_n)$ and check $\mathrm{sgn}(f_e(y)) = b$. A nice property of such a test is that it force f_e to have almost no weight on the cross terms. The complete proof of the Dictator Test as well as Theorem 10.1.2 appears in Section 10.2.

As for Theorem 10.1.4, a first observation is that the given test $\mathcal{T}_1$ already has soundness $3/4+\epsilon$ for degree 2 PTFs. To see this, notice that r and $-r$ is generated with equal probability, essentially we are testing the following 4-tuple of inequalities with equal probability:

$$f_e(r+\delta u) > 0;$$
$$f_e(r-\delta u) > 0;$$
$$f_e(-r+\delta u) < 0;$$
$$f_e(-r-\delta u) < 0.$$

Recall that $f_e(x)$ is a degree 2 polynomial, we can write it as the sum of $\theta + f_1(x) + f_2(x)$ where $f_1(x)$ is its linear (degree 1) part and $f_2(x)$ is the quadratic (degree 2) part.

If all of the above 4 inequalities hold, combining $f_e(t+\delta u) > 0$ and $f_e(-t-\delta u) < 0$, we get that $f_1(t+\delta u) > 0$; and combining $f_e(t-\delta u) < 0$ and $f_e(-t+\delta u) > 0$ we get $f_1(t-\delta u) < 0$. Therefore for some degree 2 polynomial function f, if it passes the test with probability $3/4+\epsilon$, then by an average argument, ϵ fraction of the 4-tuple inequalities all hold which implies that for ϵ fraction of the r generated, $f_1(r+\delta u) > 0$ and $f_1(r-\delta u) < 0$. Then we know

linear function f_1 pass the Dictator Test $\mathcal{T}_1$ with probability above $1/2 + \epsilon$. This essentially reduce to the problem of testing degree 1 PTF which we already know how to analyze.

To further get the soundness down to 1/2, more work has to be done. Roughly speaking, we check $\text{sgn}(f(k_1 r + k_2 \delta u)) = \text{sgn}(k_2)$ for k_1, k_2 generated from some carefully constructed distribution.

In addition to the above modification, in order to remove the need of assuming the UGC, we use the "folding trick" that is proposed in [60, 106] to ensure the consistency across different vertices. This has the benefit that we only need to design a test on one vertex (instead of an edge). The reason that we can not use "folding" for our first result on low degree PTFs is that such a folding can not handle the cross terms. The complete proof Theorem 10.1.4 appears in Section 10.3.

10.2 On Hardness of Proper Learning Degree d PTFs

In this section, we will prove Theorem 10.1.2.

10.2.1 Dictator Test

As is mentioned, a key gadget in the hardness reduction is a *Dictator Test* of whether a degree d polynomial threshold function $f : \mathbb{R}^{2n} \to \{-1, 1\}$ is of the form

$$\text{sgn}(x_i - x_{n+i}^d)$$

for some $i \in [n]$.

For any function

$$f(x) = \sum_{\text{multiset } S, |S| \leqslant d, S \subseteq [2n]} c_S \prod_{i \in S} x_i$$

where $x \in \mathbb{R}^{2n}$, our Dictator Test will query its value one a single point y and decide to accept or reject based on $\text{sgn}(f(y))$. For notation convenience, we refer the future appearance of S as *multiset* if not further clarified.

Following is the definition of the test.

Definition 10.2.1. *Fixing parameter* $\beta = \frac{1}{\log n}$ *and* $\delta = \frac{1}{2^{n^2}}$. *we generate one randomized query with the following procedures:*

Dictator Test $\mathcal{T}_d$

1. Generate independent β-biased bits $a_1, a_2, \ldots, a_n \in \{0, 1\}$ (i.e., $a_i = 1$ with probability β and 0 with probability $1 - \beta$).
2. Generate $2n$ independent unit Gaussian variables: $h_1, \ldots, h_n, g_1, \ldots, g_n$.
3. Generate a random bit $b \in \{-1, 1\}$.
4. Set $y = (a_1 h_1 + g_1^d + b\delta, a_2 h_2 + g_1^d + b\delta, \ldots, a_n h_n + g_n^d, g_1, \ldots, g_n)$.
5. Accept if $\text{sgn}(f(y)) = b$.

206

Now we state the completeness and soundness properties of $\mathcal{T}_d$.

Lemma 10.2.2. *(Completeness) When $f(x) = x_i - x_{n+i}^d$, it passes the test with probability $1 - \beta$.*

Proof. We know that

$$f(y) = a_i h_i + b\delta.$$

Therefore, when $a_i = 0$ (with probability $1 - \beta$), $f(y)$ has the same sign as b. $\qquad\square$

The more complicated part is the following soundness guarantee. To state it, we first introduce the following notion:

Definition 10.2.3. *For any degree d polynomial function $f : \mathbb{R}^n \to \mathbb{R}$, we define*

$$\mathrm{wt}(f) = \sum_{1 \leqslant |S| \leqslant d} |c_S|.$$

We also define $I_\theta(f)$ to be $\{i \mid i \in S, |c_S| \geqslant \theta \cdot \frac{\mathrm{wt}(f)}{\binom{n+d}{d}}\}$.

By above definition, when $\theta \leqslant 1$, $I_\theta(f)$ is not empty as the total number of multiset of size at most d is $\binom{n+d}{d}$.

Lemma 10.2.4. *(Soundness) For d being a constant and all n big enough, if some degree d polynomial function $f(x)$ passes the test with probability $\frac{1}{2} + \beta$, then for $f_1 = f(x_1, \ldots, x_n, 0, \ldots, 0)$, $f_2 = f(0, 0, \ldots, 0, x_{n+1}, \ldots, x_{2n})$, we have $|I_{0.5}(f_1)| \leqslant \frac{1}{\beta^2}, |I_1(f_2)| \leqslant \frac{1}{\beta^2}$. In addition, if for some $i \in [n]$, $(n + i) \in I_1(f_2)$, we must also have $i \in I_{0.5}(f_1)$.*

Proof. If $\mathrm{wt}(f) = 0$ which means $f(x)$ is a constant function, it passes the test with probability $\frac{1}{2}$.

Otherwise, as the Dictator Test only checks the sign of f at some point, with out lose of generality, we can assume that $\mathrm{wt}(f) = 1$.

Set $r = (a_1 h_1 + g_1^d, a_2 h_2 + g_1^d, \ldots, a_n h_n + g_n^d, g_1, g_2, \ldots, g_n)$ and $u = (1, 1, \ldots, 1, 0, 0, .0) \in \mathbb{R}^{2n}$ such that $u_i = 1$ when $1 \leqslant i \leqslant n$ and $u_i = 0$ when $(n + 1) \leqslant i \leqslant 2n$. Then the Dictator Test defined is essentially the following:

- Generate r;
- Test $f(r + \delta u) > 0$ with probability $\frac{1}{2}$ and $f(r - \delta u) < 0$ with probability $\frac{1}{2}$.

Suppose that some function $f(x)$ passes with probability $\frac{1}{2} + \beta$, then we know that for at least 2β fraction of the r generated, we must have both of the following hold:

$$f(r + \delta u) > 0; \tag{10.1}$$

$$f(r - \delta u) < 0. \tag{10.2}$$

Viewing d as a constant, let us first bound the difference between $f(r + \delta u)$ and $f(r)$:

$$f(z+\delta u) - f(z) = \sum_{|S|\leqslant d} c_S \left(\prod_{i\in S, i\in\{1,2,\ldots\}} (r_i+\delta) \prod_{i'\in S, i'\in\{n+1,n+2,\ldots,2n\}} r_{i'} - \prod_{i\in S} r_i \right)$$

$$\leqslant \sum_{1\leqslant|S|\leqslant d} |c_S| \cdot \sum_{T\neq\emptyset, T\subseteq(S\cap[n])} \delta^{|T|} \cdot \prod_{i\notin T} |r_i| \leqslant \sum_{1\leqslant|S|\leqslant d} |c_S| 2^{|S|} \left(\delta \prod_{|r_i|\geqslant 1, i\in S} |r_i| \right) \quad (10.3)$$

By the property of Gaussian random variables, we know that $\mathbf{E}[|r_i|] \leqslant E[|g_i|^d] + E[|h_i|] \leqslant d^d$. Then by Markov inequality, $\mathbf{Pr}[|r_i| \geqslant 2d^d n^2] \leqslant \frac{1}{2n^2}$. By union bound for all but $\frac{1}{n}$ fraction of the r generated, we have that $\max_i |r_i| \leqslant 2d^d n^2$.

Given that $\max_i |r_i| \leqslant 2d^d n^2$, we can further bound (10.3) by

$$\delta 2^d (2d^d n^2)^d \sum_{1\leqslant|S|\leqslant d} |c_S| \leqslant \delta 4^d d^{d^2} n^{2d} \leqslant \frac{1}{2^n}. \qquad \text{(for } n \text{ large enough and constant } d)$$

Similar calculation shows that when $\max_i |r_i| \leqslant 2d^d n^2$,

$$f(r) - f(r-\delta u) \leqslant \frac{1}{2^n}.$$

Therefore, for at least $2\beta - \frac{1}{n} \geqslant \frac{1}{\log n}$ fraction of the z generated, we have that

$$|f(r)| < \frac{1}{2^n}.$$

Recall that $f(r) = f(a_1 h_1 + g_1^d, \ldots, a_n h_n + g_n^d, g_1, \ldots, g_n)$; for every realization of $a \in \{0,1\}^n$, we denote the corresponding restriction on f as $f_a(g,h)$ which is a degree d^2 polynomial of on Gaussian random variables $h_1, \ldots, h_n, g_1, \ldots, g_n$. We use $\|f_a\|_2$ to denote $\mathbf{E}[f_a(g_1, \ldots, g_n, h_1, \ldots, h_n)^2]^{\frac{1}{2}}$

Then we know that

$$\frac{1}{\log n} \leqslant \mathbf{Pr}_{a,g,h}(|f_a(g,h)| \leqslant 1/2^n) \leqslant \mathbf{E}_a \left[\left(\frac{1/2^n}{\|f_a\|_2} \right)^{1/d^2} \right]. \qquad (10.4)$$

where the last inequality is due to the small ball property of Gaussian polynomials (see Lemma 10.4.1).

Suppose $a' = \arg\min_a \|f_a\|_2$. Then (10.4) implies that

$$\left(\frac{1}{2^n \cdot \|f_{a'}\|_2} \right)^{1/d^2} \geqslant \frac{1}{\log n}$$

or equivalently

$$\|f_{a'}\|_2 \leqslant \frac{(\log n)^{d^2}}{2^n}. \qquad (10.5)$$

Let us write down $f_{a'}$ as polynomial on $g_1, \ldots, g_n, h_1, \ldots, h_n$, say

$$f_{a'} = \sum_{\text{multiset } S,S} w_{T,T'} \prod_{i\in T} g_i \prod_{i\in T'} h_i,$$

Let us further simplify the notation of $w_{T,\phi}$ by w_T. Then we claim that every

$$w_T \leqslant \frac{1}{n^{10d}}.$$

Otherwise, by Lemma 10.4.2,

$$\|f_{a'}\|_2 \geqslant \frac{1}{n^{10d}} \cdot \frac{1}{\binom{2n+d^2}{d^2}(d^2)^{d^2}}.$$

which asymptotically violates (10.5) for large enough n.

Knowing that each of the w_T is small, let us now establish its relationship the original coefficient in c_S (multiset $S \subseteq [2n]$).

It is easy to see the restriction of $f_{a'}$ by setting all the h_i to 0 is the same as $f(g_1^d, \ldots, g_n^d, g_1, \ldots, g_n)$ which implies that:

$$\sum_{T \subseteq [n]} w_T \prod_{i \in T} g_i = \sum_{S \subseteq [2n]} c_S \prod_{i \in S, i \in [n]} g_i^d \prod_{n+i \in S} g_i.$$

Summing all the c_S such that S corresponding to the same term

$$\prod_{i \in S, i \in [n]} g_i^d \prod_{n+i \in S} g_i,$$

we get w_T for $T = \{i : d | i \in S\} \cup \{i | n + i \in T\}$.

Following lemma illustrates when different c_S can be corresponding to the same term.

Lemma 10.2.5. *For any multiset $S_0, S_1 \subseteq [2n]$ of size at most d and $S_0 \neq S_1$, if*

$$\prod_{i \in S_0, 1 \leqslant i \leqslant n} g_i^d \prod_{n+j \in S_0, 1 \leqslant j \leqslant n} g_j = \prod_{i \in S_1, 1 \leqslant i \leqslant n} g_i^d \prod_{n+j \in S_1, 1 \leqslant j \leqslant n} g_j, \qquad (10.6)$$

then there exists some i, such that $S_0 = \{i\}$ and $S_1 = \{n + i : d\}$ or vice versa.

Proof. (Proof of Lemma 10.2.5) Let us discuss the following two cases.

- $S_0 \cap [n] \neq S_1 \cap [n]$. Without loss of generality, let us assume that there is some $i \in S_0$ and $i \notin S_1$. Then to make (10.6) hold, it must be the case that S_1 contains d copy of $n + i$. Also since $|S_1| \leqslant d$, it can only be the case that $S_1 = \{n + i : d\}$. Therefore S_0 must be $\{i\}$.

- $S_0 \cap \{n + 1, \ldots, 2n\} \neq S_1 \cap \{n + 1, \ldots, 2n\}$. Without loss of generality, let us assume that there is some $(n + i) \in T_0$. Then it must be the case that $i \in T_1$ to make (10.6) hold. Then the power on g_i is d. This can only happen when $S_1 = \{n + i : d\}$ which enforce S_0 to be $\{i\}$.

$\square$

By Lemma 10.2.5, we have the relationship between c_S and w_T.

- For any $i \in [n]$ and $S_1 = \{i\}$, $S_2 = \{n + i : d\}$, $T = \{i : d\}$, $c_{S_1} + c_{S_2} = w_T$.

209

- If for any $i \in [n]$, $S \neq \{i\}$ and $S \neq \{n+i:d\}$, then for $T = \{i:d \mid i \in T\} \cup \{i \mid n+i \in T\}$, we have that

$$w_T = c_S.$$

Recall that

$$f_1(x) = f(x_1, \ldots, x_n, 0, \ldots, 0) = \sum_{S \subseteq [n]} c_S \prod_{i \in S} x_i$$

and

$$f_2(x) = f(0, 0, \ldots, 0, x_{n+1}, \ldots, x_{2n}) = \sum_{S \subseteq \{n+1, \ldots, 2n\}} c_S \prod_{i \in S} x_i.$$

Let us write $f(x) = f_1 + f_2 + f_{12}$ where

$$f_{12} = \sum_{|S| \leqslant d, S \cap \{1,2,\ldots,n\} \neq \emptyset, S \cap \{n+1,n+2,\ldots,2n\} \neq \emptyset} c_S \prod_{i \in S} x_i.$$

We know then

- every c_S appear in f_1 such that $|S| \geqslant 2$,

$$c_S \leqslant \frac{1}{n^{10d}}. \tag{10.7}$$

- for every c_S in f_2 and S is not the multiset $\{n+i:d\}$ for some $i \in [n]$,

$$c_S \leqslant \frac{1}{n^{10d}}. \tag{10.8}$$

- for every $i \in [n]$,

$$\big||c_{\{i\}}| - |c_{\{n+i:d\}}|\big| \leqslant |c_{\{i\}} + c_{\{n+i:d\}}| \leqslant \frac{1}{n^{10d}}. \tag{10.9}$$

- for every c_S appear in f_{12},

$$c_S \leqslant \frac{1}{n^{10d}}. \tag{10.10}$$

Since for f_1 and f_2, their coefficients are either matching (such as $c_{\{i\}}$ and $c_{\{n+i:d\}}$) or being small themselves, we have that

$$|\text{wt}(f_1) - \text{wt}(f_2)| \leqslant O\left(\frac{1}{n^{10d}} \cdot \binom{n+d}{d}\right) \leqslant \frac{1}{n}. \tag{10.11}$$

Also by (10.10) as every efficient in f_{12} is less than $\frac{1}{n^{10d}}$ and there are at most $\binom{2n+d}{d}$ of them, we know then

$$\text{wt}(f_{12}) \leqslant \frac{1}{n^{10d}}\binom{2n+d}{d} \leqslant \frac{1}{n}$$

Therefore, recall that $\text{wt}(f_1) + \text{wt}(f_2) + \text{wt}(f_{12}) = \text{wt}(f)$, we have

$$\text{wt}(f_1) + \text{wt}(f_2) \geqslant 1 - \frac{1}{n}. \tag{10.12}$$

Combining (10.11) and (10.12), we know that for $n \geqslant 10$,

$$0.51 \geqslant \mathrm{wt}(f_1), \mathrm{wt}(f_2) \geqslant 0.49.$$

Therefore, every element $(n+i)$ in $I_1(f_2)$, it must come from those sets S such that $c_S \geqslant 0.49/\binom{n+d}{d}$. By (10.8), we know it can only be the set $S = \{n+i : d\}$ as all the other c_S are less than $\frac{1}{n^{10d}}$. By (10.9), we know that $\hat{f}(\{i\}) \geqslant \frac{0.48}{\binom{n+d}{d}}$ and it must be in $I_{0.5}(f_1)$ as $\mathrm{wt}(f_1) \leqslant 0.51$.

By above proof, we also know $|I_1(f_2)| \leqslant |I_{0.5}(f_1)|$ as any $(n+i) \in I_1(f_2)$ implies that $i \in I_{0.5}(f_1)$. It remains to bound the size of $I_{0.5}(f_1)$ by $\frac{1}{\beta^2}$.

Let us prove it by contradiction. Suppose that $I_{0.5}(f_1) \geqslant \frac{1}{\beta^2}$. As $\mathrm{wt}(f_1) \geqslant 0.49$, every $j \in I_{0.5}(f_1)$ comes from the set $S = \{j\}$ as all the other c_S is less than $\frac{1}{n^{10d}}$. Then when we consider all the possible realization of a, with probability $1 - (1-\beta)^{|I(f_1)|} \geqslant 1 - \frac{1}{n}$, there exists some $i \in I_{0.5}(f_1)$ with $a_i = 1$. By the definition of $I_{0.5}(f_1)$, we also must have

$$c_{\{i\}} \geqslant \frac{0.5 \cdot 0.49}{\binom{n+d}{d}} \geqslant \frac{0.2}{\binom{n+d}{d}}.$$

Then there will be a term $c_{\{i\}} h_i$ in the expansion of f_a as a Gaussian polynomial of g and h such that $|c_{\{i\}}| \geqslant 0.2/\binom{n+d}{d}$. This suggests that for $(1 - \frac{1}{n})$ of the realization of a, $|f_a|_2 \geqslant \frac{0.2}{\binom{n+d}{d}} \cdot \frac{1}{\binom{2n+d^2}{d^2}(d^2)^{d^2}} \geqslant \Omega(\frac{1}{n^{2d^2}})$.

Then we have

$$\frac{1}{\log n} \leqslant \mathbf{Pr}_{a,g,h}(|f_a(g,h)| \leqslant 1/2^n) \leqslant \frac{1}{n} + O(\frac{n^{2d}}{2^n/d})$$

which leads to a contradiction for big enough n.

$\square$

10.2.2 Hardness Reduction from UNIQUE-GAMES

With above Dictator Test, we now prove Theorem 10.1.2. The hardness reduction is from a UNIQUE-GAMES Instance $\mathcal{L}(U, V, E, \Pi, k)$ to a distribution of positive and negative examples. The examples in the learning problem lies in the space $\mathbb{R}^{(|U|+|V|)k}$ labeled with either positive $(+1)$ or negative (-1). Denote $\dim = (|U|+|V|)k$. For $y \in \mathbb{R}^{\dim}$, each coordinate is indexed by a possible label for a vertex in $U \cup V$.

We fix the following notations: $w \in U \cup V$ and $x \in \mathbb{R}^{\dim}$, we use x_w^i to denote the coordinate corresponding to the vertex w's i-th label. Also we use x_w to indicate the collection of coordinates of coordinates corresponding to vertex w; i.e., $(x_w^1, x_w^2, \ldots, x_w^k)$. Also for any function $f(x) : \mathbb{R}^{\dim} \to \mathbb{R}$, we use f_u to denote f's restriction by setting all the coordinate to be 0 except x_u. Similarly, denote $f_{u,v}$ as the restriction of f by setting all the coordinate to be 0 except x_u, x_v.

We construct the example distribution from the UNIQUE-GAMES instance by the following procedures. Let us choose parameter $\beta = \frac{1}{\log(k)}$ and $\delta = 2^{-k^2}$.

Reduction from UNIQUE-GAMES

1. Randomly choose an edge (u,v) for $u \in U$ and $v \in V$.

2. Setting $y_w = 0$ for any $w \in U \cup V$ such that $w \neq u, w \neq v$.

3. Generate independent β-biased bits $a_1, a_2, \ldots, a_k \in \{0,1\}$ (i.e., $a_i = 1$ with probability β and 0 with probability $(1 - \beta)$) and generate $2k$ Gaussian $h_1, \ldots, h_k, g_1, \ldots, g_k$.

4. Generate a random bit $b \in \{-1, 1\}$.

5. For every $i \in [k]$, set $y_v^{(i)} = g_i$

6. For every $i \in [k]$, set $y_u^{(i)} = a_1 h_1 + (g_{\pi^e(i)})^d + \delta b$.

7. Output example-label pair (y, b).

We will prove the following two Lemmas (10.2.6 and 10.2.7) for the reduction.

Lemma 10.2.6. *(Completeness) If* $\mathrm{Opt}(\mathscr{L}) = 1 - \eta$, *then there is a degree d polynomial threshold function that is consistent with $1 - \eta - \beta$ percentage of the examples.*

Proof. (Proof of Lemma 10.2.6) Suppose that there is a labeling l that satisfies $1 - \eta$ edges. Then consider the following degree d polynomial threshold functions:

$$\mathrm{sgn}\Big(\sum_{u \in U} x_u^{(l(u))} - \sum_{v \in V} (x_v^{(l(v))})^d \Big).$$

It is easy to verify that such a PTF agrees with $1 - \eta - \beta$ fraction of the examples. $\qquad \square$

Lemma 10.2.7. *(Soundness) If* $\mathrm{Opt}(\mathscr{L}) \leqslant 1/k^{\Theta(\eta)}$, *then there is no degree d polynomial threshold function agrees with more than $\frac{1}{2} + 2\beta$ fraction of the examples.*

Proof. (Proof of Lemma 10.2.7) We prove above lemma by contradiction. Suppose that there is some degree d polynomial function f that passes the test with probability $\frac{1}{2} + 2\beta$. Then by an average argument, for β fraction of the edge (u,v) picked in the first step, we have that $f(x)$ passes the test with probability $\frac{1}{2} + \beta$. Let us call these edges "good".

For a particular "good" edge $e = (u,v)$, let us assume that π_e is the identity mapping for notation convenience.

Essentially, we are conducting our test for $\mathscr{T}_d$ for $f_{u,v}$ with parameter $n = k$.

Since $f_{u,v}$ passes the test with probability $\frac{1}{2} + \beta$, By Lemma 10.2.7, we must have that $I_{0.5}(f_u) \subseteq I_1(f_v) \neq \emptyset$ (if we index x_u^i by i when output $I_{0.5}(f_u)$ and index x_v^i by i when output $I_{0.5}(f_v)$). In addition, we have that $|I_1(f_v)|, |I_{0.5}(f_u)| \leqslant 1/\beta^2$.

We now give the following labelling strategy based on f. For every $u \in U$, we randomly pick its label from $I_{0.5}(f_u)$ and for every $v \in V$, we randomly pick its label from $I_1(f_v)$. Then for each good edge, it will get satisfied by probability β^2. Overall such a labelling strategy gives a labelling that satisfies at least $\beta^3 = \frac{1}{(\log k)^3}$ fraction of the edges in expectation. This is a contradiction to the fact that $\mathrm{Opt}(\mathscr{L}) \leqslant 1/k^{\Theta(\eta)}$ for sufficiently large k. $\qquad \square$

By above proof, we gave a way of constructing a distribution $\mathscr{D}$ of example-label pairs from a instance of Label Cover L. Let us use $\mathrm{Opt}(\mathscr{D})$ to denote the accuracy of the best degree d PTF on $\mathscr{D}$. And our constructed distribution has the following properties:

- If $\mathrm{Opt}(\mathscr{L}) = 1 - \eta$, then $\mathrm{Opt}(\mathscr{D}) = 1 - \eta - \frac{1}{\log k}$.
- If $\mathrm{Opt}(\mathscr{L}) \leqslant 1/k^{\theta(\eta)}$, then $\mathrm{Opt}(\mathscr{D}) \leqslant \frac{1}{2} + \frac{2}{\log k}$.

10.2.3   Discretizing the Gaussian Distribution

Above reduction is not in polynomial time as the resulting distribution $\mathscr{D}$ has infinite support. If we look into the construction, for every edge picked we need to generate $2k$ independent Gaussian $h = (h_1, \ldots, h_k), g = (g_1, \ldots g_k)$.

To "discretize" the reduction, we will replace each h and g by some h' and g' where each h'_i and g'_i are independently generated by sum of N bits divided $\sqrt{N}$ where $N = (2k)^{24(d^2)^2}$.

By Theorem 10.6.2, there exists a way of coupling (g, h) with (g', h') such that for every degree d^2 polynomial, it has the same sign on (g, h) as on (g', h') except for $1/k$ fraction of the (g, h) generated. Therefore, if we replace (g, h) with (g', h') in the reduction and also notice that for every realization of a, the resulting polynomial on $g_1, \ldots, g_k$ and $h_1, h_2, \ldots, h_k$ is of degree at most d^2, our discretized reduction will almost preserve the soundness and completeness guarantees with a loss of $\frac{1}{k}$:

- If $\mathrm{Opt}(\mathscr{L}) = 1 - \eta$, then $\mathrm{Opt}(\mathscr{D}) = 1 - \eta - \frac{1}{\log k} - 1/k$.
- If $\mathrm{Opt}(\mathscr{L}) \leqslant 1/k^{\theta(\eta)}$, then $\mathrm{Opt}(\mathscr{D}) \leqslant \frac{1}{2} + \frac{2}{\log k} + 1/k$.

Also notice that the distribution of (g', h') has a support of size $2^{2kN} = 2^{2k(2k)^{24(d^2)^2}}$ which is constant as the label size is regarded as constant for UNIQUE-GAMES. Then we can simply enumerate all its support to further remove the need of random bits and make the reduction deterministic.

Eventually, by picking proper η and k (e.g., $\eta = \epsilon/2$ and $k = e^{1/\epsilon^2}$), we prove Theorem 10.1.2.

10.2.4   For d being Super-constant

From above proof, our hardness result hold for any constant d. Actually, it is easy to see that $d = o(\log \log n)$, our proof will still work. (we need $2^{2k(2k)^{24(d^2)^2}}$ to be some polynomial on the size of the label cover.)

10.3   Hardness of Learning Halfspaces with degree 2 PTFs

In this section, we prove Theorem 10.1.4. Again the proof has two parts. In the first step, we construct a dictator test for degree 2 PTFs. In the second step, we compose such a dictator test with the Label Cover problem to prove NP-hardness result.

10.3.1 The Dictator Test

The key gadget in the hardness reduction is a *Dictator Test* of whether a degree 2 polynomial threshold function $f : \mathbb{R}^n \to \{-1, 1\}$ is of the form $\text{sgn}(x_i)$ for some $i \in [n]$.

Suppose p is a degree 2 polynomial function written as the following form:

$$p(x) = \theta + \sum_{i \in [n]} c_i x_i + \sum_{i,j \in [n], i \leqslant j} c_{ij} x_i x_j.$$

We also write $p_1(x) = \sum c_i x_i$ and $p_2(x) = \sum c_{ij} x_{ij}$ to denote the degree 1 and 2 part of $p(x)$.

Below is a one query Dictator Test $\mathcal{T}_1$ on $\text{sgn}(p(x))$. We choose parameter $\beta = \frac{1}{\log(n)}$ and $\delta = \frac{1}{2^n}$ for the test.

Test $\mathcal{T}_2$

1. Generate independent β-biased bits $a_1, a_2, \ldots, a_n \in \{0, 1\}$ (i.e., $a_i = 1$ with probability ϵ and 0 with probability $(1 - \beta)$ and generate n independent Gaussian variables $g_1, \ldots, g_n$. Set $r = (a_1 g_1, a_2 g_2, \ldots, a_n g_n)$.
2. Generate t by randomly pick a number $i \in \{1, 2, \ldots, (\log n)^2\}$ and set $t = n^i$.
3. Generate random bit $b \in \{-1, 1\}$.item
4. Set $u \in \mathbb{R}^n$ to be the all "1" vector $(1, 1, 1, \ldots, 1)$ and set $y = t^3 r + bt^2 \delta u$.
5. Accept if $\text{sgn}(p(y)) = b$.

For above test $\mathcal{T}_1$, We have the following completeness and soundness properties.

Lemma 10.3.1. *(Completeness) If $p(x) = x_i$ for $i = 1 \ldots, n$, then it passes with probability at least $1 - \beta$.*

Proof. If $p(x) = x_i (i \in [n])$, then as long as a_i is set to zero in step 1, $p(x) = b_2 \delta t^2$ and it passes the test. By definition of the test, this happens with probability $1 - \beta$. $\qquad\square$

Lemma 10.3.2. *(Soundness) Denote $A = \sum c_i$ and $I(p)$ to be the set $\{i \mid c_i > A/n^2\}$. If some p passes the test with probability $\frac{1}{2} + \beta$, then $|I(p)| \leqslant 1/\beta^2$ and $A > 0$.*

Proof. The proof is by contradiction. Suppose for some function p with $|I(p)| \geqslant 1/\beta^2$ or $A \leqslant 0$ passes above defined test with probability $\frac{1}{2} + \beta$.

First we show is the following lemma.

Lemma 10.3.3. $\mathbf{Pr}[p_1(r) \in (-\delta A, \delta A)] \leqslant \frac{2}{n}.$

Proof. It is obvious when $A \leqslant 0$ above inequality hold. Otherwise, assuming $A > 0$ and $|I(p)| \geqslant 1/\beta^2$. We know that in step 1 when generating a_i, with probability $1 - (1-\beta)^{|I(p)|} \geqslant 1 - \frac{1}{n}$ at least one of the coordinate in $I(p)$ is set to a Gaussian (instead of zero). For these $(1 - \frac{1}{n})$ fraction of x, we know that no matter which other coordinate is set to be Gaussian, $p_1(r)$ is a Gaussian variable with variance at least A^2/n^4 (as one of the weight is at least A/n^2). Using the anti-concentration of Gaussian variable (Lemma 10.4.1), we have that

$$\mathbf{Pr}_V[p_1(r) \in (-\delta A, \delta A)] \leqslant \frac{2\delta A}{A/n^2} \leqslant \frac{n^3}{2^n} \leqslant \frac{1}{n}.$$

By union bound, we know that for at most $\frac{2}{n}$ of the x, $p(r)$ is inside the interval $(-\delta A, \delta A)$. $\qquad\square$

Notice that r and $-r$ are generated with equal probability, essentially a equivalent test to $\mathcal{T}_2$ would be the testing the following 4 inequalities with equal probability for r, t generated.

$$p(t^3 r + t^2 \delta \alpha) > 0 \tag{10.13}$$
$$p(t^3 r - t^2 \delta \alpha) < 0 \tag{10.14}$$
$$p(-t^3 r + t^2 \delta \alpha) > 0 \tag{10.15}$$
$$p(-t^3 r - t^2 \delta \alpha) < 0. \tag{10.16}$$

As $p(y)$ passes the test with probability $\frac{1}{2} + \beta$, using an averaging argument, for $\beta/2$ fraction of the r, $\frac{1}{2} + \beta/2$ fraction of the constraints containing the r are satisfied. For these $\beta/2$ fraction of r, let us remove the fraction r (of probability at most $2/n$) such that $p_1(r) \in (-\delta A, \delta A)$, Recall that $\beta = \frac{1}{\log n}$, we know there are at least $\beta/4$ fraction of r remaining. We call these r "good".

Let us fixed a good r. By an averaging argument again, for any "good" r, for at least $\beta/4$ fraction of the t generated, 3 out of the 4 of the inequalities in the 4-tuple that contains t and r are satisfied. There are 4 different ways of choosing 3 out of the 4 constraints. Without loss of generality, let us assume that for $\beta/16$ fraction of the t, the first three constraints are satisfied. That is:

$$p(t^3 r + t^2 \delta \alpha) > 0 \tag{10.17}$$
$$p(t^3 r - t^2 \delta \alpha) < 0 \tag{10.18}$$
$$p(-t^3 r + t^2 \delta \alpha) > 0 \tag{10.19}$$

Let us call these t "good" for the corresponding r and define the set that contains all the "good" t for a given "good" r to be T_r. Since the possible choice of $t = n^i$ is from for each i from $[\log^2 n]$, therefore we know $|T| \geqslant (\log n)^2 \cdot \beta/16 = O(\log n)$.

Since $p(x)$ is a degree 2 polynomial, we can express $p(r + \delta \alpha)$ (by Taylor Expansion) as:

$$p(r + \delta \alpha) = \theta + p_1(r) + p_2(r) + \delta \sum c_i + \delta^2 \sum c_{ij} + \delta \sum_{1 \leqslant i \leqslant j \leqslant n} c_{ij}(r_i + r_j).$$

Denote $B = \sum c_{ij}$ and $p_2'(x) = \sum_{1 \leqslant i \leqslant j \leqslant n} c_{ij}(r_i + r_j)$. We can rewrite (10.17), (10.18), (10.19) as:

$$t^3 p_1(x) + t^2 \delta A + t^6 p_2(x) + t^5 \delta p_2'(x) + t^4 \delta^2 B + \theta > 0 \tag{10.20}$$
$$t^3 p_1(x) - t^2 \delta A + t^6 p_2(x) - t^5 \delta p_2'(x) + t^4 \delta^2 B + \theta < 0 \tag{10.21}$$
$$t^3 p_1(x) + t^2 \delta A - t^6 p_2(x) - t^5 \delta p_2'(x) - t^4 \delta^2 B - \theta > 0 \tag{10.22}$$

Notice that (10.20) and (10.22) are equivalent to

$$p_1(x) \geqslant -\delta A/t + \left| t^3 p_2(x) + \delta t^2 p_2'(x) + \delta^2 t B + \theta/t^3 \right|.$$

Since we already know that $p_1(x) \notin (-\delta A, \delta A)$ and $t \geqslant n$, therefore

$$p_1(x) \geqslant \delta A.$$

Also for (10.21), we can rewrite it as

$$p_1(x) \leqslant \delta A/t - (t^3 p_2(x) + \delta t^2 p_2'(x) - \delta^2 tB + \theta/t^3).$$

Let us further simplify the notation by denote $C = p_2(x)$, $D = \delta p_2'(x)$ and $E = \delta^2 B$. Then we rewrite above constrains as follows:

$$p_1(x) \geqslant -\delta A/t + \left| t^3 C + t^2 D + tE + \theta/t^3 \right|$$

and

$$p_1(x) \leqslant \delta A/t - (t^3 C + t^2 D - tE + \theta/t^3).$$

Notice that above (upper and lower) bound hold for any t in T_r. Therefore, we know that for any $t_1, t_2 \in T_r$,

$$\delta A/t_1 - (t_1^3 C + t_1^2 D - tE + \theta/t_1^3) \geqslant -\delta A/t_2 + \left| t_2^3 C + t^2 D + t_2 E + \theta/t_2^3 \right|$$

which is equivalent to

$$-(t_1^3 C + t_1^2 D - t_1 E + \theta/t_1^3) + \delta A(\frac{1}{t_1} + \frac{1}{t_2}) \geqslant \left| t_2^3 C + t_2^2 D + t_2 E + \theta/t_2^3 \right|. \tag{10.23}$$

Using the fact that $p(x) > \delta A$, therefore $-(t_1^3 C + t_1^2 D - t_1 E + \theta/t_1^3) > (1 - \frac{1}{t_1})\delta A$. Combing this with (10.23), we know that for any $t_1, t_2 \in T_r$, we have

$$-(t_1^3 C + t_1^2 D - t_1 E + \theta/t_1^3)(1 + \frac{(\frac{1}{t_1} + \frac{1}{t_2})}{1 - \frac{1}{t_1}}) \geqslant \left| t_2^3 C + t_2^2 D + t_2 E + \theta/t_2^3 \right|.$$

By definition, $t_i \geqslant n$ for any i; we have $\frac{(\frac{1}{t_1} + \frac{1}{t_2})}{1 - \frac{1}{t_1}} \leqslant 3/n$ Therefore, for any t_1, t_2 in T_r, the following inequality holds:

$$\frac{-(t_1^3 C + t_1^2 D - t_1 E + \theta/t_1^3)}{\left| t_2^3 C + t_2^2 D + t_2 E + \theta/t_2^3 \right|} \geqslant \frac{1}{1 + \frac{(\frac{1}{t_1} + \frac{1}{t_2})}{1 - \frac{1}{t_1}}} \geqslant 1 - 3/n. \tag{10.24}$$

We know $|T_r| = O(\beta(\log n)^2) = O(\log n)$. Actually, we only need the fact that $|T_r| \geqslant 5$. Let us pick $t_0 \leqslant t_1 \leqslant t_2 \leqslant t_3 \leqslant t_4$ from T_r, and denote $G = -(t_1^3 C + t_1^2 D - t_1 E + \theta/t_1^3)$. We know that

$$G \leqslant t_1^3 |C| + t_1^2 |D| + t_1 |E| + |\theta|/t_1^3.$$

Also for t_0, t_2, t_3, t_4, we write:

$$F_0 = t_0^3 C + t_0^2 D + t_0 E + \theta/t_0^3; \tag{10.25}$$

$$F_2 = t_2^3 C + t_2^2 D + t_2 E + \theta/t_2^3; \tag{10.26}$$

$$F_3 = t_3^3 C + t_3^2 D + t_3 E + \theta/t_3^3; \tag{10.27}$$

$$F_4 = t_4^3 C + t_4^2 D + t_4 E + \theta/t_4^3. \tag{10.28}$$

Denote $F = \max_{i=0,2,3,4} |F_i|$, by (10.24) we know

$$\frac{F}{G} \geqslant 1 - 3/n. \tag{10.29}$$

Viewing C, D, E, θ as unknown variable and solving above linear system consists equation (10.25),(10.26),(10.27),(10.28) using Cramer's rule, we have

$$C = \frac{\begin{vmatrix} F_0 & t_0^2 & t_0 & 1/t_0^3 \\ F_1 & t_2^2 & t_2 & 1/t_2^3 \\ F_2 & t_3^2 & t_3 & 1/t_3^3 \\ F_3 & t_4^2 & t_4 & 1/t_4^3 \end{vmatrix}}{\begin{vmatrix} t_0^3 & t_0^2 & t_0 & 1/t_0^3 \\ t_2^3 & t_2^2 & t_2 & 1/t_2^3 \\ t_3^3 & t_3^2 & t_3 & 1/t_3^3 \\ t_4^3 & t_4^2 & t_4 & 1/t_4^3 \end{vmatrix}}.$$

Notice that $t_0 < t_2 < t_3 < t_4$,

$$\begin{vmatrix} t_0^3 & t_0^2 & t_0 & 1/t_0^3 \\ t_2^3 & t_2^2 & t_2 & 1/t_2^3 \\ t_3^3 & t_3^2 & t_3 & 1/t_3^3 \\ t_4^3 & t_4^2 & t_4 & 1/t_4^3 \end{vmatrix}$$

is $O(t_4^3 t_3^2 t_2 t_0^{-3})$.

Since $F = \max_{i=0,2,3,4} |F_i|$, we know that

$$\begin{vmatrix} F_0 & t_0^2 & t_0 & 1/t_0^3 \\ F_1 & t_2^2 & t_2 & 1/t_2^3 \\ F_2 & t_3^2 & t_3 & 1/t_3^3 \\ F_3 & t_4^2 & t_4 & 1/t_4^3 \end{vmatrix} = O(F t_4^2 t_3 t_0^{-3})$$

Then we have $C = O(\frac{F}{t_4 t_3 t_2})$.

Fimilar analysis shows that

$$D = O(\frac{F}{t_3 t_2});$$

$$E = O(\frac{F}{t_2});$$

$$\theta = O(F t_0^3).$$

Therefore, we have

$$G \leqslant |C| t_1^3 + t_1^2 |D| + t_1 |E| + |\theta|/t_1^3 \leqslant O(F(t_1^2/t_2 t_3 + t_1/t_2 + t_1/t_3 + t_0^3/t_1)).$$

Then notice that $t_{i+1}/t_i \geqslant n$ as they are different power of n, we have

$$\frac{G}{F} = O(t_1^2/t_2 t_3 + t_1/t_2 + t_1/t_3 + t_0^3/t_1) \leqslant O(\frac{1}{n}).$$

This contradicts (10.29).

$\square$

10.3.2 Hardness Reduction from Label Cover

Recall that our reduction is from the LABEL-COVER instance $\mathcal{L}$ specified by (U,V,E,k,m,Π). For notation convenience, let us use $F(q):U\cup V\to\mathbb{N}$ to denote the possible choice of labels for vertex q; i.e., for $u\in U$, $F(u)=k$ and for $v\in V$, $F(v)=m$.

The examples in the learning problem we reduce to lies in the space $\mathbb{R}^{|U|k+|V|m}$ labeled with either positive $(+1)$ or negative (-1). Denote $\dim = |U|k+|V|m$. For $y\in\mathbb{R}^{\dim}$, each coordinate is indexed by a possible label for a vertex in $U\cup V$. We fix the following notations: For $q\in U\cup V$, we use $y_q^{(i)}$ to denote the coordinate corresponding to the vertex q's i-th label ($i\in[F(q)]$). We use vector y_q to denote all the coordinates of y corresponding to vertex q's labels.

Following is the reduction, briefly speaking we want to conduct the Dictator Test $\mathcal{T}_2$ on the restriction of $p_v(x)$ for $v\in V$. For given (U,V,E,k,m,Π) and choose the parameter to be $\beta=\frac{1}{\log m}$ and $\delta=\frac{1}{2^m}$.

Reduction from LABEL-COVER $\mathcal{L}$

1. Randomly pick an vertex $v\in V$.

2. For each $w\in U\cup V, w\neq v$, $y_w=0$.

3. Generate independent β-biased bits $a_1,a_2,\ldots,a_m\in\{0,1\}$ (i.e., $a_i=1$ with probability ϵ and 0 with probability $(1-\beta)$)

4. generate m independent Gaussian variables $g_1,\ldots,g_m$.

5. Generate t by uniform randomly pick a number $i\in\{1,2,\ldots,(\log m)^2\}$, then set $t=m^i$.

6. Generate random bit $b\in\{-1,1\}$.

7. Set $r=(a_1g_1,a_2g_2,\ldots,a_mg_m)$.

8. For $\alpha\in\mathbb{R}^n$ to be the all "1" vector $(1,1,1,\ldots,1)$, set $y_v=t^3r+bt^2\delta\alpha$.

9. Output example-label pair (y,b) (with folding steps specified later).

The learning problem is to find a degree 2 polynomial $p:\mathbb{R}^{dim}\to\{-1,1\}$ such that $\mathrm{sgn}(p(y))=b$ for as many example-label pairs as possible. Let us denote

$$p(y)=\theta+\sum_{q\in U\cup V,i\in[F(q)]}c_q^{(i)}y_q^{(i)}+\sum_{q_1,q_2\in U\cup V,i\in[F(q_1)],j\in[F(q_2)]}c_{(q_1,q_2)}^{(i,j)}y_{q_1}^{(i)}y_{q_2}^{(j)}.$$

Notice that in the reduction when vertex v is picked, we set all the coordinate to zero except y_v. Essentially, we are conducting test $\mathcal{T}_1$ on the function

$$p_v=\theta+\sum_{i\in[m]}c_v^{(i)}y_v^{(i)}+\sum_{i,j\in[m]}c_{(v(i),v(j))}y_v^{(i)}y_v^{(j)}$$

which is the restriction of $p(y)$ by setting all the coordinate to zero except those coordinates corresponding to vertex v. The fraction of agreement of $p(y)$ on all the examples is the averaging passing probability of all possible p_v (for any $v\in V$) on test $\mathcal{T}_m$.

Folding Trick: We use the "folding" technique that is similar to [60, 106]. The procedures are described as follows: instead of output pair (y,b) in the last step of above

reduction, we output (y', b) where y' is the projection of y into some subspace $H^\perp$ (defined later). By folding, we are able to enforce the $p(y)$ to have the same value on different points in $\mathbb{R}^{dim}$ as long as their projection on $H^\perp$ is the same. It is easy to see the projection can be done in polynomial time.

We define the subspace H and $H^\perp$ for our folding as follows:

Definition 10.3.4. *For every $e = (u, v) \in E, i \in [k]$, $b(e, i) \in \mathbb{R}^{\dim}$ is the vector with 0 at every coordinate except that $b(e, i)_u^{(i)} = 1$ and for every $j \in (\pi^e)^{-1}(i)$, $b(e, j)_v^{(j)} = -1$. Let B to be the collection of all such $b^{e,i}$: $B = \{b(e, i) \mid e = (u, v) \in E, i \in [k], j \in (\pi^e)^-1(i)\}$. Define $H = span(B)$ and $H^\perp$ to be the orthogonal complement of H in R^{dim}.*

After folding, we can further enforce $p(x)$ to have following "folding" property:

$$\text{For any } h \in B \text{ and } c \in \mathbb{R} , p(x + ch) = p(x).$$

and we call function that has above property folded. In particular for $e = (u, v) \in E$ and $i \in [k]$, $p(x + rb(e, i)) = p(x)$. If we view $p(y)$ as a polynomial only on $y_u^{(i)}$ and $y_v^{(j)}$ for $j \in (\pi^e)^{-1}(i)$ and apply Lemma 10.5.1, we have that

$$c_u^{(i)} = \sum_{j \in (\pi^e)^{-1}} c_v^{(j)}.$$

If we sum over all possible i, this implies for any edge (u, v),

$$\sum_{i \in k} c_u^{(i)} = \sum_{i \in m} c_v^{(i)}.$$

Now we prove our main result, Theorem 10.1.4. Recall the hardness result of LABEL-COVER as follows [128]:

Theorem 2.5.2 There exists some constant η such that it is NP-Hard to distinguish the following two cases:

- $Opt(\mathcal{L}) = 1$;
- $Opt(\mathcal{L}) \leqslant 1/m^\eta$.

We will show the following two properties of the reduction to complete the proof.

Theorem 10.3.5. *(Completeness) If $Opt(\mathcal{L}) = 1$, there is a folded function $p(x)$ that is consistent with $1 - 1/\log(m)$ fraction of the points.*

Theorem 10.3.6. *(Soundness) If $Opt(\mathcal{L}) \leqslant 1/m^\eta$, there is no folded degree 2 polynomial function consistent with $\frac{1}{2} + \frac{2}{\log^2(m)}$ fraction of the data.*

Combing Theorem 10.1.4, 10.3.5, 10.3.6 and notice that m can be arbitrary big number (e.g. e^{1/ϵ^2}),we can easy to get Theorem 10.1.4. (we also use a similar discretization argument in Section 10.2.3)

Following is the proof of above Theorem 10.3.5, 10.3.6.

Proof of Theorem 10.3.5

Proof. If $\text{Opt}(\mathcal{L}) = 1$, suppose there is a labeling l satisfying all the edges. Then consider function

$$p(y) = \sum_{w \in U \cup V} y_{w(l(w))}.$$

Notice that for every $v \in V$, p_v is a dictator and passes $\mathcal{T}_m$ with probability at least $1 - \frac{2}{\log m}$ by Lemma 10.3.1. Therefore p passes with probability at least $1 - 1/\log(m)$.

It is also easy to check that $p(x)$ is folded. □

Proof of Theorem 10.3.6

Proof. The proof is by contradiction. Suppose there is some folded degree 2 polynomial $p(x)$ such that $\text{sgn}(p(x))$ agrees with more than $\frac{1}{2} + \frac{2}{\log m}$ fraction of the example, i.e., the averaging passing probability of p_v on $\mathcal{T}_m$ is $\frac{1}{2} + \frac{2}{\log m}$. By an averaging argument, we know for $\frac{1}{\log m}$ fraction of the $v \in U$, p_v passes the test $\mathcal{T}_k$ with probability $\frac{1}{2} + \frac{1}{\log m}$ and we call such a v "good" vertex. Also we call an edge "good" if one of the endpoint of the edge is a good vertex. By the regularity of the graph, we know at least $\frac{1}{\log m}$ fraction of the edges are "good".

For a "good" vertex v, define

$$I_v = \{j \mid j \in [m], c_v^{(j)} > \sum_{i=1}^{m} c_v^{(i)}/m^2\},$$

then by Theorem 10.3.2, $|I_v| \leqslant (\log m)^2$. For every $u \in U$, we define $J_u = \{j : j \in [k], c_u^{(j)} \geqslant \sum_{i \in [k]} c_u^{(i)}/k\}$.

Notice J_u is not empty as

$$\max_j c_u^{(j)} \geqslant \sum_{i \in [k]} c_{u[i]}/k.$$

We define the following labeling strategy for $\mathcal{L}$. For $u \in U$, randomly assign it a label from J_u; for $v \in V$, we randomly assign it a label from I_v (if I_v is empty, just assign any label).

For every good edge $e = (u, v)$ and any $j \in J_u$, by folding, we have

$$\sum_{i \in \pi_e^{-1}(j)} c_v^{(i)} = c_u^{(j)} \geqslant \sum_{i \in [k]} c_u^{(i)}/k = \sum_{i \in [m]} c_v^{(i)}/k.$$

There is at least one label i in $\pi_e^{-1}(j)$ such that $\sum_{i \in [m]} c_v^{(i)}/km \geqslant \sum_{i \in [m]} c_v^{(i)}/m^2$, and it is therefore in I_v. Notice that $I_v \leqslant (\log m)^2$, by our randomized labeling strategy, we have $1/(\log m)^2$ chance to satisfy edge (u, v).

Therefore above labelling strategy satisfy (in expectation) at least $1/(\log(m)^2)$ fraction of the good edges and $1/(\log m)^3$ fraction of the total edges. For large enough m, this contradicts with the fact that $\text{Opt}(\mathcal{L}) \leqslant 1/m^\eta$. □

10.4 Probability Inequalities

The second fact is from [27].

Lemma 10.4.1. *Let $f(x) : \mathbb{R}^n \to \mathbb{R}$ be a degree-d polynomial function, then for $x_1, x_2, \ldots, x_n$ being independent unit Gaussian,*

$$\mathbf{Pr}[|f(x_1, \ldots, x_n)| \leqslant \alpha] \leqslant O(d \, \mathbf{E}[f(x_1, \ldots, x_n)^2]^{-\frac{1}{2d}} \alpha^{1/d}).$$

where $x_1, x_2, \ldots, x_n$ are independent standard Gaussian.

Lemma 10.4.2. *For a degree d polynomial $f = \sum_{multiset S, |S| \leqslant d, S \subseteq [2n]} c_S \prod_{i \in S} x_i$ and for any multset $T \subseteq [n]$, $\|f\|_2 = \mathbf{E}[f(x_1, \ldots, x_n)^2]^{\frac{1}{2}} \geqslant \frac{|\hat{f}(T)|}{\binom{n+d}{d} d^d}$.*

Proof. Given f, one way to calculate its $\|f_2\|_2^2$ is to convert it into its hermite expansion $\sum_S \hat{f}(S) \chi_S$ where χ_S is the Hermite polynomial. Then the variance is $\sum \hat{f}(S)^2$.

of the form $\chi_S(x) = \sum_{T \subseteq S} h_S^T \prod_{i \in T} x_i$ where h_S^T is the coefficients of the Hermite polynomial h_S.

We know that c_T can be written as the sum of $\sum_{T \subseteq S} h_S^T \hat{f}(S)$. There are at most $\binom{n+d}{d}$ term in the summation$\ldots$, Also every h_S^T is some constant only depending on d actually it is not hard to bound it by d^d. Therefore, there must be at least one hermite coefficients $\hat{f}(S)$ that has absolute value bigger than $\frac{|\hat{f}(T)|}{d^d \binom{n+d}{d}}$ and this give a lower bound for $\|f\|_2$. $\qquad \square$

10.5 Folding Lemma

Lemma 10.5.1.

$$p(x) = \theta + \sum_{i=0}^{n} w_i x_i + \sum_{0 \leqslant i \leqslant j \leqslant n} w_{ij} x_i x_j$$

is a degree 2 function. If for any $c \in \mathbb{R}$ and $f(x + c(1, -1, \ldots, -1)) = f(x)$, then $w_0 = \sum_{i=1}^{n} w_i$.

Proof. We know that

$$\theta + w_0(x_0 + c) + \sum_{i=1}^{n} w_i(x_i - c) + w_{00}(x_0 + c)^2 + \sum_{j=1}^{n} w_{0j}(x_0 + c)(x_j - c) + \sum_{1 \leqslant i \leqslant j \leqslant n} w_{ij}(x_i - c)(x_j - c)$$

$$= \theta + \sum_{i=0}^{n} w_i x_i + \sum_{0 \leqslant i \leqslant j \leqslant n} w_{ij} x_i x_j.$$

Notice that above equation hold for any c, x. Therefore if we express left hand and right hand as polynomials of variable $c, x_0, x_1, \ldots, x_n$, the corresponding coefficients should be the same. If we look at the coefficients of the term c, we have

$$w_0 - \sum_{i=1}^{n} w_i = 0.$$

$\qquad \square$

221

10.6 Discretization of the Gaussian distribution

Theorem 10.6.1. *There is a probability distribution on $(\mathscr{G}, \mathscr{H}_N) \sim \mathbb{R}^2$ such that the marginal distribution on $\mathscr{G}$ follows the standard Gaussian distribution and the marginal distribution of $\mathscr{H}_N$ follows the distribution of the summation of N random bits; i.e., $\mathscr{H}_N = \sum_{i=1}^N b_i$ where each b_i is independent random bits from $\{-1, 1\}$. In addition, $\mathscr{H}_N$ and $\mathscr{G}$ are close in the following sense: with probability at least $1 - O(\frac{1}{N^{\frac{1}{4}}})$, $|\mathscr{G} - \frac{\mathscr{H}_N}{\sqrt{N}}| \leqslant O(\frac{1}{N^{\frac{1}{4}}}))$.*

Proof. Let Φ be the CDF (Cumulative distribution function) of $\mathscr{H}_N$ and Ψ be the CDF of the marginal distribution on $\mathscr{G}$ (i.e. standard Gaussian Distribution).

We can couple random variables $\mathscr{G}, \mathscr{H}_N$ in the following ways: first we sample h_0 from the marginal distribution on $\mathscr{H}_N$. We know that

$$\mathbf{Pr}(\mathscr{H}_N = h_0) = \Psi(h_0) - \Psi(h_0 - 2)$$

since if h_0 is a feasible outcome of summing N bits, then $h_0 - 2$ is the biggest feasible outcome less than h_0 (if there is any). Then we generate $\mathscr{G}$ by keep drawing random samples from the Gaussian Distribution until the sample falls into the interval $(\Psi^{-1}(\Phi(h_0 - 2)), \Psi^{-1}(\Phi(h_0)]$ and we set its value to be $\mathscr{G}$.

By above construction, we claim $\mathscr{G}$ must follows the Gaussian distribution: essentially we use the value of h_0 as a indicator of whether $\mathscr{G}$ is in the interval $(\Psi^{-1}(\Phi(h_0 - 2)), \Psi^{-1}(\Phi(h_0)]$. We also need to check that $\mathbf{Pr}(h = h_0) = \mathbf{Pr}(\mathscr{G} \in (\Psi^{-1}(\Phi(h_0 - 2)), \Psi^{-1}(\Phi(h_0)]$. This is true because

$$\mathbf{Pr}(h = h_0) = \mathbf{Pr}(h \in (h_0 - 2, h_0] = \Phi(h_0) - \Phi(h_0 - 2)$$
$$= \mathbf{Pr}(g \in (\Psi^{-1}(\Phi(h_0 - 2)), \Psi^{-1}(\Phi(h_0)]).$$

By above coupling of $\mathscr{G}$ and $\mathscr{H}$, it remains to prove for any number in the interval $(\Psi^{-1}(\Phi(h_0 - 2)), \Psi^{-1}(\Phi(h_0))]$ is close to $h_0/\sqrt{n}$ for most of the h_0 generated.

It suffice to check the following two inequalities:

- $|\Psi^{-1}(\Phi(h_0)) - \frac{h_0}{\sqrt{N}}| \leqslant \frac{1}{N^{\frac{1}{4}}}$;

- $|\Psi^{-1}(\Phi(h_0 - 2)) - \frac{h_0}{\sqrt{N}}| \leqslant \frac{1}{N^{\frac{1}{4}}}$.

Suppose $h_0 > 0$ and we will just prove the second one. By the Berry Esseen Theorem, we know that $|\Phi(h_0 + 2) - \Psi(\frac{h_0 + 2}{\sqrt{N}})| \leqslant \frac{1}{\sqrt{N}}$.

Therefore,

$$\Psi^{-1}(\Phi(h_0 + 2)) \leqslant \Psi^{-1}(\Psi(\frac{h_0 + 2}{\sqrt{N}}) + \frac{1}{\sqrt{N}}) \leqslant \frac{h_0 + 2}{\sqrt{N}} + O(\frac{\frac{1}{\sqrt{N}}}{e^{-(s+2)^2/2}}). \tag{10.30}$$

Notice that when $h_0 \leqslant \sqrt{\frac{\log N}{2}}$, (10.30) is bounded by $O(\frac{1}{N^{1/4}})$. Also by Chernoff Bound, we know that when $\mathbf{Pr}(\mathscr{H} > \sqrt{\frac{\log N}{2}}) \leqslant O(\frac{1}{N^{\frac{1}{4}}})$.

Therefore, except with probability $O(\frac{1}{N^{\frac{1}{4}}})$, $|\mathcal{G} - \mathcal{H}| \leqslant O(\frac{1}{N^{\frac{1}{4}}})$

$\square$

By above theorem, we know that we can construct a distribution $\mathcal{H}_N$ that is point-wise close a Gaussian distribution $\mathcal{G}$ with high probability. Now we will use the constructed distribution to discretize the high dimension Gaussian space for low degree PTFs.

Theorem 10.6.2. *For any degree D polynomial $f(x_1, \ldots, x_n) = \sum_{|S| \leqslant D} \hat{f}(S) \prod_{i \in S} x_i$. Here D is some constant that does not depend on n. Let $(y, z) \in \mathbb{R}^n \times \mathbb{R}^n$ be generated by sample n times i.i.d from the distribution $(\mathcal{G}, \mathcal{H}_N)^{\otimes n}$ where we take $N = n^{24D^2}$ as is the set up of Theorem 10.6.1.*

Then

$$\mathbf{Pr}(\mathrm{sgn}(f(y)) \neq \mathrm{sgn}(f(z)) \leqslant O(\frac{1}{n}).$$

Proof. Without loss of generality, let us assume that $\sum_{S \neq \emptyset} |\hat{f}(S)| = 1$. By Lemma 10.4.2, we know that $\|f_2\| \geqslant \frac{1}{\binom{n+D}{D} D^D}$.

By union bound and Theorem 10.6.1, we know that with probability $1 - \frac{n}{N^{1/4}} = 1 - O(\frac{1}{n})$, we have that for every $i \in [n]$, $|x_i - y_i| \leqslant \frac{1}{N^{1/4}}$

Similar to the calculation in (10.3), when y and z are close on each coordinate, we have that

$$|f(y) - f(z)| \leqslant \frac{1}{N^{\frac{1}{4}}} O(n^{2D^2}) \leqslant \frac{1}{n^{3D^2}}$$

Then

$$\mathbf{Pr}(\mathrm{sgn}(f(y)) \neq \mathrm{sgn}(f(z)) \leqslant \mathbf{Pr}(f(y) \leqslant |f(z) - f(y)|)$$

$$\leqslant O(\frac{1}{n}) + \mathbf{Pr}(f(y) \leqslant \frac{1}{n^{3D^2}}) \quad (10.31)$$

By Lemma 10.4.1, we can bound

$$\mathbf{Pr}(f(y) \leqslant \frac{1}{n^{3D^2}})$$

by $O(\frac{1}{n})$.

Overall we bound the probability of $\mathbf{Pr}(\mathrm{sgn}(f(y)) \neq \mathrm{sgn}(f(z))$ by $O(\frac{1}{n})$. $\square$

Remark 10.6.3. *Above theorem immediately implies that $(\frac{\mathcal{H}_N}{\sqrt{N}})^{\otimes n}$ can be used to fool low degree PTFs over $\mathcal{G}^{\otimes n}$. Also the distribution of $\mathcal{H}_N^{\otimes n}$, by definition, can be generated with $n^{24D^2 + 1}$ random bits.*

Part IV

Open Problems

Chapter 11

Open Problems

Numerous problems are unsolved on understanding the approximability of NP-hard problems. One of the most important one is to prove or disprove the Unique Games Conjecture as well the d-to-1 conjecture. In addition to that, I list the open problems that I found intriguing, and hard to solve, during the writing of my thesis.

Efficient SDP Rounding for CSPs: In the study of MAX CUT, we gave an SDP rounding algorithm with running time $\text{poly}(n) \cdot 2^{\text{poly}(1/\epsilon)}$. Can we improve its running time to $\text{poly}(n) \cdot \text{poly}(1/\epsilon)$? Can we even give an efficient rounding algorithm for general CSPs? In the work of [125], the author gave a generic SDP rounding algorithm for almost every CSP with running time $\text{poly}(n) \cdot 2^{2^{\text{poly}(1/\epsilon)}}$. Can we improve such a running time to make the algorithm more practical?

NP-hardness for Satisfiable 3-CSP: Can we prove that MAX 3-CSP $(1, 5/8 + \epsilon)$ is NP-hard without assuming the d-to-1 conjecture?

Hardness of Approximating Satisfiable CSPs: Can we establish a more general result on the approximability of satisfiable CSPs? In particular can we prove or disprove the following conjecture:

Conjecture 11.0.4. *Let Φ be a predicate set. For the problem of* MAX Φ, $\text{Gap}_{\text{Test}}(1)$, *which is the optimal soundness of the Dictator Test using predicates from set ϕ with perfect completeness, is equal to the optimal approximation ratio for* MAX Φ *when the instance is satisfiable.*

SDP gap for 2-to-1 LABEL-COVER: Can we construct instances of 2-to-1 LABEL-COVER with SDP value 1 and optimum value $1/R^{\theta(1)}$ where R is the alphabet size? More desirably, can we obtain such a gap under even stronger form of SDP for 2-to-1 LABEL-COVER?

NP hardness results of MA-MON-PTF$_d$ $(1 - \epsilon, 1/2 + \epsilon)$: Can we show the that even there exists a monomials that is consistent with 0.99 fraction of the data, it is hard to find a low degree PTF that is consistent with 0.51 fraction of the examples. Such a hardness result would subsume almost all the previous results on hardness of agnostic learning and strengthen the belief that learning tasks under agnostic noises over arbitrary distribution are essentially hard.

Bibliography

[1] Noga Alon, Konstantin Makarychev, Yury Makarychev, and Assaf Naor. Quadratic forms on graphs. *Inventiones Mathematicae*, 163(3):499–522, 2006. 4.1.5

[2] Noga Alon and Assaf Naor. Approximating the Cut-Norm via Grothendieck's Inequality. *SIAM Journal on Computing*, 35(4):787–803, 2006. 4.1.5

[3] Noga Alon and Benjamin Sudakov. Bipartite subgraphs and the smallest eigenvalue. *Combinatorics, Probability & Computing*, 9(1), 2000. 4.1, 4.1.8, 4.1.8, 4.7, 4.10, 4.10

[4] Noga Alon, Benny Sudakov, and Uri Zwick. Constructing worst case instances for semidefinite programming based approximation algorithms. *SIAM Journal on Discrete Mathematics*, 15(1):58–72, 2002. 2.2, 4.1, 4.1.8, 4.1.8, 4.7, 4.10, 4.10

[5] Edoardo Amaldi and Viggo Kann. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 109:237–260, 1998. 2.4.3

[6] D. Angluin and P. Laird. Learning from noisy examples. *Machine Learning*, 2:343–370, 1988. 2.4.3

[7] Sanjeev Arora, László Babai, Jacques Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *Journal of Computer and System Sciences*, 54(2):317–331, 1997. 2.4.3

[8] Sanjeev Arora, Eli Berger, Elad Hazan, Guy Kindler, and Muli Safra. On non-approximability for quadratic programs. In *Proc. 46th IEEE Symp. on Foundations of Comp. Sci.*, pages 206–215, 2005. 4.1.5, 4.3.1, 4.12

[9] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. 5.1.1

[10] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: A new characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998. 5.1.1

[11] Takao Asano and David P. Williamson. Improved approximation algorithms for max sat. *J. Algorithms*, 42(1):173–202, 2002. 1.2.1

[12] Peter Auer and Manfred K. Warmuth. Tracking the best disjunction. *Machine Learning*, 32(2):127–150, 1998. 2.4.3

[13] Per Austrin. Balanced Max-2Sat might not be hardest. In *Proc. 39th ACM Symp. on the Theory of Computing*, pages 189–197, 2007. 1.2.1, 4.1.5

[14] Per Austrin. Towards sharp inapproximability for any 2-CSP. In *Proc. 48th IEEE Symp. on Foundations of Comp. Sci.*, 2007. 1.2.1, 4.1.5

[15] Adi Avidor and Uri Zwick. Rounding two and three dimensional solutions of the SDP relaxation of MAX CUT. In *Proc. 8th APPROX-RANDOM*, pages 14–25, 2005. 1.2.1, 4.1

[16] Nina Balcan and Avrim Blum. Approximation algorithms and online mechanisms for item pricing. In *Proc. 7th ACM Conference on Electronic commerce*, pages 29–35, New York, NY, USA, 2006. ACM. 2.4.3, 8.1.1

[17] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine Learning*, 56(1-3):89–113, 2004. 4.1.5

[18] William Beckner. Sobolev inequalities, the Poisson semigroup, and analysis on the sphere S^n. *Proc. Natl. Acad. Sci.*, 89:4816–4819, 1992. 4

[19] Mihir Bellare, Oded Goldreich, and Madhu Sudan. Free bits, PCPs, and nonapproximability — towards tight results. *SIAM Journal on Computing*, 27(3):804–915, 1998. 2.6.1, 4.12, 5.1.1, 5.1.1, 5.2.4

[20] Shai Ben-David, Nadav Eiron, and Philip M. Long. On the difficulty of approximately maximizing agreements. *Journal of Computer and System Sciences*, 66(3):496–514, 2003. 2.4.3

[21] E. Blais, R. O'Donnell, and K. Wimmer. Polynomial regression under arbitrary product distributions. In *Proc. 21st Annual Conference on Learning Theory (COLT)*, pages 193–204, 2008. 10.1.1

[22] Avrim Blum, Alan Frieze, Ravi Kannan, and Santosh Vempala. A polynomial-time algorithm for learning noisy linear threshold functions. *Algorithmica*, 22(1-2):35–52, 1998. 2.4.3

[23] Aline Bonami. Ensembles $\Lambda(p)$ dans le dual de D^∞. *Ann. Inst. Fourier*, 18(2):193–204, 1968. 3.2.2

[24] Christer Borell. Geometric bounds on the Ornstein-Uhlenbeck velocity process. *Z. Wahrsch. Verw. Gebiete*, 70(1):1–13, 1985. 4.1.5, 4.2.3, 4.3.3

[25] Mark Braverman and Stephen Cook. Computing over the reals: Foundations for scientific computing. *Notices of the AMS*, 53(3):318–329, 2006. 4.6.1

[26] Patrick Briest and Piotr Krysta. Single-minded unlimited supply pricing on sparse instances. In *Proc. 17th ACM-SIAM Symp. on Discrete Algorithms*, pages 1093–1102, New York, NY, USA, 2006. ACM. 1.3.2, 8.1.1

[27] A. Carbery and J. Wright. Distributional and L_q norm inequalities for polynomials over convex bodies in $\mathbb{R}^n$. 8(3):233–248, 2001. 10.4

[28] Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Near-optimal algorithms for unique games. In *Proc. 38th ACM Symp. on the Theory of Computing*, pages 205–214, 2006. 6.1.3, 6.2

[29] Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Near-optimal algorithms for maximum constraint satisfaction problems. In *SODA '07: Proceedings of*

the eighteenth annual ACM-SIAM symposium on Discrete algorithms, pages 62–68, 2007. 1.2.1

[30] Moses Charikar and Anthony Wirth. Maximizing quadratic programs: Extending Grothendieck's Inequality. In *Proc. 45th IEEE Symp. on Foundations of Comp. Sci.*, pages 54–60, 2004. 4.1, 4.1.3, 4.1.5, 4.1.7, 4.1.8, 4.11, 4.14, 4.14

[31] Sourav Chatterjee. A simple invariance theorem. *arxiv:math/0508213v1.*, 2005. 9.3.2

[32] Eden Chlamtac and Gyanit Singh. Improved approximation guarantees through higher levels of sdp hierarchies. In *APPROX '08 / RANDOM '08: Proceedings of the 11th international workshop, APPROX 2008, and 12th international workshop, RANDOM 2008 on Approximation, Randomization and Combinatorial Optimization*, pages 49–62, Berlin, Heidelberg, 2008. Springer-Verlag. 1.2.1

[33] Edith Cohen. Learning noisy perceptrons by a perceptron in polynomial time. In *Proceedings of the 38th IEEE Symposium on the Foundations of Computer Science*, pages 514–523, 1997. 2.4.3

[34] Stephen A. Cook. The complexity of theorem-proving procedures. In *STOC '71: Proceedings of the third annual ACM symposium on Theory of computing*, pages 151–158, 1971. 1.2.1

[35] Pierluigi Crescenzi, Riccardo Silvestri, and Luca Trevisan. On weighted vs unweighted versions of combinatorial optimization problems. *Information and Computation*, 167(1):10–26, 2001. 4.12, 1

[36] Charles Delorme and Svatopluk Poljak. Combinatorial properties and the complexity of max-cut approximation. *Eur. J. Combinatorics*, 14(4):313–333, 1993. 4.1

[37] Charles Delorme and Svatopluk Poljak. Laplacian eigenvalues and the maximum cut problem. *Mathematical Programming*, 62:557–574, 1993. 1.2.1, 4.1, 1, 4.1.2

[38] Erik D. Demaine, Uriel Feige, Mohammadtaghi Hajiaghayi, and Mohammad R. Salavatipour. Combination can be hard: Approximability of the unique coverage problem. In *Proc. 17th ACM-SIAM Symp. on Discrete Algorithms*, pages 162–171, 2006. 8.1.1

[39] Persi Diaconis and David Freedman. A dozen Finetti-style results in search of a theorem. *Annals de l'I. H. P., sec. B*, 23(S2):397–423, 1987. 4.10

[40] Persi Diaconis and Laurent Saloff-Coste. Logarithmic Sobolev inequalities for finite Markov chains. *Annals of Applied Probability*, 6(3):695–750, 1996. 3.2.2

[41] Ilias Diakonikolas, Parikshit Gopalan, Ragesh Jaiswal, Rocco A. Servedio, and Emanuele Viola. Bounded independence fools halfspaces. In *FOCS*, pages 171–180, 2009. 9.2, 9.3.1, 9.3.1, 9.3.2, 9.3.1, 9.3.6

[42] Ilias Diakonikolas, Prahladh Harsha, Adam Klivans, Raghu Meka, Prasad Raghavendra, Rocco A. Servedio, and Li-Yang Tan. Bounding the average sensitivity and noise sensitivity of polynomial threshold functions. In *STOC*, pages 533–542, 2010. 2.4.3, 10.1.1

[43] Irit Dinur, Elchanan Mossel, and Oded Regev. Conditional hardness for approximate

coloring. In *Proc. 38th ACM Symp. on the Theory of Computing*, pages 344–353, 2006. 1.2.1, 2.5.1, 3.2.1, 6.1.1, 2, 6.3.1, 3

[44] Irit Dinur and Samuel Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162(1):439–485, 2005. 6.1.1

[45] B. Efron and C. Stein. The jackknife estimate of variance. *The Annals of Statistics*, 9(3):586–596, 1981. 3.1.8

[46] Lars Engebretsen, Piotr Indyk, and Ryan O'Donnell. Derandomized dimensionality reduction with applications. In *Proc. 13th ACM-SIAM Symp. on Discrete Algorithms*, pages 705–712, 2002. 4.6.2, 4.13, 4.13

[47] Uriel Feige, Marek Karpinski, and Michael Langberg. Improved approximation of Max-Cut on graphs of bounded degree. *J. Algorithms*, 43(2):201–219, 2002. 4.1

[48] Uriel Feige and Michael Langberg. The RPR^2 rounding technique for semidefinite programs. *J. Algorithms*, 60(1):1–23, 2006. First appeared ICALP 2001. 4.1, 4.1.3, 4.1.3, 4.1.5, 4.1.6, 4.1.8, 4.5, 4.13

[49] Uriel Feige and Lászlo Lovász. Two-prover one-round proof systems, their power and their problems. In *Proc. 24th ACM Symp. on the Theory of Computing*, pages 733–744, 1992. 1.2.1, 4.1.2

[50] Uriel Feige and Gideon Schechtman. On the optimality of the random hyperplane rounding technique for Max-Cut. *Random Structures & Algorithms*, 20(3):403–440, 2002. 1.3.2, 4.1, 4.1.5, 4.1.8, 4.1.8, 4.2.1, 4.3.1, 4.10, 4.10.1, 4.10, 4.10, 4.12, 4.13

[51] Vitaly Feldman. Optimal hardness results for maximizing agreements with monomials. In *Proceedings of the 21st Annual IEEE Conference on Computational Complexity*, pages 226–236, 2006. 2.4.3

[52] Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. New results for learning noisy parities and halfspaces. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 563–574, 2006. 2.4.3

[53] Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. New results for learning noisy parities and halfspaces. In *IEEE FOCS*, pages 563–574, 2006. 2.6.2

[54] Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. On agnostic learning of parities, monomials, and halfspaces. *SIAM J. Comput.*, 39(2):606–645, 2009. 10.1.1

[55] Vitaly Feldman, Venkatesan Guruswami, Prasad Raghavendra, and Yi Wu. Agnostic learning of monomials by halfspaces is hard. In *Proc. 50th IEEE Symp. on Foundations of Comp. Sci.*, page To appear, 2009. 10.1.1

[56] Maria florina Balcan, Avrim Blum, Hubert Chan, and Mohammadtaghi Hajiaghayi. Hajiaghayi: A theory of loss-leaders: Making money by pricing below cost. In *Proc. 3rd Intern. Workshop on Internet and Network Economics, Lecture Notes in Computer Science*. Springer, 2007. 1.3.2, 8.1.1, 8.1.2, 8.1.2

[57] S. Galant. Perceptron based learning algorithms. *IEEE Trans. on Neural Networks*,

1(2), 1990. 2.4.3

[58] Claudio Gentile and Manfred K. Warmuth. Linear hinge loss and average margin. In *Proceedings of NIPS*, pages 225–231, 1998. 2.4.3

[59] Michel Goemans and David Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42:1115–1145, 1995. 1.2.1, 2.2, 2.3.2, 4.1, 4.1.2, 4.1.5, 4.1.8, 4.11, 4.13

[60] Parikshit Gopalan, Subhash Khot, and Rishi Saket. Hardness of reconstructing multivariate polynomials over finite fields. In *FOCS '07: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 349–359, Washington, DC, USA, 2007. IEEE Computer Society. 10.1.3, 10.3.2

[61] Parikshit Gopalan, Subhash Khot, and Rishi Saket. Hardness of reconstructing multivariate polynomials over finite fields. *SIAM J. Comput.*, 39(6):2598–2621, 2010. 9.10

[62] Leonard Gross. Logarithmic Sobolev inequalities. *Amer. J. Math.*, 97:1061–1083, 1975. 3.2.2

[63] Alexandre Grothendieck. Résumé de la théorie métrique des produits tensoriels topologiques. *Bol. Soc. Mat. Sao Paulo*, 8:1–79, 1953. 4.1.5

[64] Venkatesan Guruswami, Jason D. Hartline, Anna R. Karlin, David Kempe, Claire Kenyon, and Frank McSherry. On profit-maximizing envy-free pricing. In *Proc. 15th ACM-SIAM Symp. on Discrete Algorithms*, pages 1164–1173, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics. 8.1.1

[65] Venkatesan Guruswami and Subhash Khot. Hardness of max 3SAT with no mixed clauses. In *Proc. 20th IEEE Conference on Computational Complexity*, pages 154–162, 2005. 1.2.1

[66] Venkatesan Guruswami, Daniel Lewin, Madhu Sudan, and Luca Trevisan. A tight characterization of NP with 3 query PCPs. *Electronic Colloq. on Comp. Complexity (ECCC)*, 5(34), 1998. 5.1.1, 5.1.1

[67] Venkatesan Guruswami, Rajsekar Manokaran, and Prasad Raghavendra. Beating the random ordering is hard: Inapproximability of maximum acyclic subgraph. In *Proc. 49th IEEE Symp. on Foundations of Comp. Sci.*, pages 573–582, 2008. 1.2.1, 6.1.4, 7.2

[68] Venkatesan Guruswami and Prasad Raghavendra. Hardness of learning halfspaces with noise. In *Proc. 47th IEEE Symp. on Foundations of Comp. Sci.*, pages 543–552, 2006. 2.2, 2.4.3, 2.6.2

[69] Venkatesan Guruswami and Prasad Raghavendra. A 3-query PCP over integers. In *Proc. 39th ACM Symp. on the Theory of Computing*, pages 198–206, 2007. 7.1.2, 7.2.1

[70] Venkatesan Guruswami and Prasad Raghavendra. Hardness of learning halfspaces with noise. *SIAM J. Comput.*, 39(2):742–765, 2009. 10.1.1

[71] Venkatesan Guruswami and Ali Kemal Sinop. Improved inapproximability results for maximum k-colorable subgraph. In *Proceedings of the 12th International Work-*

shop on Approximation, Randomization, and Combinatorial Optimization: Algorithms and Techniques (APPROX), pages 163–176, 2009. 2.5.1

[72] Uffe Haagerup. A new upper bound for the complex Grothendieck constant. *Israel J. of Math.*, 60:199–224, 1987. 4.6.1

[73] Eran Halperin, Dror Livnat, and Uri Zwick. MAX CUT in cubic graphs. *J. Algorithms*, 53(2):169–185, 2004. 4.1

[74] Johan Håstad. Some optimal inapproximability results. In *Proc. 29th ACM Symp. on the Theory of Computing*, pages 1–10, 1997. 7.1.2

[75] Johan Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica*, 182(1):105–142, 1999. 2.6.1

[76] Johan Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001. 1.2.1, 2.6.1, 4.1.8, 5.1.1, 5.1.1, 5.1.3, 5.2.4

[77] Johan Håstad and Srinivasan Venkatesh. On the advantage over a random assignment. *Random Structures & Algorithms*, 25(2):117–149, 2004. 4.1.5

[78] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, 100(1):78–150, 1992. 1.2.2, 2.4.2

[79] D. Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and Computation*, 100(1):78–150, 1992. 2.4.2

[80] K. Hoffgen, K. van Horn, and H. U. Simon. Robust trainability of single neurons. *J. Comput. Syst. Sci.*, 50(1):114–125, 1995. 2.4.3

[81] J. Jackson. An efficient membership-query algorithm for learning DNF with respect to the uniform distribution. *Journal of Computer and System Sciences*, 55:414–440, 1997. 10.1.1

[82] David S. Johnson. Approximation algorithms for combinatorial problems. In *STOC '73: Proceedings of the fifth annual ACM symposium on Theory of computing*, pages 38–49, 1973. 1.2.1

[83] Jeff Kahn, Gil Kalai, and Nathan Linial. The influence of variables on boolean functions. In *Proc. 29th IEEE Symp. on Foundations of Comp. Sci.*, pages 68–80, 1988. 2

[84] Adam Kalai, Adam Klivans, Yishay Mansour, and Rocco Servedio. Agnostically learning halfspaces. pages 11–20, 2005. 2.4.3, 10.1.1

[85] D.M. Kane. The Gaussian surface area and noise sensitivity of degree-d polynomial threshold functions. CCC 2010, to appear, 2010. 10.1.1

[86] Howard Karloff. How good is the Goemans-Williamson MAX CUT algorithm? *SIAM Journal on Computing*, 29(1):336–350, 1999. 4.1, 4.1.8, 4.1.8, 4.7, 4.10, 9, 4.10

[87] Howard Karloff and Uri Zwick. A 7/8-approximation algorithm for Max-3Sat? In *Proc. 38th IEEE Symp. on Foundations of Comp. Sci.*, pages 406–415, 1997. 1.2.1

[88] Richard Karp. *Reducibility among combinatorial problems*, pages 85–103. Plenum

234

Press, 1972. 1.2.1, 4.1

[89] M. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6):983–1006, 1998. 2.4.3

[90] Michael Kearns, Robert Schapire, and Linda Sellie. Toward efficient agnostic learning. *Machine Learning*, 17:115–141, 1994. 1.2.2, 2.4.2, 2.4.2, 2.4.3

[91] Michael J. Kearns and Ming Li. Learning in the presence of malicious errors. *SIAM J. Comput.*, 22(4):807–837, 1993. 2.4.3

[92] Michael J. Kearns and Robert E. Schapire. Efficient distribution-free learning of probabilistic concepts. *J. Comput. Syst. Sci.*, 48(3):464–497, 1994. 2.4.3

[93] Rohit Khandekar, Tracy Kimbrel, Konstantin Makarychev, and Maxim Sviridenko. On hardness of pricing items for single-minded bidders. In *APPROX-RANDOM*, pages 202–216, 2009. 8.1.1, 8.1.2

[94] S. Khot and O. Regev. Vertex cover might be hard to approximate to within $2 - \epsilon$. *Journal of Computer & System Sciences*, 74(3):335–349, 2008. 7.5

[95] Subash Khot. New techniques for probabilistically checkable proofs and inapproximability results (thesis). *Princeton University Technical Reports*, TR-673-03, 2003. 9.2, 9.10

[96] Subash Khot and Rishi Saket. Sdp integrality gaps with local l1-embeddability. In *Proc. 50th IEEE Symp. on Foundations of Comp. Sci.*, 2009. To appear. 6.1.4

[97] Subhash Khot. On the power of unique 2-prover 1-round games. In *Proc. 34th ACM Symp. on the Theory of Computing*, pages 767–775, 2002. 1.2.1, 2.5, 1

[98] Subhash Khot. On the power of unique 2-Prover 1-Round games. In *ACM STOC*, pages 767–775, May 19–21 2002. 9.2

[99] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM Journal on Computing*, 37(1):319–357, 2007. 1.2.1, 1.2.1, 2.5, 2.5, 4.1, 4.1.4, 4.1.11, 4.1.8, 4.1.8, 4.7, 4.8, 4.9, 4.10, 7.1.1, 7.3.3, 7.5, 7.5.1, 8.2.1, 8.4

[100] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM J. Comput*, 37(1):319–357, 2007. 9.2

[101] Subhash Khot and Dana Moshkovitz. Hardness of approximately solving linear equations over reals. *Electronic Colloq. on Comp. Complexity (ECCC)*, 5(53), 2010. 7.1.2

[102] Subhash Khot and Ryan O'Donnell. SDP gaps and UGC-hardness for MaxCutGain. In *Proc. 47th IEEE Symp. on Foundations of Comp. Sci.*, pages 217–226, 2006. 1.2.1, 4.1, 4.1.7, 4.1.8, 4.2.2, 4.3.1, 4.3.1, 4.3.3, 4.8, 4.12, 4.14, 4.14, 4.14

[103] Subhash Khot and Oded Regev. Vertex Cover might be hard to approximate to within $2 - \epsilon$. In *Proc. 18th IEEE Conference on Computational Complexity*, pages 379–386, 2003. 1.2.1, 8.4, 9.2

[104] Subhash Khot and Rishi Saket. A 3-query non-adaptive PCP with perfect comple-

ness. In *Proc. 21st IEEE Conference on Computational Complexity*, pages 159–169, 2006. 1.3.2, 5.1.1

[105] Subhash Khot and Rishi Saket. Hardness of minimizing and learning DNF expressions. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 231–240, 2008. 2.4.3

[106] Subhash Khot and Rishi Saket. On hardness of learning intersection of two halfspaces. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 345–354, 2008. 10.1.3, 10.3.2

[107] Subhash Khot and Nisheeth Vishnoi. The Unique Games Conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ_1. In *Proc. 46th IEEE Symp. on Foundations of Comp. Sci.*, pages 53–62, 2005. 4.1, 4.1.8, 4.7, 6.1.3, 6.1.4, 6.2, 6.4, 6.7

[108] A. Klivans, R. O'Donnell, and R. Servedio. Learning geometric concepts via Gaussian surface area. In *Proc. 49th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 541–550, 2008. 10.1.1

[109] W. S. Lee, P. L. Bartlett, and R. C. Williamson. On efficient agnostic learning of linear combinations of basis functions. pages 369–376, 1995. 2.4.3

[110] Michael Lewin, Dror Livnat, and Uri Zwick. Improved rounding techniques for the max 2-sat and max di-cut problems. In *Proceedings of the 9th International IPCO Conference on Integer Programming and Combinatorial Optimization*, pages 67–82, 2002. 1.2.1

[111] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, fourier transform, and learnability. *J. ACM*, 40(3):607–620, 1993. 10.1.1

[112] Sanjeev Mahajan and Hariharan Ramesh. Derandomizing approximation algorithms based on semidefinite programming. *SIAM Journal on Computing*, 28(5):1641–1663, 1999. 4.13, 4.13, 4.13

[113] Kevin Matulef, Ryan O'Donnell, Ronitt Rubinfeld, and Rocco A. Servedio. Testing halfspaces. In *SODA '09: Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms*, pages 256–264, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics. 9.2, 9.3.1

[114] E. Mossel. Gaussian bounds for noise correlation of functions. *In FOCS '08: Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science*, 2008. 9.3.2

[115] Elchanan Mossel. Gaussian bounds for noise correlation of functions and tight analysis of long codes. In *Proc. 49th IEEE Symp. on Foundations of Comp. Sci.*, pages 156–165, 2008. 3.2.1, 3.2.1, 5.2.2, 5.3.1, 5.4, 5.4, 5.5

[116] Elchanan Mossel, Ryan O'Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. In *Proc. 46th IEEE Symp. on Foundations of Comp. Sci.*, pages 21–30, 2005. To appear, *Annals of Mathematics*. 3.2.1, 4.1.8, 4.2.4, 4.7, 4.8, 4.8.1, 4.8.1, 5.2.2, 5.3.2

[117] Elchannan Mossel, Ryan O'Donnell, and Krzysztof Oleszkiewicz. Noise stability of

functions with low influences: Invariance and optimality. In *FOCS: IEEE Symposium on Foundations of Computer Science (FOCS)*, 2005. 9.3.2, 9.9.1

[118] R. O'Donnell and R. Servedio. Learning monotone decision trees in polynomial time. *SIAM J. Comput.*, 37(3):827–844, 2007. 10.1.1

[119] Ryan O'Donnell and Rocco A. Servedio. The chow parameters problem. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 517–526, New York, NY, USA, 2008. ACM. 9.2, 9.3.1

[120] Ryan O'Donnell and Yi Wu. An optimal SDP algorithm for Max-Cut, and equally optimal Long Code tests. In *Proc. 40th ACM Symp. on the Theory of Computing*, pages 335–344, 2008. 1.2.1, 2.5.1

[121] Ryan O'Donnell and Yi Wu. 3-bit dictator testing: 1 vs. 5/8. In *SODA '09: Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms*, pages 365–373, 2009. 5.2.2, 5.2.3

[122] Michal Parnas, Dana Ron, and Alex Samorodnitsky. Testing basic boolean formulae. *SIAM Journal on Discrete Mathematics*, 16(1):20–46, 2002. 2.6.1

[123] Svatopluk Poljak and Franz Rendl. Nonpolyhedral relaxations of graph-bisection problems. *SIAM Journal on Optimization*, 5(3):467–487, 1995. 1.2.1, 4.1.2

[124] Y. Rabani and A. Shpilka. Explicit construction of a small epsilon-net for linear threshold functions. In *Proc. 41st Annual ACM Symposium on Theory of Computing (STOC)*, pages 649–658, 2009. 10.1.1

[125] Prasad Raghavendra. Optimal algorithms and inapproximability results for every CSP? In *Proc. 40th ACM Symp. on the Theory of Computing*, pages 245–254, 2008. 1.2.1, 4.1.9, 11

[126] Prasad Raghavendra. *Approximating NP-hard problems: efficient algorithms and their limits*. PhD thesis, University of Washington, 2009. 3.2.1

[127] Prasad Raghavendra and David Steurer. How to round any CSP? In *Proc. 50th IEEE Symp. on Foundations of Comp. Sci.*, 2009. To appear. 4.1.9, 6.1.3, 6.1.4, 6.7

[128] Ran Raz. A parallel repetition theorem. *SIAM Journal on Computing*, 27(3):763–803, 1998. 2.5, 10.3.2

[129] Sartaj Sahni and Teofilo Gonzales. P-Complete approximation problems. *Journal of the ACM*, 23:555–565, 1976. 4.1

[130] Alex Samorodnitsky and Luca Trevisan. A PCP characterization of NP with optimal amortized query complexity. In *Proc. 32nd ACM Symposium on Theory of Computing*, pages 191–199, 2000. 5.1.1

[131] G. Schoenebeck. Linear level Lasserre lower bounds for certain k-CSPs. pages 593–602, 2008. 6.2

[132] R. Servedio. Every linear threshold function has a low-weight approximator. *Computational Complexity*, 16(2):180–209, 2007. 10.1.1

[133] Rocco A. Servedio. Every linear threshold function has a low-weight approximator. *Comput. Complex.*, 16(2):180–209, 2007. 9.2, 9.3.1

[134] Daniel Stefankovic. Fourier transforms in computer science. Master's thesis, University of Chicago, 2002. 5.3

[135] Michel Talagrand. On Russo's approximate zero-one law. *Annals of Probability*, 22(3):1576–1587, 1994. 6.3.2

[136] Suguru Tamaki and Yuichi Yoshida. A query efficient non-adaptive long code test with perfect completeness. *ECCC*, TR09-074, 2009. 5.2.3

[137] L. Tang. Conditional hardness of approximating satisfiable Max 3CSP-q. In *Proc. 20th ISAAC*, pages 923–932, 2009. 2.5.1, 5.2.3

[138] Luca Trevisan, Gregory Sorkin, Madhu Sudan, and David Williamson. Gadgets, approximation, and linear programming. *SIAM Journal on Computing*, 29(6):2074–2097, 2000. 1.2.1, 4.1.8

[139] Madhur Tulsiani. CSP gaps and reductions in the Lasserre hierarchy. pages 303–312, 2009. 6.2

[140] Leslie Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984. 1.2.2, 2.4.2

[141] Uri Zwick. Approximation algorithms for constraint satisfaction problems involving at most three variables per constraint. In *Proc. 9th ACM-SIAM Symp. on Discrete Algorithms*, pages 201–210, 1998. 1.2.1, 1.3.2, 5.1.1, 5.1.2

[142] Uri Zwick. Outward rotations: A tool for rounding solutions of semidefinite programming relaxations, with applications to MAX CUT and other problems. In *Proc. 31st ACM Symp. on the Theory of Computing*, pages 679–687, 1999. 1.2.1, 4.1, 4.1.3, 4.1.5, 4.10, 4.11, 4.11

[143] Uri Zwick. Computer assisted proof of optimal approximability results. In *Proc. 13th ACM-SIAM Symp. on Discrete Algorithms*, pages 496–505, 2002. 1.2.1