Tel Aviv University
Raymond and Beverly Sackler
Faculty of Exact Sciences
School of Computer Sciences

Property Testing

PCP

and

Juntas

Thesis submitted for the degree of “Doctor of Philosophy” by

Guy Kindler

under the supervision of prof. **Shmuel Safra**

Submitted to the Senate of Tel-Aviv University
October 2002

ISSN 1433-8092

To my father, to whom I owe my love of thought.

To my mother, for her overwhelming love.

And to Michal, for tolerating me for so long.

Acknowledgments

I would like to express my deepest gratitude to my thesis advisor, Prof. Muli Safra, who never gave up on me, even at rough times. I have learned a great deal from his enthusiasm for every aspect of life and knowledge, and from his un-orthodox views on math, the universe, and everything.

I wish to thank Eldar Fischer, who co-authored much of the work herein, and spent many hours reading and correcting earlier versions of it. I would also like to thank my other collaborators on parts of this work, Irit Dinur, Ran Raz, Dana Ron, and Alex Samorodnitsky.

Abstract

Part I

The first part of this thesis strengthens the low-error PCP characterization of NP, coming closer to the upper limit of the conjecture of [BGLR93]. Consider the task of verifying a witness for the membership of a given input in an NP language, using a constant number of accesses. If the witness is given as a string of characters, we show that it is possible to acheive this task with an error probability that is polynomially small in the range of the characters, where the size of that range is as high as $2^{\log^\beta n}$, for *any* constant $\beta < 1$. The BGLR conjecture asserts the same for a constant β where $\beta \leq 1$.

Our results are in fact stronger, implying that the Gap-Quadratic-Solvability problem with a constant number of variables in each equation is NP-hard. That is, given a system of n quadratic-equations over a field $\mathcal{F}$ of size up to $2^{\log^\beta n}$, where each equation depends on a constant number of variables, it is NP-hard to distinguish between the case where there is a common solution to all of the equations, and the case where any assignment satisfies at most a $\frac{2}{|\mathcal{F}|}$ fraction of them.

At the same time, our proof presents a *direct* construction of a low-degree-test whose error-probability is polynomially small in the range of the variables accessed. Such a result was previously known only relying on recursive applications of the entire PCP theorem.

Part II

In the second part of the theis we show that a boolean function over n variables can be tested for the property of depending on only J of them, using a number of queries that depends only on J and the approximation parameter ϵ.

We present three tests, that require a number of queries that is polynomial in J and linear in ϵ^{-1}. We show a non-adaptive test that has one-sided error, an adaptive version of it that requires less queries, and a non-adaptive two-sided version of the test that requires the least number of queries.

We then provide a lower bound of $\tilde{\Omega}(\sqrt{J})$ on the number of queries required for the non-adaptive testing of the above property; a lower bound of $\Omega(\log(J+1))$ for adaptive algorithms naturally follows from this. In providing this we also prove a result about random walks on the group $\mathbb{Z}_2^q$ that may be interesting in its own right. We show that for some $t(q) = \tilde{O}\left(q^2\right)$, the distributions of the random walk at times t and $t+2$ are close to each other, independently of the step distribution of the walk.

We also discuss a related question, as follows. When given in advance a known J-junta function h, we show how to test whether h can be obtained from a given function f by a permutation on its variables, using a number of queries that is polynomial in J and ϵ.

Part III

In the third part of this thesis we show that any boolean function $\mathsf{f}\colon \{0,1\}^n \rightarrow \{-1,1\}$ whose weight on Fourier-Walsh products of size larger than k is bounded by $(\epsilon/k)^{(\ell+1)/\ell}$, where ℓ is any

fixed positive integer, is $O(\epsilon)$-close to a junta whose size is independent of n. As a corollary, we obtain that juntas are the only highly noise-resistant boolean functions. These results are proven with respect to the p-biased distribution over the discrete cube, extending a result by Bourgain that has somewhat better parameters but holds only for the uniform measure. Out method of proof is different than that of Bourgain, thereby providing a conceptually simpler proof for the uniform case as well.

We also show that for small values of ϵ, if the weight of a boolean function f on Fourier-Walsh products of size larger than k is ϵ, then f is $\epsilon(1+o(1))$-close to a junta, whose size depends only on k. This is a generalization of the result in [FKN01], who proved this for $k = 1$, and only with respect to the uniform measure.

Contents

Introduction

Property testing deals with the following task: For a fixed property $\mathcal{P}$ and any given input I, one has to distinguish with high probability between the case where I satisfies $\mathcal{P}$ and the case where I is 'far' from satisfying it. The goal in property testing is to make this distinction using the least possible number of accesses to I.

The notion of property testing was first formulated by Rubinfeld and Sudan [RS92], who were motivated mainly by its connection to the study of program checking. They considered the problem of verifying that a given function, given as a table of its values, is a polynomial of low-degree. Perhaps the most fundamental property for which efficient tests are applicable, is the correctness of mathematical proofs. The study of proof testing (called PCP, for Probabilistic Checkable Proofs), which originated in [FRS88, AS98], turned out to have numerous applications in complexity and hardness of approximation, in addition to its theoretical appeal.

The notion of property testing was extended for other combinatorial objects, mainly for graphs, by Goldreich, Goldwasser and Ron [GGR98], and has become a very active area of research. Boolean functions were also given much attention from the point of view of property testing [GGL+00, DGL+99, FLN+02, PRS01]. For a more comprehensive description of property testing and its applications, the reader is referred to the surveys [Ron00] and [Fis01].

Part I, PCP

The first part of this thesis deals with proof testing, namely PCP. The framework considered is as follows. A mathematical statement S is given, and P is an alleged proof for it, given as a sequence of bits. One wishes to test the validity of P by accessing only a constant number of bits from it. Remarkably, it was shown in [ALM+98] (improving the result of [AS98]) that if valid proofs are required to be encoded in a certain way, this is indeed possible.

Let us describe the above result a bit more formally. It was actually shown in [ALM+98] that given the statement S and the length of the alleged proof P, one can construct in polynomial time a system Ψ of *local-tests*, that verifies the correctness of P as follows. Each local-test in Ψ is a boolean function, which depends on a constant number of bits from the alleged proof, P. The local-tests of Ψ are all satisfied if P is a legal encoding of a correct proof for S. However if the initial statement S is incorrect, it is assured that no proof, may it be properly or improperly encoded, can satisfy more than an ϵ-fraction of the local-test, where ϵ is some positive constant.

It follows that one can test P by evaluating a random local-test ψ from Ψ. If S is correct, and P is a legal encoding of a correct proof for it, then ψ is surely satisfied. If S is incorrect, the probability that ψ be satisfied 'by mistake' is always bounded by the parameter ϵ, also called the *error parameter* of Ψ.

PCP and In-Approximability

One can consider the result of [ALM$^+$98] as a reduction from the problem of determining whether a given mathematical statement S has a proof of a given length, to the problem of distinguishing between the case where Ψ is completely satisfiable and the case where it cannot be more than ϵ-satisfied. Since the first problem above is NP-complete (supposing the required length of the proof is given in unary format), this implies that given a system Ψ of local-tests, it is NP-hard to approximate the maximal number of satisfiable local-tests in it within a factor of $1/\epsilon$. So in fact the result of [ALM$^+$98] implies that unless $NP = P$, given a set of local-constraints over boolean variables, it is intractable to even approximately maximize the number of satisfied constraints by an assignment to the variables.

The introduction of probabilistically checkable proofs led to a swarm of in-approximability results ([FGL$^+$91, ALM$^+$98, LY94, BGLR93, BGS98, Hås99, Hås97, DKRS98], to mention a few), obtained by introducing PCP's with special properties and parameters, or by applying appropriate reductions to known PCP systems.

The Sliding-Scale Conjecture

One version of a PCP system, first considered in [BGLR93], is where the proof P is not given as a sequence of bits, but rather as a sequence of characters from a larger set $\mathcal{R}$ of non-constant size. It was conjectured in [BGLR93] that it is possible to construct a PCP system where the size of $\mathcal{R}$ is up to polynomial in the required length of P, and the error parameter is polynomially small in the size of $\mathcal{R}$. This was conjectured to hold even if each local-test in the system still accesses a constant number of characters in P.

Optimality of the conjecture. Note that it is not possible to decrease the error parameter below some polynomial in the size of $\mathcal{R}$, namely there is no way to ensure that if S is incorrect, the number of satisfiable local-tests is smaller than some polynomial in $\mathcal{R}$: assuming without loss of generality that each local-test is satisfiable in its own, the probability that a local-test is satisfied by taking P to be a random sequence of characters is bounded below by some fixed polynomial in $|\mathcal{R}|$. Using the linearity of expectation, one obtains that for *every* system of satisfiable local-tests there exists an alleged proof satisfying some fixed polynomial fraction of its local-tests.

In addition, it is unlikely that the size of $\mathcal{R}$ can be made to exceed $n^{O(1)}$, where n is the required length of the proof, while keeping the error parameter polynomially small in $|\mathcal{R}|$. This is since the system Ψ, which is constructed in polynomial time, can only contain a polynomial number of local-tests. If the size of $\mathcal{R}$ exceeds $n^{O(1)}$ and the error parameter, ϵ, is polynomially small in it, then ϵ must be smaller than $1/|\Psi|$. It follows that when the initial statement S is false, no alleged proof can satisfy even one of the local-tests. Distinguishing between the case where S can be given a short proof and the case where S is false thus reduces to the problem of deciding whether any of the local-tests in Ψ can be satisfied at all. This leads immediately to the conclusion that $NP = P$.

Our results. The first part of this thesis presents a proof for the sliding scale conjecture, applicable for range-sizes of up to $2^{\log^{\beta} n}$, where β can be taken to be *any* constant smaller than one. This result comes close to the limit of the conjecture, namely polynomially-sized range. It is an improvement of the result in [RS97], which only obtained ranges of size $2^{\log^{\beta} n}$ for some fixed β which is separated away from 1.

In fact, the result presented here is somewhat stronger. The conjecture is proven for the aforementioned range, where the characters of the proof are elements of a finite field $\mathcal{F}$, and each local-test it in fact a quadratic-equation over $\mathcal{F}$. Specifically, we prove that for a quadratic equation-system of n equations over a field $\mathcal{F}$ of size $2^{\log^{\beta} n}$ (for any constant $\beta < 1$), where each equation depends on a constant number of variables, it is NP-hard to distinguish between the case where there exists a common solution to all equations, and the where any assignment to the variables satisfies no more than a $\frac{2}{|\mathcal{F}|}$ fraction of them.

Low-Degree Tests

A crucial part of the proof presented here, as well as of other PCP results, is the construction of a test for low-degree polynomial functions (low-degree functions for short).

Roughly speaking, a test for an alleged proof must accomplish two tasks. First it should verify that the alleged proof is well formatted, namely that it is a legal encoding of some, not necessarily correct, proof. Assuming that the alleged proof is indeed a legal encoding, the second task is to test that it is the encoding of a correct proof for the given statement. In many PCP constructions, as well in the one herein, the encoding of the proof makes use of low-degree polynomial functions. In such constructions it is therefore necessary to construct a test for the property of being a low-degree polynomial function.

It is worth mentioning that the approach of encoding a proof using low-degree polynomial functions already appears in [BFL91], where interactive-proof protocols are shown for non-deterministic exponential time. There, a proof is encoded using a low-degree polynomial, and to verify its correctness the proof is first tested for being a polynomial of low-degree, and then it is tested for being correct.

To better explain our low-degree test, take $\mathcal{F}$ to be a finite field and let $f : \mathcal{F}^d \rightarrow \mathcal{F}$

be some function, given as a table of its values. Generally, a low-degree test is a random procedure that is given f, and perhaps some additional auxiliary variables taking values in $\mathcal{F}$. Its goal is to test whether f is close to a low-degree polynomial function (a function of degree up to, say, $\sqrt{|\mathcal{F}|}$). Previous direct constructions could only obtain low-degree tests by evaluating f at a non-constant number of points. Better parameters could only be achieved by a technique called composition, taking a PCP construction and composing it over another.

We present a direct construction of a low-degree test, based on the low-degree test of [RS97], that makes a constant number of queries to f and to the auxiliary variables. This is achieved, in part, by some relaxation of the requirements, following [RS97]. Instead of requiring that f corresponds to one low-degree function or else the low-degree test rejects, we allow the low-degree test to accept in case f corresponds to a short list of low-degree functions, the *permissible functions*.

Permissible functions. We say that a low-degree function $g : \mathcal{F}^d \to \mathcal{F}$ is ρ-permissible with respect to f, if it agrees with f on at least ρ-fraction of the points in $\mathcal{F}^d$.

Our low-degree test is guaranteed to reject with high probability, unless all the values of f that it queries are consistent with one of the ρ-permissible low-degree functions with respect to f, where ρ is some non-negligible parameter. For parameters ρ in the range used, it can be shown that there are never more than $O(\rho^{-1})$ permissible functions *for any* given f. Hence although our low-degree test does not ensure that f is close to one low-degree function, when it accepts the evaluations it makes of f are with high probability consistent with one of a *short list* of low-degree functions. It turns out that this is enough for our PCP construction.

Part II, Testing Juntas

The principle discussed above, of checking consistency by verifying non-negligible agreement with one of a short list of legal encodings, appears in many PCP constructions. In many of them (e.g. [Hås99, Hås97, Kho02]) this approach is applied to the so called long-code.

The long-code. The long-code is used to encode elements i within the set $\{1, \ldots, n\}$. It has one entry for every element $x \in \{0,1\}^n$ of the n-dimensional discrete hyper-cube, which, in the encoding of i, contains x_i. The encoding of i is thus the truth table of the function $\mathsf{LC}_i : \{0,1\}^n \to \{0,1\}$, defined by $\mathsf{LC}_i(x) = x_i$. Note that the function LC_i depends, in fact, only on the i'th coordinate of its input.

Suppose we wish to test whether a given codeword, represented as the truth table of a function $\mathsf{f} : \{0,1\}^n \to \{0,1\}$, corresponds to a short list of legal encodings. This can be interpreted in more than one way.

Since in a legal encoding there is only one coordinate which affects the value of f, one may test that there is a short list of coordinates that have 'high influence' on f. The tests in [Hås99] and [Hås97] fall in that category. For a codeword f to pass these tests, there must be a small number of coordinates, each of which highly influences f. It may be, however, that f passes these tests although there is a large set of coordinates whose individual influence on f might be small, but whose aggregate influence on f is very high.

The above approach is not always sufficient. For example, in the constructions of [DS02] and [Kho02] it is necessary to verify that almost all the values of f depend on a short list of coordinates. In other words, it is necessary to test whether f is close to a junta.

Juntas. A boolean function over $\{0,1\}^n$ is said to be a J-junta, if it depends on at most J coordinates. The term junta originates from considerations related to social choice, where a boolean function f is seen as an election scheme, and each coordinate in its domain corresponds to a voter. An element $x \in \{0,1\}^n$ represents an election where $x_i = 1$ if the i'th voter votes 'yes' and $x_i = 0$ otherwise, and $\mathsf{f}(x)$ is the outcome of the election. In these terms, a J-junta is an election scheme the outcome of which is completely determined by at most J voters.

Our Results

The second part of this thesis considers the problem of testing whether a given boolean function f is a J-junta. We show several tests that are given access to the truth table of f, and distinguish between the case where f is a J-junta, and the case where the values of f must be changed on at least an ϵ-fraction of the inputs for it to become a J-junta. The number of queries these tests make is independent of the number of coordinates of f, and their dependency on ϵ and J is polynomial.

The first test shown makes $O(J^4 \ln(J+1)/\epsilon)$ queries to the given function f, and always accepts if it is a J-junta. On the other hand, if f is more than ϵ-far from being a J-junta, the test rejects with probability at least $1/2$. This test is non-adaptive, namely it decides on which inputs to query f independently of the results of previous queries.

Variation. The main observation utilized by our junta-test, is that there is a very simple way to measure the effect of a set I of coordinates on the values of a given boolean function $\mathsf{f} : \{0,1\}^n \to \{0,1\}$. This is an extension of the *influence*, defined in [BL89, KKL88], which measure the influence of one coordinate on f. The *variation* of f on a set I of coordinates is twice the probability that f yields different values when evaluated on two random inputs that differ only on coordinates from I.

Note that it is very easy to test whether the variation of f on I is small, by just selecting two inputs randomly as above and evaluating f on them. In fact, this procedure actually tests whether f depends on coordinates from I: it turns out that if the variation of f on I is smaller than ϵ (namely with high probability f yields the same values for both inputs queried), then it is possible to change less than $(\epsilon/2)$-fraction of the values of f and obtain a function that is completely independent of the coordinates in I.

Other tests. The junta-test mentioned above is in fact somewhat more general. By extending the notion of variation to general products of probability spaces, we can apply the same test for boolean functions whose variables are not necessarily boolean, but rather take values in general probability spaces. The number of queries made by the test remains the same, and is independent of the domain of the function being tested. In these settings we also show an adaptive variant of the junta-test that makes $O(J^3 \ln^2(J+1)/\epsilon)$ queries, and a version that is non-adaptive, but may reject a J-junta with probability up to 1/3. The latter test makes only $O(J^2 \ln^2(J+1)/\epsilon)$ queries.

Another test, shown using the same techniques, uses a polynomial (in J and ϵ) number of queries, and verifies whether a fixed J-junta h can be obtained from a given function f by a permutation of its variables. This test is shown only for the case of boolean variables, and it has two-sided error, namely it rejects with probability 2/3 if h is ϵ-far from every variable-permutation of f, and it accepts with probability at least 2/3 if h can be obtained by a variable-permutation of f.

Lower-bound. In addition, we show a lower-bound for the number of queries made by non-adaptive tests for J-juntas. It is shown that such a test must make at least $\Omega(\sqrt{J})$ queries (up to logarithmic factors). Recently a better lower-bound was achieved by Chockler and Gutfreund ([CG02]), that also holds for adaptive tests. However, our proof for the lower bound may be of independent interest, as it relies on an interesting convergence of random walks on weighted Cayley graphs of $\mathbb{Z}_2^n$. While it may take a long time before a random walk on such a graph becomes stationary, we prove that the distribution on the graph after a walk of length t must be close the distribution after a walk of length $t+2$, already for relatively small values of t. This bound is independent of the choice of weights on the set of generators.

Part III, Noise-Resistant Boolean Functions are Juntas

The tests discussed in Part II exploit properties of juntas that are easy to check using few queries, such as the behavior of the variation on certain subsets of coordinates. However, the number of queries made by these tests is too high for many applications (it depends on J). For applications to PCP and hardness of approximation, one is often willing to compromise certain properties of a test, as long as it achieves an extremely small number of queries. It is thus natural to seek for properties which characterize juntas, and yet are easily testable by a few queries. The noise-sensitivity is such a property.

Noise-sensitivity. The noise-sensitivity of a boolean function $\mathsf{f} : \{0,1\}^n \rightarrow \{0,1\}$ measures the probability that the value of f changes when noise is applied to a random input for f. To state this more formally, fix p, λ to be two positive parameters, $p, \lambda < 1$. Now let x be a random input for f, chosen according to the p-biased distribution, namely by setting each coordinate to be 1 with probability p and 0 with probability $1 - p$. To apply noise to x, first choose a 'noisy subset' $I \subseteq [n]$ of coordinates, taking each coordinate into I with probability λ. Now let x' be obtained from x by re-selecting each coordinate $i \in I$ according to the p-biased distribution. The probability that f yields different values on x and x' is called the λ-noise-sensitivity of f with respect to the p-biased distribution.

Note that for an appropriate choice of λ, a J-junta f must have small λ-noise-sensitivity. This holds since there is a small probability that any of the variables that f depends on are taken into the noisy subset. In addition, it is possible to test whether a given function f has small noise-sensitivity using just two queries, by randomly choosing x and x' as above, and querying $\mathsf{f}(x)$ and $\mathsf{f}(x')$.

The notion of noise-sensitivity makes sense in terms of social choice as well. If f is regarded as an election scheme, and each voter casts his vote randomly with bias p, the noise-sensitivity of f measures the stability of the outcome of the election scheme, with respect to certain faults or changes of opinion of some voters.

Our Results

In the third part of this thesis we show that a boolean function whose λ-noise-sensitivity with respect to p-biased distribution is smaller than a certain threshold, must be close to a J-junta for some constant J. To be precise, we show that if $\mathsf{f} : \{0,1\}^n \rightarrow \{0,1\}$ has small λ-noise-sensitivity with respect to p-biased distribution, then there exists a J-junta h, such $\mathsf{f}(x) = \mathsf{h}(x)$ with high probability (if x is chosen according to the p-biased distribution). In terms of social choice, this means that juntas are the only election schemes which are resilient against noise.

Our result is a generalization of a theorem of Bourgain ([Bou01]), which holds only for the case where $p = 1/2$. The technique of [Bou01], though achieving somewhat better parameters than ours, does not immediately generalize to the p-biased case, since it relies on certain inequalities whose correctness for the p-biased case is unclear. Our result is thus proven using a different technique, that is conceptually simpler than that of Bourgain, and may be of independent importance.

The importance of the bias. We hope that the generalization of the result from [Bou01] to the case of biased measure will be useful in complexity theory, not only due to the techniques of its proof. It seems that applying biased measures on entries of codes, and especially the long-code, brings out combinatorial properties that are not present with respect to the uniform distribution. Such properties are crucial, for example, in [DS02], where the Vertex-Cover problem is shown to be hard to approximate within a 1.361 factor. The proof of [DS02] utilizes the p-biased measure on $\{0,1\}^n$, where p is in the vicinity of 1/3. By allowing some leverage on the choice of p, they show that certain boolean functions resulting from their construction must be close to juntas. Other combinatorial properties of their construction, related to intersecting families, are obtained from the fact that the bias is separated away from 1/2.

Noise-sensitivity and Fourier expansion. Let $\mathcal{V}$ denote the space of real-valued functions over $\{0,1\}^n$. By viewing a boolean function $\mathsf{f} : \{0,1\}^n \to \{0,1\}$ as an element of $\mathcal{V}$, one can obtain f as a linear combination of the elements of some basis for $\mathcal{V}$, such as the *Fourier-Walsh basis.* This well-known basis, which was first used to analyze boolean functions in [KKL88], contains one function χ_S for every set $S \subseteq [n]$. The Fourier-Walsh basis is orthonormal with respect to the natural inner-product that corresponds to the uniform measure on $\{0,1\}^n$. A similar basis appears in [Tal94], which is orthonormal with respect to the inner-product that corresponds to the p-biased measure. Hence fixing p, any boolean function f can be written in the form

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

where $\{\chi_S\}_{S\subseteq[n]}$ is the appropriate orthonormal basis.

It turns out that there is a simple formula, connecting the λ-noise-sensitivity of f with respect to μ_p, to the coefficients $\widehat{\mathsf{f}}(S)$. It follows from the formula that in order for the λ-noise-sensitivity of f to be small, the coefficients $\widehat{\mathsf{f}}(S)$ related to large sets S must be small as well. More precisely, the formula shows that if the λ-noise-sensitivity of f is small then for an appropriate number k, $\sum_{|S|>k} \widehat{\mathsf{f}}(S)^2$ must be small as well. We denote the latter sum by $\|\mathsf{f}^{>k}\|_2^2$.

Asymptotic behavior. In showing that if f has small λ-noise-sensitivity it must be close to a junta, it is actually proven that it is close to a junta if $\|\mathsf{f}^{>k}\|_2^2$ is smaller than a certain threshold. This naturally gives rise to the question of how the distance of boolean functions f from a junta behaves, as a function of $\|\mathsf{f}^{>k}\|_2^2$.

This question is discussed in [FKN01] for the case $k = 1$, where it is shown that the distance from a junta behaves linearly as a function $\|\mathsf{f}^{>1}\|_2^2$. We extend this result, showing that the distance of f from a junta that is dominated by a constant number of coordinates, is bounded by $(4 + o(1))\|\mathsf{f}^{>k}\|_2^2$ for small values of $\|\mathsf{f}^{>k}\|_2^2$. This bound is tight up to a constant factor (note that for convenience, the result is formulated in Part III for functions that take values in $\{-1,1\}$ rather than $\{0,1\}$, hence the constant which appears there is 1 and not 4).

Collaborators

This first part of this thesis is based on the paper [DFK$^+$99], written in collaboration with Irit Dinur, Eldar Fischer, Ran Raz, and Muli Safra. The second part is based on [FKR$^+$], written in collaboration with Eldar Fischer, Dana Ron, Muli Safra, and Alex Samorodnitsky. The third part of this thesis is based on [KS02], written with Muli Safra.

Part I

PCP Characterizations of NP: Towards a Polynomially-Small Error-Probability

Chapter 1

Introduction to Part I

Cook-Levin's characterization of NP implies that every $L \in$ NP is reducible to 3-SAT. The reduction from L to 3-SAT is a polynomial-time algorithm that receives an input string I, and produces a set Ψ of boolean functions (called local-tests), each depending on a constant number of variables. Ψ represents the membership of I in L, in the sense that there exists an assignment satisfying all local-tests if and only if $I \in L$.

A PCP characterization of NP differs from Cook-Levin's characterization in regards to what is guaranteed in the case where the input is not in L: In Cook's characterization, one can only be sure that the reduction will produce a system that cannot be entirely satisfied. To characterize NP in terms of PCP, it must be guaranteed that the reduction algorithm produces a system Ψ such that no assignment can satisfy even a small fraction ϵ of its local-tests.

In both cases, a satisfying assignment to Ψ can be viewed as a witness for I's membership in L (and hence Ψ can be viewed as a membership-verification system). In a PCP framework, however, this witness can be efficiently verified by randomly picking a local-test of Ψ and verifying that it holds (hence the term PCP – Probabilistic Checking of Proofs). In this case, the *error probability* parameter, ϵ, of the PCP, bounds the probability of accepting I even though $I \notin L$. Other parameters of Ψ, such as the variable range and the number of variables accessed by each local-test, are also part of the PCP characterization.

For many applications of PCP, the characterization of NP with a constant error-probability and variables of a constant range [AS98, ALM$^+$98] suffices. In order to prove NP-hardness of other problems, however, sub-constant error-probability has turned out to be essential. For example, [LY94] and [BGLR93] were able to prove that approximating SET-COVER to within logarithmic factors is *almost* NP-hard, using the constant error-probability PCP characterization of NP. To improve this result to strict NP-hardness, [BGLR93] had suggested the "sliding scale" conjecture.

The sliding scale conjecture states that it is possible to keep the number of variables accessed by each local-reader constant, and to make the variables' range non-constant, obtaining an error probability polynomially small in the size of the variable-range. In other words, it is possible to acheive a membership verification system for any NP-language where each local-test accesses a constant number of 'words' (variables), and where the error-probability is exponentially small in the 'word-length' (number of bits in each variable).

One cannot expect the error-probability to be more than polynomially small in the size of the variables' range, since a random assignment will satisfy any satisfiable local-test with such a probability (recall that each test depends on a constant number of variables). Hence the sliding scale conjecture is optimal in the sense of error-probability.

According to the conjecture, the variables' range may be increased up to a size polynomial in the length of the original input (note that each local-test can be given as a truth-table). Reaching larger range-size is unlikely since the error-probability would then become less than $1/|\Psi|$. In the case where the input is not in L, this implies that no local-test succeeds, so the problem of deciding whether the input is in L reduces to that of deciding whether any of the local-tests is satisfiable.

The sliding scale conjecture was shown to hold for a sizable portion of the applicable range-size in [RS97], where a PCP characterization of NP was shown that achieves error-probability polynomially small in the size of the variable range for a variable range of size up to $2^{\log^\beta n}$, where $\beta < 1$ is a certain positive constant (see also [AS97]).

Our Main Results

In this part of the thesis, we prove the sliding scale conjecture for variable range sizes of up to $2^{\log^\beta n}$ where β is *any* constant smaller than one (as opposed to "*some* constant" achieved by [RS97]), thus coming closer to proving the sliding scale conjecture for the full applicable range.

In fact our result is somewhat stronger, proving the conjecture for the aforementioned range using proof verification systems of a specific structure. In these systems the local-tests have the form of quadratic-equations instead of being general boolean functions, with the variables' range representing a finite field. This result implies that for a quadratic equation-system of n equations over a field $\mathcal{F}$ (with $|\mathcal{F}| \approx 2^{\log^\beta n}$ for any fixed constant $\beta < 1$), where each equation depends on a constant number of variables, it is NP-hard to decide whether there exists a common solution to all equations, or whether any assignment to the variables satisfies no more than a $\frac{2}{|\mathcal{F}|}$ fraction of them.

One of the main tools used to obtain the above result, which is interesting in its own right, is that of LDF-readers, a version of what is known in the literature as a low-degree test (see [RS97, AS97]). A direct construction of an LDF-reader is shown herein, that achieves an exponentially-small error-probability with respect to the number of bits it accesseses. Such LDF-readers could previously be attained only by recursive applications of the entire PCP theorem.

Related Results

We note that there is no known PCP characterization of NP, where the size of the variable-range is polynomial in the size of the membership-verification system (or equivalently, the length of each variable is logarithmic in it), and the error probability is exponentially small in the number of accessed bits. This is true even when allowing a super-polynomial time reduction. The repetition lemma of [Raz98] shows that by two accesses to $\Theta(\log n)$ bits, the error-probability can be made polynomially small in n, where n is the size of the original input, while the size of the generated system is $n^{\log n}$. Similarly, the multi-linear extension of [BFL91] yields a system with a $\frac{1}{n}$ error-probability, whose size is $n^{\log n}$. In fact, in any known reduction there is always a factor of at least $\log^{\epsilon} n$ in the exponent that separates the error-probability from the size of the generated instance.

Achieving an error-probability polynomially small in the size of the generated instance is an important open problem. Such a characterization of NP would improve hardness results for several problems. For example, approximating the 'Monotone-Minimum-Satisfying-Assignment' problem (which is closely related to approximating the length of propositional proofs [ABMP98]) has been shown to be NP-hard in [DS98] via a reduction from PCP, such that the hardness of approximation ratio is preserved. Hence a polynomially small error-probability PCP characterization of NP would immediately imply that it is NP-hard to approximate the length of propositional proofs to within an n^{ϵ} factor for some constant $\epsilon > 0$.

[RS97] managed to keep the exponential relation between the number of bits accessed and the error-probability, thus showing the sliding scale conjecture true for a variable range of size up to $2^{\log^{\beta} n}$ for *some* constant $\beta < 1$. For larger β (any constant $\beta < 1$) [RS97] showed a system whose error probability is $2^{-\log^{\beta} n}$, yet without the exponential relation between the number of accessed bits and the error-probability, since the number of bits accessed was $O(\log^{\beta} n \cdot \text{poly} \log \log n)$. This factor of $\text{poly} \log \log n$ is significant when viewing, for example, the result in terms of Gap-Quadratic-Solvability. The result of [RS97], if it were to be translated to Gap-Quadratic-Solvability terms, would at best give an equation system with each equation depending on $O(\text{poly} \log \log n)$ variables. In comparison, our result translates to a quadratic equation-system with the same error-probability, but where every equation depends on a *constant* number of variables, namely $\Theta(\frac{1}{(1-\beta)^2})$.

Techniques

We use the general framework of [AS98, ALM$^+$98, RS97] for our proof. However, instead of the generalized form of the composition paradigm utilized in previous PCP proofs, we use a more concrete representation. Our result could have been obtained using the previous structure, but this representation simplifies our proof, and some of its techniques may be of independent interest.

In [HPS93], it was shown that given a system of quadratic equations over a finite field, it is NP-hard to distinguish between the case that the system can be completely satisfied, and the case that not even a small fraction of the equations can be satisfied by a single assignment. The crucial difference between this and our main result is that in the [HPS93] reduction each equation depends on almost all the variables in the system, while our main result claims the same for the case where the equations are restricted to having a constant number of variables each.

Our proof begins with a system Ψ of quadratic equations as in [HPS93], and reduces it to a system Ψ' of quadratic equations with a constant number of variables in each. The key property of our proof is that throughout the reduction we use systems of equations over the same field $\mathcal{F}$, the field over which Ψ is defined. The field structure is utilized through various steps of composition, thus enabling us to cross the barrier that limits the proof technique of [RS97].

To simplify the exposition, the reduction partitions the variables of Ψ' into subsets called domains. In each such domain a mapping is defined, associating each variable to a point in a linear space of the form $\mathcal{F}^d$ over $\mathcal{F}$. An assignment to these variables can thus be regarded as a function over the linear space.

The reduction has two main steps. At first, it transforms Ψ into a system Ψ_{sc} where the number of variables in each equation is constant. This is accomplished by an iterative application of the sum-check technique from [BFL91]. The system Ψ_{sc} has the required properties only if the assignment to the variables in each domain, when viewed as a function, is a low-degree polynomial. In order to get rid of this restriction, the reduction then generates LDF-readers and plugs them into the equations of Ψ_{sc}, thereby obtaining the final system Ψ'.

LDF-readers. LDF-readers are used to obtain evaluations of polynomial functions of low-degree that are represented by a set of variables, by accessing only a very small part of their representation. An LDF-reader should either reject or return values that are consistent with some low-degree polynomial, even if the assignment to the representation variables is not totally consistent with the representation of one polynomial. The probability that, given an incorrect representation, the LDF-reader does not reject but still the returned evaluations are not consistent with a low-degree polynomial, is its error probability. For a more accurate definition of an LDF-reader, the reader is referred to Chapter 2.

An LDF-reader of sub-constant error-probability seems necessary in order to attain PCP characterizations of NP with sub-constant error-probability. The plane-vs.-plane LDF-reader introduced by [RS97], where a polynomial is represented by its restriction to planes, achieves a sub-constant probability. The previously used line-vs.-point LDF-reader was shown by [AS97] to have a small error-probability as well. However, the error probability of these LDF-readers is not exponentially small in the number of bits they access.

In fact, in any direct LDF-reader comparing subspaces (lines, planes, etc.) for consistency, the error-probability can be no smaller than a polynomial in the number of bits accessed. This occurs since many bits are required to represent the restriction of a polynomial to a subspace. One way to attain exponentially small error-probability from these LDF-readers is by utilizing the composition technique, applying the entire PCP theorem to them. Our proof, in contrast, makes this recursion concrete, utilizing an explicit representation of low-degree polynomial functions that yields LDF-readers with an exponentially small error probability.

The composition-recursion LDF-reader. Our LDF-reader uses a representation of low-degree polynomials as follows. We begin with a representation where a multi-dimensional polynomial is represented by all of its point evaluations, and also by its restriction to certain constant dimensional subspaces. We use a new *power-substitution* technique to then replace each constant dimensional restriction of the polynomial by a multi-dimensional polynomial of a much smaller degree. This is done, roughly, by replacing monomials of high degree with new variables. The polynomials whose degree was reduced are then represented by their point evaluations and their restriction to constant dimensional subspaces, and the process is repeated.

After a constant number of such iterations we obtain polynomials of linear degree over constant dimensional spaces. Each of these polynomials is then represented by a constant number of variables that range over the field $\mathcal{F}$. Hence to obtain evaluations our LDF-reader is not required to completely read a low-degree polynomial over some subspace – instead it only accesses a constant number of variables that range over $\mathcal{F}$.

Organization of this part

Our main result and the main definitions required for its proof are stated in Chapter 2. The proof of the main result, based on lemmas that are proven in the following chapters, appears in Chapter 3. The construction of the LDF-reader that is utilized in the proof of the main result appears in Chapter 4. In particular, the power-substitution technique, used in the construction of the LDF-reader to represent polynomials using other polynomials with more variables but with considerably smaller degrees, is described in Subsection 4.3. Finally, Chapter 5 describes the recursive application of the sum-check (and other) techniques, which are used in the reduction to obtain from the original system Ψ a system Ψ_{sc} with a constant number of variables in each equation.

Chapter 2

Preliminaries

In this chapter we describe the basic ideas and definitions utilized in the proof of our main result.

Gap-Quadratic-Solvability

The Gap-Quadratic-Solvability problem is that of determining whether all the equations in a given system of quadratic-equations can be simultaneously satisfied, or whether only a small fraction of the equations can be satisfied. Viewing the quadratic equations as local-tests of a PCP system, showing this problem to be NP-hard yields a PCP characterization of NP.

Definition 2.1 (Gap-Quadratic-Solvability). *The Gap-Quadratic-Solvability problem with parameters D, $\mathcal{F}$ and ϵ (which are, implicitly, functions of the system size n), is denoted by gap-QS$[D, \mathcal{F}, \epsilon]$. An instance of the problem is a set of n quadratic-equations over a finite field $\mathcal{F}$, where each equation has at most D variables (D is called the dependency parameter). The problem is to distinguish between the following two cases:*

Yes. There is an assignment to the variables that satisfies all of the equations.

No. Every assignment to the variables satisfies at most an $\epsilon(n)$ fraction of the equations – ϵ is called the error parameter.

An instance which falls under one of the above criteria is said to have the gap property. Any outcome is acceptable for instances that do not have the gap property.

Our main theorem shows NP-hardness of gap-QS for a field of size $|\mathcal{F}| = 2^{c \log^{\beta} n}$ and an error parameter $\epsilon = \frac{2}{|\mathcal{F}|}$, where $\beta < 1$ is any constant smaller than 1 and $c > 0$ is some constant. Note that the requirement for a $\frac{2}{|\mathcal{F}|}$ error is almost optimal, since it is easy to satisfy a $\frac{1}{|\mathcal{F}|}$ fraction of the satisfiable equations of any system by a random assignment.

On the other hand, from hardness for any error $\epsilon = \frac{1}{|\mathcal{F}|^c}$ which is polynomially small in the size of the field, one can obtain hardness for a $\frac{2}{|\mathcal{F}|}$ error by a simple amplification technique.

We therefore abbreviate gap-QS$[D, \mathcal{F}]$ for the gap-QS problem where ϵ is fixed to be $\frac{2}{|\mathcal{F}|}$. Note that this error probability is polynomially small in the size of the field, and therefore exponentially small in the length, measured in bits, of each variable.

Theorem 2.2 (main theorem). *For every constant $\beta < 1$ there exists a constant $c > 0$ such that gap-QS$[O(1), \mathcal{F}]$ is NP-hard, where $|\mathcal{F}| = 2^{c \log^\beta n}$, n being the number of equations in the system.*

We actually prove Theorem 2.2 via a many-to-one reduction. Informally speaking, this means that gap-QS$[O(1), \mathcal{F}]$ is proven to be NP-complete.

Gap-QS$[n, \mathcal{F}]$, where the number of variables in each equation is not bounded, is proven to be NP-hard in [HPS93] for any field size bounded by n^γ, using simple linear codes:

Theorem 2.3 ([HPS93])**.** *Gap-QS$[n, \mathcal{F}]$ is NP-hard, for any field size bounded by n^γ, where $\gamma < 1$ is a constant.*

This theorem is proven by a relatively simple reduction from the Cook-Levin characterization of NP, so the entire difference between this characterization of NP and the PCP characterization boils down to the constant bound on the number of variables that each equation accesses.

Theorem 2.2 is proven by showing a reduction algorithm, taking as input a system Ψ of n quadratic-equations and producing a new system Ψ' where the number of variables in each equation is bounded by a constant. The number of variables in each equation is reduced while roughly preserving the fraction of satisfiable equations. Specifically, if Ψ is completely satisfiable then Ψ' is completely satisfiable as well; and if no more than a $\frac{2}{|\mathcal{F}|}$ fraction of the equations of Ψ can be satisfied then the same occurs for Ψ'.

LDFs and Domains

Let us set a notation for polynomial functions of low degree – an object used extensively in this part of the thesis.

Definition 2.4 (LDF - low degree function). *An $[r, d]$-LDF is a polynomial function from $\mathcal{F}^d$ to $\mathcal{F}$, of total-degree at most r.*

The variables of the system Ψ' that is generated by our reduction all range over $\mathcal{F}$ (since Ψ' is a system of quadratic equations over $\mathcal{F}$). For the exposition of the reduction algorithm and for the proof of correctness, it is useful to consider certain subsets of these variables and associate the variables in such a subset with points in a vector field $\mathcal{F}^d$ over $\mathcal{F}$. An assignment to the variables in the subset can thus be viewed as a function $f : \mathcal{F}^d \to \mathcal{F}$, as explained in the following definition of a domain.

Definition 2.5 (domain). *A* domain *F is a set of variables ranging over $\mathcal{F}$, that has one variable $F[x]$ for every point $x \in \mathcal{F}^d$, where $d = d(F)$ is called the* dimension *of the domain. F is said to be* assigned *a function $f : \mathcal{F}^d \to \mathcal{F}$ if for every x, the variable $F[x]$ is assigned $f(x)$.*

Two more parameters are associated with each domain in addition to the dimension – The lower-degree*, denoted $s(F)$, and the* upper-degree *$r(F)$ ($r(F)$ will always be larger than $s(F)$).*

Since a variable can be assigned every value in $\mathcal{F}$, a domain F can be assigned any function $f : \mathcal{F}^{d(F)} \to \mathcal{F}$. However, in the proof we give special consideration to assignments of domains which correspond to LDFs. In particular, it will be shown that if the system Ψ' generated by the reduction is satisfiable, there is a satisfying assignment where every domain F is assigned an $s(F)$-degree LDF. In case Ψ cannot be more than $\frac{2}{|\mathcal{F}|}$ satisfiable, it should be proven that *no assignment* can satisfy more than $\frac{2}{|\mathcal{F}|}$ of the equations of Ψ'. However, we prove later that it suffices to only show this for assignments where each domain F is assigned an $r(F)$-degree LDF.

Definition 2.6 (assignment of a domain). *The assignment f of a domain F is said to be* good*, if f is an $[s(F), d(F)]$-LDF, and it is said to be* feasible *if f is an $[r(F), d(F)]$-LDF. An assignment for a set of variables containing one or more domains is said to be good (feasible) if the assignment to each domain is good (feasible).*

The reduction which transforms Ψ into Ψ' goes through an intermediate system Ψ_1 where the number of variables in each equation is constant, but which does not yet have the desired properties. In particular, Ψ_1 might be completely satisfiable even when there exists no assignment satisfying more than a $\frac{2}{|\mathcal{F}|}$ fraction of the equations of Ψ. However, Ψ_1 behaves much better if we restrict the set of assignments considered: On one hand if Ψ is completely satisfiable then not only Ψ_1 is completely satisfiable, but there exists a good satisfying assignment for it. On the other hand, if Ψ is no more than $\frac{2}{|\mathcal{F}|}$-satisfiable, then there is no good or even feasible assignment for Ψ_1 satisfying more than a $\frac{2}{|\mathcal{F}|}$ fraction of its equations.

LDF-Readers

To transform Ψ_1 into the final system Ψ', we should prevent it from being satisfiable by assignments where domains are not assigned LDFs. In fact what we manage is a bit weaker (although it suffices). Consider an equation $\psi \in \Psi_1$ that has the variables $F[x_1], \ldots, F[x_k]$ in a domain F. To prevent it from being satisfiable by unwanted assignments we use a mechanism called an *LDF-reader*, which is plugged into ψ in place of these variables. The

LDF-reader ensures that ψ either reads evaluations at $(x_1, \ldots, x_k)$ of an LDF (even if the assignment to F is not feasible) or it is not satisfied.

An LDF-reader evaluating the tuple of points $(x_1, \ldots, x_k)$ in a domain F has two parts – the *representation*, and the set of *local-readers* which produce the evaluations.

The representation. The representation is a set V that contains F and maybe other variables and domains. The variables in V, including those associated with domains in it, are called *representation variables*. The LDF-reader uses the representation variables to produce evaluations. Every good assignment for F can be extended into a good assignment for all the variables in V, called the *encoding-assignment* of the LDF assigned to F. If F is assigned an LDF f and V is its encoding-assignment, then the LDF-reader will return the evaluations of f.

The local-readers

The evaluations for $(x_1, \ldots, x_k)$ are produced by a set of local-readers, where each local-reader accesses only a constant number of representation variables – this property is essential since the local-readers are plugged into Ψ_1 to produce Ψ' and the number of variables in each equation must remain constant. Each local-reader may either produce evaluations, or it may reject if it finds that the assignment is not an encoding-assignment.

Local-tests and evaluators. Each local-reader is a pair containing a *local-test* – a conjunction of linear equations over representation variables, and a tuple of k *evaluators*. Each evaluator is a linear-combination of representation variables. A local-reader is said to *accept* an assignment for the representation variables if the local-test is satisfied by it, and otherwise it is said to *reject* it. If $\mathcal{A}$ is an encoding-assignment of an LDF f, then all local-readers must accept, and moreover, the i'th evaluator in each local-reader must evaluate to $f(x_i)$.

In case the representation variables are not given a correct encoding-assignment, we would like the local-readers to always reject. This is not possible to achieve, however, with local-readers that access a constant number of representation variables, not even if we allow a small fraction of the local-readers to falsely accept. It is in fact not even possible to ensure that local-readers which do not reject return evaluations of a single LDF. What we can achieve (and turns out to be enough), is that apart from a small fraction, the local-readers either reject or return evaluations of one of a short list of LDFs. This is the list of LDFs which are *permissible* with respect to the assignment of F.

Definition 2.7 (permissible assignment). *An $[r(F), d(F)]$-LDF f is said to be ρ-permissible with respect to an assignment of F if for at least a ρ-fraction of the points x, $F[x]$ is assigned $f(x)$.*

We show later that for a wide range of permissibility parameters ρ, the list of permissible LDFs is bounded by $O(\rho^{-1})$. Since the list is only determined by the assignment to F and is independent of the rest of the representation variables, it will be the same for all LDF-readers evaluating tuples in F. This means that all equations that have variables in F will read evaluations that are consistent with one of the LDFs on the relatively short list.

We now give the formal definition of the parameters of an LDF-reader.

Definition 2.8 ((ρ, ϵ)-LDF-reader). *Let $\mathcal{R}$ be an LDF-reader evaluating a tuple $(x_1, \ldots, x_k)$ in a domain F, and fix an assignment to its representation-variables. A local-reader L is said to be ρ-*erroneous *if it accepts,* **and** *there exists no ρ-permissible LDF f (with respect to the assignment of F) such that the i'th evaluator evaluates to $f(x_i)$ for all i.*

$\mathcal{R}$ is said to be a (ρ, ϵ)-LDF-Reader, if for any assignment to the representation-variables, the fraction of ρ-erroneous local-readers is at most ϵ.

Chapter 3

Proof of the Main Theorem

In this chapter we prove Theorem 2.2 by showing for any constant $\beta < 1$, a polynomial time reduction from Gap-QS$[n, \mathcal{F}]$, where $|\mathcal{F}| = 2^{\log^\beta n}$, to Gap-QS$[O(1), \mathcal{F}]$. Given a system Ψ of n quadratic equations over $\mathcal{F}$, with up to n variables in each equation, the reduction generates a system Ψ' *over the same field* with at most a constant number of variables in each equation. Denoting by m the number of equations in Ψ', the size of the field as a function of m would be of the form $2^{c \log^\beta m}$ where $c > 0$ is some constant (as stated in Theorem 2.2), since m is polynomial in n.

Ψ' will have the *completeness* property, namely that if the given system Ψ can be completely satisfied then Ψ' will be completely satisfiable as well; and the *soundness* property – if Ψ is no more than $\frac{2}{|\mathcal{F}|}$-satisfiable (namely no assignment can satisfy more than a $\frac{2}{|\mathcal{F}|}$ fraction of its equations), then Ψ' is at most $\frac{2}{|\mathcal{F}|}$-satisfiable as well.

The reduction begins by transforming Ψ into a system Ψ_{sc} of quadratic-equations where the number of variables in each equation is bounded by constant, and that has the desired properties only if the assignments for its variables are restricted to being feasible. The transformation of Ψ into Ψ_{sc} is done by a *sum-check* algorithm, whose properties are stated in the following Lemma, proven in Chapter 5.

Lemma 3.1 (sum-check). *There exists a polynomial-time algorithm as follows. It takes as input a system Ψ of n quadratic equations over a field $\mathcal{F}$, $|\mathcal{F}| = 2^{\log^\beta n}$, where there are up to n variables in each equation. Given Ψ, the algorithm generates a system Ψ_{sc} of quadratic-equations over $\mathcal{F}$ where each equation has a constant number of variables (some of which in domains), and that has the following properties:*

- *Completeness: If Ψ is completely satisfiable then Ψ_{sc} is completely satisfiable by a good assignment.*
- *Soundness: If Ψ is no more than $\frac{2}{|\mathcal{F}|}$-satisfiable then Ψ_{sc} cannot be more than $\frac{2}{|\mathcal{F}|}$-satisfied by a feasible assignment.*

Moreover, all the domains of Ψ_{sc} have the same dimension $d = \Theta(\log^{1-\beta} n)$, lower-degree s, and upper-degree r, where $s \leq |\mathcal{F}|^{c_1}$ and $r \geq |\mathcal{F}|^{c_2}$ for some global constants $c_1 < c_2 < 1$.

To transform Ψ_{sc} into Ψ', the reduction generates LDF-readers and plugs them into Ψ_{sc} as follows. For each equation ψ of Ψ_{sc}, that has the variables $F(x_1), \ldots, F(x_k)$ of a domain F, it generates an LDF-reader evaluating $(x_1, \ldots, x_k)$ in F. To plug the LDF-reader into ψ, many copies of ψ are made, and one of the local-readers is plugged into each copy.

The local-tests are added in conjunction with each copy, and hence a system of *conjunctions* is formed. In addition, some of the gap is lost when the LDF-readers are plugged in – if Ψ is no more than $\frac{2}{|\mathcal{F}|}$-satisfiable, the fraction of satisfiable conjunctions in the system obtained from Ψ_{sc} might be somewhat higher. A simple amplification technique is hence applied to the system of conjunctions to avoid that, and then each conjunction is replaced by equations, obtaining Ψ'.

Generating the LDF-Readers

To generate LDF-readers we use a *constructor* algorithm, as defined below.

Definition 3.2 (constructor). *A constructor is an algorithm that takes as input a domain F and a k-tuple $(x_1, \ldots, x_k)$ of points in $\mathcal{F}^{d(F)}$, where k is a constant. It generates an LDF-reader evaluating $(x_1, \ldots, x_k)$ in F, in time polynomial in $|\mathcal{F}|^{d(F)}$. The number of variables appearing in each local-reader must be bounded by a constant, and so should be the number of equations in the local-test of each local-reader. In addition, the number of local-readers must only depend on the parameters of F.*

The reduction uses a constructor that generates *Composition-Recursion* LDF-readers. The existence of the constructor and the properties of the LDF-readers it generates are stated in the next lemma, which is proven in Chapter 4.

Lemma 3.3 (Composition-Recursion constructor). *There exists a global constant c_g, such that for every $c_1 < c_2 < 1$ and $\beta < 1$ the following holds. There exist a constant $c > 0$, and an LDF-Reader constructor for domains of dimension $d = \Theta(\log^{1-\beta} n)$, lower-degree $s \leq |\mathcal{F}|^{c_1}$, and upper-degree $r \geq |\mathcal{F}|^{c_2}$ (the algorithm runs independently of s and r). The LDF-readers generated by the algorithm are $(\rho, O(\rho^c))$-LDF-readers, for all ρ's which satisfy $\rho > (r/|\mathcal{F}|)^{c_g} d$.*

Before the LDF-readers are actually generated, we make some small technical alterations to Ψ_{sc} as follows.

Uniformization. The number of variables in each equation of Ψ_{sc} is bounded by some constant k. This implies that an equation in Ψ_{sc} may have variables from up to k distinct domains, and that the number of variables it has from each domain is bounded by k.

Before generating LDF-readers, let us assume for simplicity that each equation of Ψ_{sc} has variables from *exactly* k distinct domains, and that it has exactly k variables from each domain. This requires the reduction to add arbitrary variables to the equations, multiplied by zero coefficients.

LDF-Reader generation. After the uniformization, the reduction generates the LDF-readers as described above – For each equation ψ of Ψ_{sc}, that has the variables $F(x_1), \ldots, F(x_k)$ in a domain F, it generates an LDF-reader evaluating $(x_1, \ldots, x_k)$ in F (this takes polynomial time in the size of Ψ_{sc}). Note that since all domains in Ψ_{sc} have the same parameters (dimension d, lower-degree s and upper-degree r, as stated in Lemma 3.1), the number of local-readers in each LDF-reader is the same as well.

The representation variables of the LDF-readers are added to the variables of the system, and the local-readers are plugged into the equations of Ψ_{sc} as described below.

Plugging LDF-Readers In

For each equation $\psi \in \Psi_{sc}$ there are now k associated LDF-readers – one for each domain it has variables from. The first step in plugging the LDF-readers into Ψ_{sc} is to replace each such equation ψ by its *representation set* $\mathcal{E}_\psi$, containing *conjunctions* of quadratic equations that are obtained by plugging local-readers into ψ. $\mathcal{E}_\psi$ represents ψ in the sense that an assignment satisfying a large enough fraction of the conjunctions in $\mathcal{E}_\psi$ implies a satisfying assignment for ψ, as shown in the proof of Claim 3.4 below.

Generating $\mathcal{E}_\psi$. Let $\psi \in \Psi_{sc}$ be an equation that has variables from the domains $F_1, \ldots, F_k$. For each j, let us denote the variables of ψ in F_j by $F_j[x_1^j], \ldots, F_j[x_k^j]$. ψ is therefore associated with k LDF-readers $\mathcal{R}_1, \ldots, \mathcal{R}_k$, where $\mathcal{R}_j$ evaluates the tuple $(x_1^j, \ldots, x_k^j)$ in F_j. The reduction generates one conjunction in $\mathcal{E}_\psi$ for each choice of k local-readers $L_1, \ldots, L_k$, where $L_j \in \mathcal{R}_j$. The first equation in each such conjunction, denoted ψ', is the quadratic equation obtained from ψ by replacing each variable of the form $F_j[x_i^j]$ with the i'th evaluator of L_j (it evaluates x_i^j in F_j). ψ' is then put in conjunction with the local-tests of the local-readers $L_1, \ldots, L_k$.

The system Ψ_2. Note that the number of conjunctions in $\mathcal{E}_\psi$ is the same for every $\psi \in \Psi_{sc}$ – it is $|\mathcal{R}|^k$, where $|\mathcal{R}|$ denotes the number of local-readers in each of the LDF-readers we have generated. We set Ψ_2 to be the union of all the sets $\mathcal{E}_\psi$. Since the number of variables in each local-reader is constant, the number of variables in each conjunction of Ψ_2 is bounded by a constant as well. The system of conjunctions Ψ_2 obviously retains the completeness property of Ψ_{sc}. As the next claim shows, it also retains *some* of its soundness property, even with respect to assignments which are not necessarily feasible.

Claim 3.4. *There exists a constant α, $0 < \alpha < 1$, such that Ψ_2 has the following properties:*

- *Completeness: If Ψ is completely satisfiable, then Ψ_2 is completely satisfiable as well.*
- *Weakened Soundness: If Ψ is at most $2/|\mathcal{F}|$-satisfiable, then Ψ_2 is at most $|\mathcal{F}|^{-\alpha}$-satisfiable (by any assignment).*

To prove the claim we need the following proposition, showing that there cannot be many permissible LDFs for a domain – this implies that most local-readers in an LDF-reader will either reject or return the evaluation of one of a *short list* of permissible LDFs. This proposition appears in Chapter 4 as Claim 4.12, and is proven there.

Proposition 3.5. *Let F be a domain, and let $\rho > \left(\frac{r(F)}{|\mathcal{F}|}\right)^{c_g} d(F)$ where c_g is the same constant as in Lemma 3.1. Then for any assignment to F there can be at most $2\rho^{-1}$ ρ-permissible LDFs in all.*

Proof of Claim 3.4:

Completeness. If Ψ is satisfiable, then there is a good assignment satisfying Ψ_{sc}. For each of the constructed LDF-readers, extend the assignment to its representation using the encoding-assignment of the associated domain. The extended assignment satisfies Ψ_2: A conjunction in Ψ_2 contains local-tests, which are all satisfied by encoding-assignments, and an equation ψ'. ψ' was generated from an equation $\psi \in \Psi_{sc}$ by replacing variables with evaluators. But for encoding-assignments, the evaluators and the replaced variables have the same values. Hence since ψ is satisfied, ψ' is satisfied as well.

Weakened soundness. Fix an assignment $\mathcal{A}$ for Ψ_2, and let γ be the fraction of conjunctions it satisfies. For an appropriate α, we will show that if $\gamma > |\mathcal{F}|^{-\alpha}$ then there exists a feasible assignment for Ψ_{sc} satisfying more than a $\frac{2}{|\mathcal{F}|}$ fraction of its equation. This implies that Ψ is more than $\frac{2}{|\mathcal{F}|}$-satisfiable – a contradiction.

We denote $a \doteq (1-c_1)c_g/k$, where c_1 is the global constant mentioned in the Sum-Check Lemma (Lemma 3.1). Let $\rho \doteq |\mathcal{F}|^{-a}$. By the choice of a it follows that $\rho > (r/|\mathcal{F}|)^{c_g} d$, and therefore the LDF-readers have parameters (ρ, ρ^c) where c is the constant mentioned in the Composition-Recursion Constructor Lemma.

An equation $\psi \in \Psi_{sc}$ such that the fraction of satisfied conjunctions in $\mathcal{E}_\psi$ is higher than $k\rho^c$, is said to be *potentially satisfiable*. Since the sets $\mathcal{E}_\psi$ are all of the same size, it follows that the fraction of potentially satisfiable equations is at least $\gamma - k\rho^c$.

Consider a potentially satisfiable equation ψ. $\mathcal{E}_\psi$ was generated from ψ by plugging in k LDF-readers $\mathcal{R}_1, \ldots, \mathcal{R}_k$, evaluating tuples in k domains $F_1, \ldots, F_k$ respectively. A conjunction in $\mathcal{E}_\psi$ is defined by choosing a local-reader L_j out of each LDF-reader $\mathcal{R}_j$. For every j, the fraction of conjunctions in $\mathcal{E}_\psi$ where L_j is ρ-erroneous is bounded by ρ^c, as implied by the parameters of the LDF-readers, and hence the fraction of conjunctions where *any* of the readers are erroneous is bounded by $k\rho^c$.

It follows that there exists a satisfied conjunction in $\mathcal{E}_\psi$ in which no local-reader is erroneous, namely that the evaluator of each local-reader L_j gives the evaluations of a ρ-permissible LDF f_j with respect to the assignment of F_j. Hence if each domain F_j were re-assigned the function f_j, ψ would be satisfied.

So far we have shown that the potentially satisfiable equations, which make at least a $\gamma - k\rho^c$ fraction of the equations $\psi \in \Psi_{sc}$, can be satisfied by re-assigning the domains with ρ-permissible LDFs. For each domain F in Ψ_{sc}, choose a *random* ρ-permissible LDF, or the zero LDF if no such LDF exists, and re-assign it to F. We have obtained a *feasible assignment* for Ψ_{sc}. We compute the chance of a potentially satisfiable equation to be satisfied by the new assignment.

There are at most $O(\rho^{-1})$ ρ-permissible LDFs for each domain by Proposition 3.5, and each equation has variables from k domains. Hence the probability of a potentially satisfiable equation in Ψ_{sc} to be actually satisfied by the re-assignment is at least $\Omega(\rho^k)$. It follows that the expected fraction of satisfied equations in Ψ_{sc} is $\Omega(\rho^k(\gamma - k\rho^c))$, and hence at least one of the re-assignments achieves this fraction of satisfaction. We have thus shown that there exists a feasible assignment for Ψ_{sc} satisfying an $\Omega(\rho^k(\gamma - k\rho^c))$ fraction of its equations.

We now choose a constant α so that $0 < \alpha < \min\{1 - ka\,,\, ac\}$ (note that such an α exists). If $\gamma > |\mathcal{F}|^{-\alpha}$, then

$$\rho^k(\gamma - k\rho^c) = |\mathcal{F}|^{-ak}(\gamma - k|\mathcal{F}|^{-ac}) \gg \frac{2}{|\mathcal{F}|}$$

hence there exists a feasible assignment for Ψ_{sc} satisfying more than a $\frac{2}{|\mathcal{F}|}$ fraction of its equations.

Gap Amplification

The reduction now amplifies the soundness of Ψ_2 by joining conjunctions together into larger conjunctions, generating Ψ_3. The soundness of Ψ_3 is even stronger than needed, but it still has conjunctions rather than equations. The next subsection describes how conjunctions may be replaced by equations with only a small cost in the soundness, thus completing the reduction.

The system Ψ_3. Denote $N \doteq \lceil 1/\alpha \rceil$, where α is the constant mentioned in Claim 3.4. The reduction generates Ψ_3 by taking the conjunction of every ordered N-tuple of (not necessarily distinct) conjunctions from Ψ_2, that is

$$\Psi_3 = \{ \quad \bigwedge_{i=1}^{N} \chi_i \ : \quad \forall\, i \ \ \chi_i \in \Psi_2 \ \}$$

Note that it takes polynomial time in $|\mathcal{F}|^N$, and hence in n, to generate Ψ_3. Since each conjunction in Ψ_3 is composed of a constant number of conjunctions from Ψ_2, the number of variables as well as the number of equations in each such conjunction is bounded by a constant. The next claim states the completeness and soundness properties of Ψ_3.

Claim 3.6. *Ψ_3 has the following properties:*

- *Completeness: If Ψ is completely satisfiable, then Ψ_3 is completely satisfiable as well.*
- *Soundness: If Ψ is at most $2/|\mathcal{F}|$-satisfiable, then Ψ_3 is at most $1/|\mathcal{F}|$-satisfiable.*

The claim follows easily from Claim 3.4 and from the construction of Ψ_3.

From Conjunctions to Equations

Ψ_3 is a system of conjunctions where, as mentioned above, the number of equations in each conjunction is bounded by a constant. We would like the reduction to transform it from a system of conjunctions into the final system Ψ' of quadratic-equations, but first we make sure that the number of equations in all the conjunctions of Ψ_3 is the same. To do so the reduction adds equations of the form $0 = 0$ where necessary.

The system Ψ'. To transform Ψ_3 into Ψ', the reduction replaces each conjunction in Ψ_3 with the set of all linear-combinations over its equations (equations can be added or multiplied by a scalar, so the notion of a linear-combination of equations is well defined). Since the number of equations in each conjunction is constant the blow-up is polynomial in $|\mathcal{F}|$, and hence in n.

Since the number of variables in each conjunction of Ψ_3 is bounded by a constant, the number of variables in each equation of Ψ' is constant as well. In order to complete the proof of Theorem 2.2, it is left to show that Ψ' has the soundness and completeness properties. This follows immediately from Claim 3.6 together with the next proposition.

Proposition 3.7 (conjunction representation). *Let Ψ_a be a system of conjunctions of equations over $\mathcal{F}$, where the number of equations in each conjunction is the same. Let Ψ_b be the system obtained from Ψ_a where every conjunction $\chi \in \Psi_a$ is replaced by all linear combinations over $\mathcal{F}$ of its equations (with multiplicities, if the same equation occurs more than once). Then*

- *If Ψ_a is completely satisfied by a certain assignment, then the same assignment will satisfy Ψ_b as well.*
- *If Ψ_a is at most γ-satisfiable then Ψ_b is at most $(\gamma + 1/|\mathcal{F}|)$-satisfiable.*

Proof. The first property is obvious from the definition of Ψ_b. To prove the second property, fix an assignment for the variables of Ψ_a and Ψ_b. Then it satisfies at most a γ fraction of the conjunctions in Ψ_a. For each conjunction χ in Ψ_a denote by $\omega(\chi)$ the fraction of equations replacing χ that are satisfied in Ψ_b. Since each conjunction of Ψ_a is replaced by the same number of equations, the fraction of satisfied equations in Ψ_b is the average of $\omega(\chi)$ over all the conjunctions $\chi \in \Psi_a$.

For a satisfied conjunction χ, $\omega(\chi) = 1$, and it is easy to observe that $\omega(\chi) = 1/|\mathcal{F}|$ for any unsatisfied conjunction χ. Since satisfied conjunctions make at most a γ fraction of the conjunctions in Ψ_a, we conclude that the fraction of satisfied equations in Ψ_b is no more than $\gamma + 1/|\mathcal{F}|$, as required. ■

Chapter 4

The Composition-Recursion Constructor

This chapter contains the proof of the Composition-Recursion Constructor Lemma (Lemma 3.3), showing the constructor for Composition-Recursion LDF-readers, CR's for short. As a first step, we show a constructor for *restricted* LDF-readers, where some of the domains in the representation are considered *active*. These LDF-readers have good parameters only in the case where active domains are given feasible assignments. By a composition of several such LDF-readers we then get a CR.

Definition 4.1 (restricted LDF-readers.). *A restricted LDF-reader $\mathcal{R}$ is an LDF-reader where some of the domains in the representation are considered* active. *The dimension, and the upper and lower degree parameters of all active domains must be the same. These parameters are called the active dimension, active upper-degree and active lower-degree of $\mathcal{R}$, and are denoted by $d_\star(\mathcal{R})$, $r_\star(\mathcal{R})$, and $s_\star(\mathcal{R})$ respectively.*

A local-reader L in $\mathcal{R}$ may have variables from at most one active domain, which is called the active domain of L and is denoted by $\mathrm{Dom}_\star(L)$.

Parameters of restricted LDF-readers. We measure the parameters of restricted LDF-readers only with respect to feasible assignments: An assignment for the representation of a restricted LDF-reader $\mathcal{R}$ is said to be *feasible* if the assignment of every active domain is feasible (unlike in the case of equation-systems, we do not require the assignment of all domains to be feasible). $\mathcal{R}$ is hence said to be a restricted (ρ, ϵ)-LDF-Reader if for any feasible assignment, the fraction of ρ-erroneous local-readers is at most ϵ. Note that in an encoding-assignment, all domains must still be given a good assignment.

Outline of this chapter. The next section shows a constructor for restricted LDF-readers (the definition of a constructor generalizes naturally for restricted LDF-readers), called Subspace-vs.-Point LDF-readers, SP's for short. The representation of an SP evaluating a tuple in a domain F contains, apart from F itself, only active domains, which have the same degree-parameters as F but a *constant* dimension parameter. Therefore, informally speaking, an SP LDF-reader uses constant-dimensional LDFs to represent an LDF over a space of higher dimension, and using evaluations of these constant-dimensional LDFs it produces consistent evaluations of the original LDF.

In Section 4.3 it is shown how the constant-dimensional active domains of an SP, $\mathcal{R}$, can be replaced by active domains that have non-constant dimension, but greatly decreased degree parameters. This allows the composition of other SP's over $\mathcal{R}$, as described in Section 4.4, to evaluate tuples in the replaced active domains. Section 4.5 shows how an iterative application of this procedure yields the Composition-Recursion LDF-reader (which is not restricted) and Section 4.6 proves its properties, thus completing the proof of Lemma 3.3.

4.1 Subspace-vs.-Point LDF-Readers

In this section we show the SP constructor – a constructor that generates Subspace-vs.-Point restricted LDF-readers. The representation of an SP that evaluates a k-tuple in a domain F contains, in addition to F, active domains with the same degree parameters as F but of dimension $k+2$. Each domain is associated with a $(k+2)$-dimensional subspace U in $\mathcal{F}^{d(F)}$; in an encoding-assignment each of them is assigned the restriction to U of the LDF assigned to F. Before we go into the description of the constructor, let us state its properties in the following lemma.

Lemma 4.2 (Subspace-vs.-Point LDF-reader). *There exists a constructor that given a domain F and a k-tuple of points in $\mathcal{F}^{d(F)}$, generates a restricted LDF-reader $\mathcal{R}$ as follows. The active domains of $\mathcal{R}$ have the same upper and lower-degree parameters as F, and their dimension parameter equals $k+2$. Moreover, $\mathcal{R}$ will have parameters $(\rho, O(\rho^{1/3}))$ for all ρ's which satisfy $\rho > (r(F)/|\mathcal{F}|)^{c_g} d(F)$, where $0 < c_g \leq 1/2$ is a global constant*[*].*

The Subspace-vs.-Point constructor

We now describe how the SP constructor generates an LDF-reader $\mathcal{R}$, given a domain F and a k-tuple $(x_1, \ldots, x_k)$ of points in $\mathcal{F}^{d(F)}$. The SP constructor is used later as a procedure of the CR constructor, however in proving the parameters of the CR constructor we only rely on the properties that are stated in Lemma 4.2. Without loss of generality, throughout the construction it is assumed that $d(F) \geq k+2$ – it is easy to adapt the construction for the case $d(F) < k+2$.

[*] This is the same constant as in Lemma 3.3, and in all other places where c_g appears

The representation. Other than F itself, the representation only includes active domains, with upper-degree $r(F)$, lower-degree $s(F)$, and dimension $k+2$. The constructor first picks any $(k-1)$-dimensional affine subspace $U_0 \subseteq \mathcal{F}^{d(F)}$ which contains all the points x_i of the tuple (if the x_i's are in general position, there exists exactly one such subspace). Denote by $\mathcal{S}ubSp(\mathcal{R})$ the set of $(k+2)$-dimensional affine subspaces $U \subseteq \mathcal{F}^{d(F)}$ which contain U_0. One active domain $\mathcal{D}_U$ is then constructed for every affine subspace $U \in \mathcal{S}ubSp(\mathcal{R})$.

Identification functions. A good assignment $\mathcal{A}$ to F assigns to it an $[s(F), d(F)]$-LDF f. In the encoding-assignment, the assignment to each domain $\mathcal{D}_U$ represents the restriction of f to U. In order to represent $f|_U$ as an LDF over $\mathcal{F}^{k+2}$ the constructor chooses for each domain $\mathcal{D}_U$ an arbitrary linear isomorphism $\phi_{_U} : U \to \mathcal{F}^{k+2}$, called the *identification function* of U, that identifies each point $y \in U$ with the point $\phi_{_U}(y)$ in $\mathcal{F}^{k+2}$.

Encoding-assignments. Let $\mathcal{A}$ be a good assignment for F, assigning to it a $[d(F), s(F)]$-degree LDF f. The encoding-assignment for $\mathcal{A}$ extends it by assigning to each domain $\mathcal{D}_U$ the LDF $f \circ ({\phi_{_U}}^{-1})$. Composing the assignment of $\mathcal{D}_U$ with $\phi_{_U}$ therefore gives $f|_U$. Since $\phi_{_U}$ is linear, $\mathcal{D}_U$ is assigned an $s(F)$-degree LDF, and hence the encoding-assignment of $\mathcal{A}$ is a good assignment.

Local-readers. The SP constructor generates one local-reader for each domain $\mathcal{D}_U$ and point $y \in U$. Its active domain is $\mathcal{D}_U$, its local-test is the single linear equation $\mathcal{D}_U[\phi_{_U}(y)] = F[y]$, and for every $1 \leq i \leq k$ its i'th evaluator is the term $\mathcal{D}_U[\phi_{_U}(x_i)]$.

To get a better understanding of the structure of local-readers, fix a feasible assignment $\mathcal{A}$ for the representation variables, and consider a local-reader associated with a domain $\mathcal{D}_U$ and a point y. The LDF assigned to $\mathcal{D}_U$ represents an $r(F)$-degree LDF $g_{_U}$ over U, defined by $g_{_U}(x) \doteq \mathcal{A}(\mathcal{D}_U[\phi_{_U}(x)])$ (this is the composition of the LDF assigned to $\mathcal{D}_U$ with $\phi_{_U}$). The local-test therefore compares $g_{_U}(y)$ with the assignment of $F[y]$, and the i'th evaluator returns $g_{_U}(x_i)$.

If $\mathcal{A}$ is the encoding-assignment of an LDF f, then $F[y]$ is assigned $f(y)$, and $g_{_U}$ is the restriction of f to U. The local-test is hence satisfied in that case, and the values $g_{_U}(x_i)$ returned by the evaluators are in fact the values of f at the points x_i.

The SP constructor works.

It is easy to verify that the SP constructor indeed falls under the definition of a constructor. To verify the parameters of SP LDF-readers (which would conclude the proof of Lemma 4.2) consider an SP LDF-reader $\mathcal{R}$, that evaluates a k-tuple $(x_1, \ldots, x_k)$ in a domain F, and fix a feasible assignment $\mathcal{A}$ for its representation. As explained above, $\mathcal{A}$ determines an $r(F)$-degree LDF $g_{_U}$ over every affine subspace $U \in \mathcal{S}ubSp(\mathcal{R})$.

The local-test of the local-reader determined by an affine subspace $U \in \mathcal{S}ubSp(\mathcal{R})$ and a point $y \in U$ verifies that $g_{_U}(y) = \mathcal{A}(F[y])$, and its evaluators return the values of $g_{_U}$ at the points $x_1, \ldots, x_k$. The local-reader is ρ-erroneous if the local-test is satisfied, yet the values $g_{_U}(x_1), \ldots, g_{_U}(x_k)$ are not the evaluation of any ρ-permissible LDF (with respect to the assignment of F) at $x_1, \ldots, x_k$. The next proposition bounds the fraction of erroneous local-readers, thus proving that $\mathcal{R}$ has the parameters required in Lemma 4.2.

Proposition 4.3 (SP parameters). *For some global constant $0 < c_g \leq 1/2$, and for all ρ's which satisfy $\rho > (r(F)/|\mathcal{F}|)^{c_g} d(F)$, the following holds.*

Let U be a random affine subspace in $\mathcal{S}ubSp(\mathcal{R})$, and y be a random point in U (this determines a random local-reader). Let $\mathbf{E}rr$ be the event that $g_{_U}(y) = \mathcal{A}(F[y])$, yet $g_{_U}(x_1), \ldots, g_{_U}(x_k)$ are not the evaluation of any ρ-permissible LDF (with respect to the assignment of F) at $x_1, \ldots, x_k$. Then $\Pr[\mathbf{E}rr] = O(\rho^{1/3})$.

In fact we show an stronger statement than the above proposition. We bound by $O(\rho^{1/3})$ the probability that $g_{_U}$ agrees with the assignment of F at y, yet $g_{_U}$ is not the restriction to U of any ρ-permissible LDF (it is easy to observe that this implies Proposition 4.3).

Subspace-vs.-Point Parameters

The proof is based on a lemma from [RS97] where a slightly different setting is discussed. While we are interested in the case where an LDF is associated with every $(k+2)$-dimensional subspace in a certain set $\mathcal{S}ubSp(\mathcal{R})$, the [RS97]-Lemma deals with the case where each plane (2-dimensional affine subspace) is associated with an LDF defined over it.

Definition 4.4 (plane-assignment). *Suppose that every plane P in $\mathcal{F}^{d(F)}$ is associated with an $r(F)$-degree LDF $g_{_P}$ over P. The correspondence $P \rightarrow g_{_P}$ is called a plane-assignment. An LDF $g_{_P}$ is called the plane-LDF assigned to P.*

Another difference is that the [RS97]-Lemma discusses a different kind of permissibility, measured with respect to the plane-assignment instead of with respect to the assignment of F.

Definition 4.5 (planewise-permissibility). *Let $(P \rightarrow g_{_P})$ be a plane-assignment. An $r(F)$-degree LDF f over $\mathcal{F}^{d(F)}$ is said to be ρ-planewise-permissible if for at least a ρ-fraction of the planes P, $g_{_P} = f|_P$.*

A plane-LDF $g_{_P}$ is said to be ρ-planewise-permissible if it is the restriction to P of a ρ-planewise-permissible LDF f over $\mathcal{F}^{d(F)}$.

We now state the discussed lemma from [RS97]. It shows that planewise-permissibility can be tested by comparing the plane-LDFs assigned to two line-intersecting planes. If the plane-LDFs agree on the line then with high probability they are both planewise-permissible.

Lemma 4.6 ([RS97]). *There exists a global constant $0 < c_g \leq 1/2$, such that for any $\rho > (r(F)/|\mathcal{F}|)^{c_g} d(F)$ the following holds.*

Fix a plane-assignment $(P \rightarrow g_P)$. Let ℓ be a random line in $\mathcal{F}^{d(F)}$ and let P_1 and P_2 be two random, independently chosen planes that contain ℓ. Denote by $\mathbf{E}rr$ *the event that the plane-LDF assigned to P_1 agrees on ℓ with the plane-LDF assigned to P_2, yet they are not both ρ-planewise-permissible. Then* $\Pr[\mathbf{E}rr] = O(\rho)$.

Starting from Lemma 4.6, we gradually approach Proposition 4.3 by a sequence of technical claims: Claim 4.7 shows Lemma 4.6 to hold even if two plane-LDFs are compared on a point rather than a line. Claim 4.8 deals with the case where a plane-LDF is compared against the assignment of F at a certain point, rather than against another plane-LDF. Claim 4.10 goes from planewise-permissibility to permissibility, showing that a random plane-LDF g_P that agrees with the assignment of F at a random point on P is with high probability the restriction to P of a ρ-permissible LDF. Finally Claim 4.14 completes the proof of Proposition 4.3 by showing that the same holds for LDFs g_U associated with a random subspace $U \in \mathcal{S}ubSp(\mathcal{R})$, instead of plane-LDFs.

Claim 4.7. *Fix a plane assignment $(P \rightarrow g_P)$, and suppose ρ satisfies the requirements of Lemma 4.6. Let y be a random point in $\mathcal{F}^{d(F)}$, let ℓ be a random line containing it, and let P_1 and P_2 be random independently chosen planes that contain ℓ. Denote by* $\mathbf{E}rr$ *the event that the plane-LDF assigned to P_1 agrees on y with the plane-LDF assigned to P_2, yet they are not both ρ-planewise-permissible. Then* $\Pr[\mathbf{E}rr] = O(\rho)$.

Proof. First note that y can be considered to be a random point on a randomly chosen line ℓ in $\mathcal{F}^{d(F)}$, instead of the other way around.

The only case where $\mathbf{E}rr$ occurs yet the event from Lemma 4.6 does not, is when the restrictions to ℓ of the plane-LDFs of P_1 and P_2 differ, yet they agree on y. Since the restrictions to ℓ are $r(F)$-degree LDFs, if they differ then the probability of agreement on the random point y is $r(F)/|\mathcal{F}| \leq \rho$. Hence by Lemma 4.6, $\Pr[\mathbf{E}rr] \leq \rho + O(\rho) = O(\rho)$. ∎

The next step is to compare a plane-LDF to the assignment of $F[y]$ for some point y on it, instead of to the value at y of another plane-LDF .

Claim 4.8. *Fix a plane assignment $(P \rightarrow g_P)$, and suppose ρ satisfies the requirements of Lemma 4.6. Let P be a random plane and y be a random point on P. Denote by* $\mathbf{E}rr$ *the event that $g_P(y) = \mathcal{A}(F[y])$, yet g_P is not ρ-planewise-permissible. Then* $\Pr[\mathbf{E}rr] = O(\rho^{1/2})$.

Proof. To be able to apply Lemma 4.6, we redefine y and P, and introduce new random variables as follows. Let y be a random point in $\mathcal{F}^{d(F)}$, ℓ be a random line containing y, and P and P' be random independently chosen planes that contain ℓ (note that to obtain the claim it is enough to the prove bound on $\Pr[\mathbf{E}rr]$ in these settings). Let $\mathbf{E}rr_2$ be the event that g_P and $g_{P'}$ agree on y yet they are not both ρ-planewise-permissible. Claim 4.7 implies that the probability of $\mathbf{E}rr_2$ is bounded by $O(\rho)$.

To use the bound we have for $\mathbf{E}rr_2$ we first show that for every fixed line ℓ_0 and point $y_0 \in \ell_0$,

$$\Pr[\mathbf{E}rr|\ell = \ell_0, y = y_0] \leq (\Pr[\mathbf{E}rr_2|\ell = \ell_0, y = y_0])^{1/2} \qquad (4.1)$$

Let $\mathbf{E}rr'$ be the event that $g_{P'}(y) = \mathcal{A}(F[y])$ yet $g_{P'}$ is not ρ-planewise-permissible (it is similar to $\mathbf{E}rr$, only for P' instead of P). Obviously

$$\Pr[\mathbf{E}rr_2|\ell = \ell_0, y = y_0] \geq \Pr[\mathbf{E}rr \wedge \mathbf{E}rr'|\ell = \ell_0, y = y_0]$$

because the event on the left-hand side contains the one on the right-hand side. Since P and P' are independently chosen given ℓ_0, we have

$$\begin{aligned}\Pr[\mathbf{E}rr \wedge \mathbf{E}rr'|\ell = \ell_0, y = y_0] &= \Pr[\mathbf{E}rr|\ell = \ell_0, y = y_0] \cdot \Pr[\mathbf{E}rr'|\ell = \ell_0, y = y_0] \\ &= (\Pr[\mathbf{E}rr|\ell = \ell_0, y = y_0])^2\end{aligned}$$

which together with the above inequality implies Equation 4.1.

One may discard the conditioning in Equation 4.1, obtaining

$$\Pr[\mathbf{E}rr] \leq \Pr[\mathbf{E}rr_2]^{1/2}$$

using the law of complete probability and the concavity of the square-root function. Since the probability of $\mathbf{E}rr_2$ is bounded by $O(\rho)$, this obtains the claim. ■

Our next step is to convert the statement of Claim 4.8 from planewise-permissibility to permissibility in the usual sense. This requires the following bound on the number of planewise-permissible LDFs.

Claim 4.9. *Fix a plane assignment $(P \rightarrow g_P)$, and suppose ρ satisfies the requirements of Lemma 4.6. Then the number of ρ-planewise-permissible LDFs is less than $2\rho^{-1}$.*

Proof. The proof is similar to that of Claim 4.12 below. ■

We can now prove the analogue of Claim 4.8 for permissibility in the usual sense.

Claim 4.10. *Fix a plane assignment $(P \rightarrow g_P)$, and suppose ρ satisfies the requirements of Lemma 4.6. Let P be a random plane and y be a random point on P. Denote by* $\mathbf{E}rr$ *the event that $g_P(y) = \mathcal{A}(F[y])$, yet there is no ρ-permissible LDF f (with respect to the assignment of F), such that $g_P = f|_P$. Then* $\Pr[\mathbf{E}rr] = O(\rho^{1/3})$.

Proof. We separate $\mathbf{E}rr$ into two events, and bound the probability of each by $O(\rho^{1/3})$: Let $\mathbf{E}rr_1$ be the event where $\mathbf{E}rr$ occurs and in addition g_P **is not** $\rho^{2/3}$-planewise-permissible; and let $\mathbf{E}rr_2$ be the event where $\mathbf{E}rr$ occurs and in addition g_P is $\rho^{2/3}$-planewise-permissible.

By applying Claim 4.8 using $\rho^{2/3}$ instead of ρ, we obtain that the probability of $\mathbf{E}rr_1$ is bounded by $O(\rho^{1/3})$ as required (since $\rho > (r(F)/|\mathcal{F}|)^{c_g} d(F)$ as required in Lemma 4.6, $\rho^{2/3}$ satisfies this requirement as well).

It is left to bound the probability of $\mathbf{E}rr_2$. By definition it occurs only when $g_{_P}(y) = \mathcal{A}(F[y])$ and there exists a $\rho^{2/3}$-planewise-permissible LDF f which is not ρ-permissible, such that $g_{_P} = f|_P$. For an LDF f over $\mathcal{F}^{d(F)}$, denote by $\mathbf{E}rr_3(f)$ the event where $g_{_P}(y) = \mathcal{A}(F[y])$ and $g_{_P} = f|_P$ (note that this implies $f(y) = \mathcal{A}(F[y])$). Then the probability of $\mathbf{E}rr_2$ is bounded by the sum of $\Pr[\mathbf{E}rr_3(f)]$ over all $\rho^{2/3}$-planewise-permissible LDFs f which are not ρ-permissible.

Let us bound the probability of $\mathbf{E}rr_3(f)$ for such an LDF f. Since f is not ρ-permissible the probability that it satisfies $f(y) = \mathcal{A}[F(y)]$ is bounded by ρ, because y is a uniformly random point in $\mathcal{F}^{d(F)}$. The probability of $\mathbf{E}rr_3(f)$ is therefore bounded by ρ as well. Since f should be $\rho^{2/3}$-planewise-permissible, there can be at most $2\rho^{-2/3}$ such f's by Claim 4.9, and therefore a bound of $2\rho^{-2/3}\rho = O(\rho^{1/3})$ is obtained for the probability of $\mathbf{E}rr_2$. ■

Note that the statement of Claim 4.10 is similar to what we wish to establish (see the remark following Proposition 4.3), only for planes rather than $(k+2)$-dimensional subspaces in $\mathcal{S}ubSp(\mathcal{R})$. Claim 4.14 proves that by considering a random plane P, $y \in P \subseteq U$, in addition to the random subspace U and the random point $y \in U$. Claim 4.10 can be applied to P to obtain Claim 4.14, but for it to be applicable it should be shown that when a random subspace $U \in \mathcal{S}ubSp(\mathcal{R})$ is chosen , then a random plane P contained in U, and then a point $y \in P$, P and y are almost uniformly distributed. This is shown in the following claim.

Claim 4.11. *Let U be a random subspace in $\mathcal{S}ubSp(\mathcal{R})$, and let P be a random plane contained in U and y a random point in P. The distribution of P and y is almost uniform, that is if $\mathcal{P}\mathrm{lns}$ denotes the set of planes in $\mathcal{F}^{d(F)}$ then*

$$\sum_{P_0 \in \mathcal{P}lns} |\Pr(P = P_0) - |\mathcal{P}lns|^{-1}| \leq O(|\mathcal{F}|^{-1})$$

and

$$\sum_{y_0 \in \mathcal{F}^{d(F)}} |\Pr(y = y_0) - |\mathcal{F}|^{-d(F)}| \leq O(|\mathcal{F}|^{-1})$$

Proof. We only prove the second inequality. The proof of the first one is similar though more tedious.

Observe that since all the subspaces in $\mathcal{S}ubSp(\mathcal{R})$ contain U_0 the probability of the random point y in U to yield a specific point in U_0 is higher than $|\mathcal{F}|^{-d(F)}$, the probability of a uniformly random point, and the probability of y to yield a point outside of U_0 is smaller than $|\mathcal{F}|^{-d(F)}$. Hence

$$\begin{aligned}
&\sum_{y_0\in\mathcal{F}^{d(F)}} |\Pr[y=y_0]-|\mathcal{F}|^{-d(F)}| \\
&\quad= \sum_{y_0\in U_0}(\Pr[y=y_0]-|\mathcal{F}|^{-d(F)}) + \sum_{y_0\in\mathcal{F}^{d(F)}\setminus U_0}(|\mathcal{F}|^{-d(F)}-\Pr[y=y_0]) \\
&\quad< \Pr[y\in U_0] + \sum_{y_0\in\mathcal{F}^{d(F)}\setminus U_0}(|\mathcal{F}|^{-d(F)}-\Pr[y=y_0])
\end{aligned}$$

The total probability of y to belong to U_0 is $\Pr[y\in U_0]=|U_0|/|U|=|\mathcal{F}|^{-3}$. Additionally, note that y is uniformly distributed on $\mathcal{F}^{d(F)}\setminus U_0$. Hence for a point y_0 outside U_0 the probability that $y=y_0$ equals the probability of y to be outside U_0 divided by the size of $\mathcal{F}^{d(F)}\setminus U_0$, so

$$\Pr[y=y_0]=\frac{1-|\mathcal{F}|^{-3}}{|\mathcal{F}|^{d(F)}-|U_0|} > (1-|\mathcal{F}|^{-3})|\mathcal{F}|^{-d(F)} = |\mathcal{F}|^{-d(F)}-|\mathcal{F}|^{-d(F)-3}$$

Overall we obtain

$$\begin{aligned}
&\sum_{y_0\in\mathcal{F}^{d(F)}} \left|\Pr[y=y_0]-|\mathcal{F}|^{-d(F)}\right| < \\
&\qquad< |\mathcal{F}|^{-3}+|\mathcal{F}|^{d(F)}\cdot|\mathcal{F}|^{-d(F)-3} \;=\; 2|\mathcal{F}|^{-3} \;=\; O(|\mathcal{F}|^{-1})
\end{aligned}$$

■

Before we move to the final claim, we need the following two bounds. Claim 4.12, which appears in Chapter 3 as Proposition 3.5, bounds number of ρ-permissible LDFs. Claim 4.13 bounds the fraction of planes on which two distinct LDFs may agree.

Claim 4.12. *Suppose ρ satisfies the requirements of Lemma 4.6. Then there are less than $2/\rho$ ρ-permissible LDFs.*

Proof. Assume for the sake of contradiction that there exists a set $\mathcal{P}er$ containing $2/\rho$ distinct ρ-permissible LDFs. For each LDF $f\in\mathcal{P}er$ denote by $U(f)$ the set of points $y\in\mathcal{F}^{d(F)}$ such that f is the only LDF in $\mathcal{P}er$ satisfying $f(y)=\mathcal{A}(F[y])$.

Each LDF $f\in\mathcal{P}er$ is ρ-permissible, hence it agrees with $\mathcal{A}(F)$ on at least a ρ-fraction of the points. We bound from above the fraction of points for which it also agrees with other LDFs in $\mathcal{P}er$. Any other LDF in $\mathcal{P}er$ agrees with f on at most an $\frac{r(F)}{|\mathcal{F}|}$ fraction of the points, so overall f may agree with other LDFs in $\mathcal{P}er$ on at most a $\frac{2r(F)}{\rho|\mathcal{F}|}$ fraction of the points. From the assumption $\rho > \left(\frac{r(F)}{|\mathcal{F}|}\right)^{c_g} d(F)$ it follows in particular that $\rho^2 > 4r(F)/|\mathcal{F}|$ (recall that $c_g<1/2$ and that we assume $d(F)\geq k+2>2$), and therefore $\frac{2r(F)}{\rho|\mathcal{F}|}<\frac{1}{2}\rho$.

f thus agrees with $\mathcal{A}(F)$ on at least ρ of the points, and on less than a $\frac{1}{2}\rho$ fraction of the points it agrees with other LDFs in $\mathcal{P}er$. It follows that for every $f \in \mathcal{P}er$, $U(f)$ contains more than a $\frac{1}{2}\rho$ fraction of the points. Since the sets $U(f)$ are disjoint, it follows that the fraction of all points contained in any of the $U(f)$'s is greater than $\frac{1}{2}\rho \cdot |\mathcal{P}er| \geq 1$. This is a contradiction. ■

Claim 4.13. *For any $t > 0$, two distinct $[r,t]$-LDFs must disagree on all but at most an $r/|\mathcal{F}|$ fraction of their possible restrictions to planes.*

Proof. Let f and g be two distinct r-degree LDFs over $\mathcal{F}^t$, and let P be a random plane in $\mathcal{F}^t$. We are to evaluate the probability that $f|_P$ equals $g|_P$. Let y be a random point on P. y is uniformly distributed in $\mathcal{F}^t$, and therefore it produces a disagreement with probability $1 - \frac{r}{|\mathcal{F}|}$ (this is a well known property of LDFs). Since y can produce a disagreement only in the case that there is a disagreement over P, it implies that there is a disagreement over P with probability at least $1 - \frac{r}{|\mathcal{F}|}$. ■

The following claim directly implies Proposition 4.3.

Claim 4.14. *Suppose ρ satisfies the requirements of Lemma 4.6. Let U be a random affine subspace in $\mathcal{S}ub\mathcal{S}p(\mathcal{R})$, and y be a random point in U. Let $\mathbf{E}rr$ be the event that $g_{_U}(y) = \mathcal{A}(F[y])$, yet there is no ρ-permissible LDF (with respect to the assignment of F) whose restriction to U gives $g_{_U}$. Then $\Pr[\mathbf{E}rr] = O(\rho^{1/3})$.*

Proof. Without loss of generality, we assume that P is a random plane contained in (the random subspace) U, and y is a random point in P.

We define two events $\mathbf{E}rr_1$ and $\mathbf{E}rr_2$ such that $\mathbf{E}rr_1 \cup \mathbf{E}rr_2$ contains $\mathbf{E}rr$, and bound the probability of each by $O(\rho^{1/3})$. Let $\mathbf{E}rr_1$ be the event that $g_{_U}(y) = \mathcal{A}(F[y])$, yet there is no ρ-permissible LDF f that agrees with $g_{_U}$ on P, namely $f|_P = g_{_U}|_P$. Let $\mathbf{E}rr_2$ be the event that there is no ρ-permissible LDF whose restriction to U gives $g_{_U}$, yet there exists such an LDF that agrees with $g_{_U}$ on P. Obviously $\mathbf{E}rr \subseteq \mathbf{E}rr_1 \cup \mathbf{E}rr_2$.

Bounding $\Pr[\mathbf{E}rr_2]$**.** For a ρ-permissible LDF f, let $\mathbf{E}rr_3(f)$ be the event that $f|_U \neq g_{_U}$, yet $f|_P = g_{_U}|_P$. $\mathbf{E}rr_2$ is contained in the union of the events $\mathbf{E}rr_3(f)$ over all ρ-permissible LDFs f. For a ρ-permissible LDF f,

$$\Pr[\mathbf{E}rr_3(f)|U = V] \leq r(F)/|\mathcal{F}|$$

for every fixed subspace $V \in \mathcal{S}ub\mathcal{S}p(\mathcal{R})$, by applying Claim 4.13 to V. It follows that $\Pr[\mathbf{E}rr_3(f)]$ is bounded by $r(F)/|\mathcal{F}|$ as well. Since there are less than $2/\rho$ ρ-permissible LDFs in all, we obtain that

$$\Pr[\mathbf{E}rr_2] < \frac{2r(F)}{\rho|\mathcal{F}|} < \rho$$

(the last inequality follows easily from the restriction on ρ).

Bounding $\Pr[\mathbf{E}rr_1]$**.** We change the distribution of U, P and y, by letting P be a random plane in $\mathcal{F}^{d(F)}$, U be a random space in $\mathcal{S}ubSp(\mathcal{R})$ that contains P, and y be a random point in P. By Claim 4.11, the statistical distance between the new distribution of P, which is uniform, and the original distribution is $O(|\mathcal{F}|^{-1})$. Under both the original and the new distributions, the distribution of U and y conditioned on P being a fixed plane P_0 are the same – U is a random space in $\mathcal{S}ubSp(\mathcal{R})$ that contains P_0 and y is a random point in P_0. It follows that the statistical distance between the new joint distribution of U, P and y, and the original distribution is bounded by $O(|\mathcal{F}|^{-1})$. It is hence enough to bound the probability of $\mathbf{E}rr_1$ according to the new distribution.

Let $g_P \doteq g_U|_P$ be considered as a random plane-assignment for the (random) plane P. The definition of $\mathbf{E}rr_1$ can hence be articulated as the event that $g_P(y) = \mathcal{A}(F[y])$, yet there is no ρ-permissible LDF f such that $g_P = f|_P$. Claim 4.10 naturally extends to the case where the plane-assignments is random as long as the assignment to F is not random, hence it implies that $\Pr[\mathbf{E}rr_2] = O(\rho^{1/3})$ (note that P is a uniformly random plane and y is a random point in P). ■

4.2 Overview of the CR-Constructor

Let us give an overview of the CR (Composition-Recursion) constructor. Given a domain and a tuple, the CR constructor generates a constant-length sequence of restricted LDF-readers that ends with the final, unrestricted, CR LDF-reader. Each transformation of an LDF-reader $\mathcal{R}$ in the sequence into the next (except for the final one) has the same two steps as follows.

Extension. In the first step, an *extension-procedure* is applied to each active domain of $\mathcal{R}$, replacing it by a domain with greatly reduced degree parameters in the price of an increased dimension parameter. The active degree and dimension parameters of $\mathcal{R}$ are thus changed, but its other properties are maintained.

Composition. The second step is the application of the *composition* procedure, which incorporates new LDF-readers into $\mathcal{R}$. First, it generates several new LDF-readers using the SP constructor (actually, any constructor with properties as in Lemma 4.2 will do), applying it to different active domains and tuples. The domains generated in the process then become active instead of the old active domains. These new active domains are of constant dimension, and because of the extension step their degree parameters are greatly reduced with respect to the active domains of $\mathcal{R}$. Finally the newly generated local-readers are plugged into the local-readers of $\mathcal{R}$, generating the next LDF-reader in the sequence.

We proceed as follows. First, in the next section, we give a formal definition of an extension and show the two extension-procedures used by the CR constructor. In Section 4.4

we describe the composition procedure and prove its properties. Then, in Section 4.5 we describe the CR constructor and prove its correctness in Section 4.6.

4.3 Extensions

An extension of a domain F is a domain G which *contains* the variables of F: Each variable $F[x]$ is endowed with another name $G[\phi(x)]$. The function $\phi : \mathcal{F}^{d(F)} \to \mathcal{F}^{d(G)}$ is called the *gluing* of F to G. The extension must preserve good and feasible assignments as follows.

Definition 4.15 (extension). *Let F be a domain, and let G be a domain that contains the variables of F. G is called an* extension *of F if the following properties hold:*

- *Extension Property: Any good assignment $\mathcal{A}$ for F can be extended to a good assignment for G, called the* encoding-assignment *or the* encoding LDF *of $\mathcal{A}$.*
- *Restriction Property: The restriction to F of any feasible assignment for G is a feasible assignment for F.*

The point about extensions is that they allow the representation of an LDF assigned to a domain F by an encoding LDF with different properties. We can hence replace the active domains of a restricted (ρ, ϵ)-LDF-reader $\mathcal{R}$ by their extensions and obtain a restricted (ρ, ϵ)-LDF-reader where the active degree parameters are different, usually considerably smaller.

Proposition 4.16 (extension). *Let $\mathcal{R}$ be a restricted (ρ, ϵ)-LDF-reader, evaluating a tuple in a domain F. Suppose that for each active domain G of $\mathcal{R}$, $e(G)$ is an extension of G, and that all extensions have the same parameters. If an LDF-reader $\mathcal{R}'$ is obtained from $\mathcal{R}$ by just declaring these extensions as the active domains of $\mathcal{R}'$ then $\mathcal{R}'$ is a restricted (ρ, ϵ)-LDF-reader.*

Proof. It is given that all the active domains of $\mathcal{R}'$ have the same parameters. To show that $\mathcal{R}'$ is a valid restricted LDF-reader we define an encoding-assignment of $\mathcal{R}'$, for every good assignment to F.

Given a good assignment for F, let $\mathcal{A}$ be its encoding-assignment with respect to $\mathcal{R}$. For each active domain G of $\mathcal{R}$, assign the encoding-assignment of $\mathcal{A}(G)$ to its extension $e(G)$. This obtains a good assignment for the representation of $\mathcal{R}'$, and since the assignments to the variables of $\mathcal{R}$ are not changed all local-tests are satisfied and all local-readers return evaluations consistent with the assignment of F.

The fact that $\mathcal{R}'$ has parameters (ρ, ϵ) follows easily from the restriction property of each extension $e(G)$, which implies that the restriction of a feasible assignment for $\mathcal{R}'$ yields a feasible assignment for $\mathcal{R}$. ■

Extension-procedures. An extension-procedure is an algorithm which given a domain F, generates an extension G of F. The running time of the algorithm must be polynomial in $|\mathcal{F}|^{d(G)}$. We next show the two extension-procedures used by the CR constructor – the power-substitution and the linearization extension-procedures. In the sequence of restricted LDF-readers that is generated by the CR constructor (see the overview above), the power-substitution extension-procedure is used in the generation of all restricted LDF-readers but the last. The last restricted LDF-reader is generated using the linearization extension-procedure, and thus has active domains with linear lower-degree and upper-degree. The final, unrestricted, CR LDF-reader is obtained by replacing each such domain with variables that represent the coefficients of a linear function.

Given a domain F, the power-substitution extension-procedure constructs an extension G with greatly reduced degree parameters in the price of increasing the dimension parameter. The linearization extension-procedure, when applied to a domain F, yields a domain G with linear lower-degree and linear upper-degree. The dimension of G is, however, exponential in the degree parameters of F, hence the linearization is applied by the CR constructor only after very small active degree parameters are achieved.

Gap consumption. The power-substitution extension-procedure has the following property. Suppose G is the extension of a domain F, obtained by the power-substitution extension-procedure. Then $r(G)/s(G) < r(F)/s(F)$, namely some of the gap between the lower-degree and the upper-degree parameters is consumed by applying the power-substitution extension-procedure. The CR constructor applies the power-substitution extension-procedure several times as described in the overview, hence if it is applied to a domain F where the gap between the upper-degree and lower-degree is not large enough (see Lemma 3.3), domains are created where the upper-degree is smaller than the lower-degree. Since the linearization extension-procedure is not applicable to such domains, the CR constructor would not be able to construct an LDF-reader for F.

The power-substitution extension-procedure

We begin by stating the properties of the power-substitution extension-procedure. For simplicity, we omit floor and ceiling signs where they are not essential.

Proposition 4.17 (power-substitution). *There exists an extension-procedure, called* power-substitution*, which given a domain F and a parameter $b > 1$, generates an extension G of F with the following parameters: For $t \doteq \lfloor \log_b(s(F)+1) \rfloor$,*

- $d(G) = d(F)t$
- $s(G) = d(F)t(b-1)$

- $r(G) = r(F)/b^t \ \big(\geq r(F)/s(F)\big)$

The procedure is based on the idea that by replacing powers of variables in an LDF f with new auxiliary variables, the degree of f may be decreased dramatically. For example, we fix an LDF over one variable $f(u_1) = u_1^{12} + u_1^{25}$ (the handling of multi-variate LDFs is very similar), and show an encoding LDF g over three variables.

g is obtained from f by substituting powers of u_1 with new variables. Informally speaking, if v_0 is considered as representing u_1, v_1 is considered as representing u_1^3, and v_2 – as representing u_1^9, then $u_1^{12} = v_1 v_2$, and $u_1^{25} = v_0 v_1^2 v_2^2$ (note that we used the base 3 representation of 12 and 25). Replacing these terms in f obtains an LDF $g(v_0, v_1, v_2) = v_1 v_2 + v_0 v_1^2 v_2^2$ of degree 5 rather than 25. g encodes and extends f in the sense that $g(u_1, u_1^3, u_1^9) = f(u_1)$ for every $u_1 \in \mathcal{F}$.

For a domain G, obtained from a domain F using the power-substitution extension-procedure with parameter b, an LDF f of degree $s(F)$ assigned to F is encoded by an LDF g over $\mathcal{F}^{d(G)}$ as follows. g is obtained from f by taking an auxiliary variable for each power of the form $u_i^{b^e}$ of a variable u_i of f. Any other power u_i^j of u_i can then be replaced by a monomial over the new variables of degree at most $b-1$ in each variable, using the base-b representation of j.

Proof of Proposition 4.17: We begin by describing the power-substitution extension-procedure and then prove that it has the required properties.

The procedure. Given a domain F and a parameter b, the procedure first generates a domain G with parameters as stated in the proposition. It then generates a gluing function $\phi : \mathcal{F}^{d(F)} \rightarrow \mathcal{F}^{d(G)}$ as follows:
For every $x = (u_1, \ldots, u_{d(F)}) \in \mathcal{F}^{d(F)}$, $\phi(x)$ is defined to be

$$(u_1, u_1^b, u_1^{b^2}, \ldots, u_1^{b^t}, \quad u_2, u_2^b, u_2^{b^2}, \ldots, u_2^{b^t}, \quad \ldots \quad , u_{d(F)}, \ldots, u_{d(F)}^{b^t}) \ \in \mathcal{F}^{d(G)}$$

Finally, each variable of the form $G[\phi(x)]$ is discarded, and the name $G[\phi(x)]$ is endowed to the variable $F[x]$ (which now has more than one name).

It is clear that the above procedure generates a domain G with the required parameters, in time polynomial in $|\mathcal{F}|^{d(G)}$. It remains to show that G is indeed an extension of F, namely that it has the extension and restriction properties.

Extension property. Suppose F is assigned an $[s(F), d(F)]$-LDF f (namely a good assignment). We now show its encoding LDF g – it should be an $[s(G), d(G)]$-LDF satisfying $g \circ \phi = f$, so that when assigned to G it does not conflict with F. First, let $f(u_1, \ldots, u_{d(F)})$ be written as a polynomial formula P over the variables $u_1, \ldots, u_{d(F)}$. P is transformed into a polynomial formula P' over the variables

$$v_{(1,0)}, v_{(1,1)}, .., v_{(1,t-1)},\ v_{(2,0)}, v_{(2,1)}, .., v_{(2,t-1)},\quad \cdots\quad , v_{(d(F),0)}, .., v_{(d(F),t-1)}$$

by replacing each term u_i^j in P with a monomial $m_{(i,j)}$ over $v_{(i,0)}, .., v_{(i,t-1)}$: Since the term u_i^j appears in P we gather that $j \leq s(F)$, and hence its representation as a number in base b has at most t digits. Let $e_{t-1}, \ldots, e_1, e_0$ be the base b representation of j, and let

$$m_{(i,j)} \doteq (v_{(i,0)})^{e_0} (v_{(i,1)})^{e_1} \ldots (v_{(i,t-1)})^{e_{t-1}}$$

Replacing each term u_i^j in P with the monomial $m_{(i,j)}$ we obtain P', and then we define g by

$$g(v_{(i,0)}, \ldots, v_{(d(F),t-1)}) \doteq P'(v_{(i,0)}, \ldots, v_{(d(F),t-1)})$$

Since each monomial $m_{(i,j)}$ is of degree at most $t(b-1)$, it easily follows that g is an $[s(G), d(G)]$-LDF. Considering $m_{(i,j)}$ as a function over $\mathcal{F}^{d(G)}$, it is also easy to see that for all $(u_1, \ldots, u_{d(F)})$, $\quad (m_{(i,j)} \circ \phi)(u_1, \ldots, u_{d(F)}) = u_i^j$. It follows that $g \circ \phi = f$, and hence g is indeed an encoding-LDF.

Restriction property. Suppose G is given a feasible assignment, namely it is assigned an $[r(G), d(G)]$-LDF g. The restriction of the assignment to F is hence an LDF f over $\mathcal{F}^{d(F)}$, given by $f = g \circ \phi$. The degree of f is at most $\deg(f) = \deg(g)\deg(\phi) = r(G)b^t = r(F)$. The restriction is hence a feasible assignment for F, as required.

The linearization extension-procedure

The linearization extension-procedure is very similar to the power-substitution procedure. The idea is to encode an LDF f by a *linear* LDF, replacing every monomial by a new auxiliary variable (recall that in the power-substitution, auxiliary variables where only introduced for some powers of variables in f). Since many auxiliary variables are used, the dimension increases dramatically.

Proposition 4.18 (linearization). *There exists an extension-procedure called* linearization*, which given a domain F with $s(F) \leq r(F)$, generates an extension G with the following parameters: For $t \doteq \binom{s(F)+d(F)}{d(F)}$,*

- $d(G) = t$
- $s(G) = 1$
- $r(G) = 1$

Proof. We begin by describing the linearization extension-procedure, and then prove that it has the required properties.

The procedure. Given a domain F, the procedure first generates a domain G with parameters as stated above. To generate the gluing function, the procedure picks an arbitrary enumeration $m_1, \ldots, m_t$ of the monomial functions of degree at most $s(F)$ over $\mathcal{F}^{d(F)}$ (note that there are exactly t such monomials). The gluing function $\phi : \mathcal{F}^{d(F)} \to \mathcal{F}^{d(G)}$ is then defined by

$$\forall\, x \in \mathcal{F}^{d(F)} \qquad \phi(x) \doteq (m_1(x), \ldots, m_t(x))$$

Having defined the gluing function, F and G are then "glued" in the usual way – each variable of the form $G[\phi(x)]$ is discarded, and the name $G[\phi(x)]$ is endowed to the variable $F[x]$ (which now has more than one name).

The procedure clearly generates a domain G with the required parameters, in time polynomial in $|\mathcal{F}|^{d(G)}$. It remains to verify that G has the extension and restriction properties.

Extension property. Suppose F is assigned an $[s(F), d(F)]$-LDF f, and let us construct its encoding LDF – a linear LDF over $\mathcal{F}^{d(G)}$ satisfying $g \circ \phi = f$. First, one can write f as a linear-combination of the monomial functions of degree at most $s(F)$:

$$f = \sum_{i=1}^{t} \gamma_i m_i$$

g is then defined by

$$\forall\, (v_1, \ldots, v_t) \in \mathcal{F}^{d(G)} \qquad g(v_1, \ldots, v_t) \doteq \sum_{i=1}^{t} \gamma_i v_i$$

It is clear that $g \circ \phi = f$, as desired.

Restriction property. Suppose G is given a feasible assignment, namely it is assigned a linear LDF g. The restriction of the assignment to F is the LDF $f = g \circ \phi$. Since ϕ is of degree $s(F)$ and g is linear, the degree of f is at most $s(F) \le r(F)$, as required. ■

4.4 The Composition Procedure

We now turn to describe the composition procedure of the CR constructor algorithm. It takes as input a restricted LDF-reader $\mathcal{R}$, and generates a restricted LDF-reader $\mathcal{R}'$ with the same active degree parameters, but where the dimension of the active domains is constant.

Suppose an LDF-reader $\mathcal{R}$ is given, which evaluates a tuple $(u_1, \ldots, u_k)$ in a domain F. The composition procedure has two main steps: First it generates new LDF-readers using the SP constructor as a sub-procedure, and then it incorporates them into $\mathcal{R}$.

Uniformization. Recall that each local-reader L of $\mathcal{R}$ has variables from only one active domain, $\mathrm{Dom}_\star(L)$. Before the main two steps, it is convenient to make sure that all local-readers L in $\mathcal{R}$ have the same number of *active variables*, namely variables from $\mathrm{Dom}_\star(L)$. Denoting the maximal number of active variables in a local-reader of $\mathcal{R}$ by t, the composition procedure adds arbitrary variables to local-readers so that all have t active variables (the variables may be added anywhere in the local-reader, with zero coefficients).

Generating new LDF-readers. For each local-reader L in $\mathcal{R}$, the procedure now generates an LDF-reader denoted $\mathcal{R}_L$ as follows. If G is the active domain of L and $G[x_1], \ldots, G[x_t]$ are its active variables, then $\mathcal{R}_L$ is generated by calling the SP constructor with parameters G and $(x_1, \ldots, x_t)$.

Domain incorporation. The composition procedure now incorporates the domains of the new LDF-readers into $\mathcal{R}$: The newly generated domains are added to the representation. The active domains of $\mathcal{R}$ cease to be active – the active domains of $\mathcal{R}'$ are the active domains of the newly generated LDF-readers.

Local-reader incorporation. In a feasible assignment for $\mathcal{R}'$, the active variables of $\mathcal{R}$-local-readers L are no longer promised to be assigned the evaluation of a single feasible LDF over $\mathrm{Dom}_\star(L)$. Instead, these variables are replaced by the evaluators of local-readers of $\mathcal{R}_L$, since their values are supposedly the evaluations of one of the (not many) permissible LDFs over $\mathrm{Dom}_\star(L)$.

For each pair of local-readers, L of $\mathcal{R}$ and M from $\mathcal{R}_L$, the composition procedure generates a local-reader of $\mathcal{R}'$, denoted by $L \circ M$, as follows. Let G denote the active domain of L, and let $G[x_1], \ldots, G[x_t]$ denote its active variables. To obtain $L \circ M$ each variable $G[x_i]$ in the evaluator or the local-test of L is replaced by the i'th evaluator of M, and then the local-test of M is added in conjunction to the local-test of L (where the $G[x_i]$'s have been replaced).

Properties of the composition procedure

We now analyze the properties of the composition procedure that are important for its application by the CR constructor – the time it takes, and the properties and parameters of the LDF-readers it generates. In the analysis we assume that the composition procedure is applied to LDF-readers where the the number of variables in each local-reader and the number of conjunctions in each local-test is bounded by a constant, since the CR constructor indeed applies it to such LDF-readers. Notice that under this assumption it is clear that the composition procedure generates LDF-readers where the number of variables in each local-reader and the number of conjunctions in each local-test is also bounded by a (different) constant.

Time. When applied to an LDF-reader $\mathcal{R}$, the composition procedure applies the SP constructor several times. Each call to the SP constructor takes time polynomial in $|\mathcal{F}|^{d_\star(\mathcal{R})}$, according to the definition of a constructor (note that this is polynomial the number of variables in each of the active domains of $\mathcal{R}$). Since the number of calls to the SP constructor equals the number of local-readers in $\mathcal{R}$, it follows that overall the composition procedure takes time polynomial in the size of $\mathcal{R}$.

Encoding-assignments. When the composition procedure is applied to an LDF-reader $\mathcal{R}$ that evaluates a tuple $(x_1, \ldots, x_k)$ in a domain F, the resulting structure $\mathcal{R}'$ has representation-variables and local-readers. To be a valid LDF-reader, we show that for every good assignment $\mathcal{A}$ for F there is an encoding-assignment with respect to $\mathcal{R}'$: First extend $\mathcal{A}$ to an encoding-assignment $\mathcal{A}'$ for $\mathcal{R}$. In particular $\mathcal{A}'$ assigns a good assignment to the active domain $\mathrm{Dom}_\star(L)$ of each local-reader L in $\mathcal{R}$. Then extend the assignment of each active domain $\mathrm{Dom}_\star(L)$ to an encoding-assignment with respect to $\mathcal{R}_L$. This obtains an assignment for all the variables of $\mathcal{R}'$. It is easy to verify that it is an encoding-assignment of $\mathcal{A}$ with respect to $\mathcal{R}'$.

Parameters of $\mathcal{R}'$. Given an LDF-reader $\mathcal{R}$, the composition procedure generates an LDF-reader $\mathcal{R}'$. The parameters of $\mathcal{R}'$ can be computed from the parameters of $\mathcal{R}$ according to the following composition proposition.

Proposition 4.19 (composition). *Let $\mathcal{R}$ be a (ρ, ϵ)-LDF-reader where $\epsilon^{3/4} > (r_\star(\mathcal{R})/|\mathcal{F}|)^{c_g} d_\star(\mathcal{R})$. Then the LDF-reader $\mathcal{R}'$, generated from $\mathcal{R}$ by the composition procedure, has parameters $(\rho, \epsilon^{1/4})$.*

Before the formal proof is given, we describe its main ideas. There are two types of ρ-erroneous local-readers $L \circ M$. One is where M is $\epsilon^{3/4}$-erroneous – this happens for at most an $O(\epsilon^{1/4})$ fraction of the local-readers since the $\mathcal{R}_L$'s are $(\epsilon^{3/4}, O(\epsilon^{1/4}))$-LDF-readers.

In case M is not erroneous, its evaluators yield evaluations of an $\epsilon^{3/4}$-permissible LDF f with respect to the assignment of $\mathrm{Dom}_\star(L)$. $L \circ M$ is thus ρ-erroneous if and only if L remains ρ-erroneous when $\mathrm{Dom}_\star(L)$ is assigned the feasible LDF f. Since for any feasible assignment at most an ϵ-fraction of the local-readers of $\mathcal{R}$ may be ρ-erroneous, and since the number of $\epsilon^{3/4}$-permissible LDFs for every domain is less than $2\epsilon^{-3/4}$, a counting argument implies that the fraction of local-readers $L \circ M$ where M is not erroneous is bounded by $2\epsilon^{-3/4} \cdot \epsilon = O(\epsilon^{1/4})$.

Proof of Proposition 4.19:

Fix a feasible assignment $\mathcal{A}'$ for the representation of $\mathcal{R}'$ and a parameter $\rho' \doteq \epsilon^{3/4}$, and let us divide the ρ-erroneous local-readers of $\mathcal{R}'$ into two sets according to ρ' – the *periferal-erroneous* local-readers are the local-readers $L \circ M$ where M is ρ'-erroneous as a local-reader of $\mathcal{R}_L$, and the *core-erroneous* are those where M is not ρ'-erroneous. We bound the fraction of both types of local-readers by $O(\epsilon^{1/4})$.

Periferal-erroneous local-readers. Since $\rho' > (r_\star(\mathcal{R})/|\mathcal{F}|)^{c_g} d_\star(\mathcal{R})$, Lemma 4.2 implies that every LDF-reader $\mathcal{R}_L$ generated by the composition procedure has parameters $(\rho', O((\rho')^{1/3}))$, so the fraction of ρ'-erroneous local-readers in it is $O((\rho')^{1/3}) = O(\epsilon^{1/4})$. Hence for every local-reader L of $\mathcal{R}$, the fraction of periferal-erroneous local-readers among local-readers of the form $L \circ M$ is $O(\epsilon^{1/4})$, and therefore the overall fraction of periferal-erroneous local-readers in $\mathcal{R}'$ is bounded by $O(\epsilon^{1/4})$ as desired.

We move to bound the fraction of core-erroneous local-readers. We first show that in such a local-reader $L \circ M$, L has to be erroneous as a local-reader of $\mathcal{R}$ with respect to a certain class of assignments, as explained below.

The assignments $\mathcal{A}(G, g)$ for $\mathcal{R}$. For an active domain G of $\mathcal{R}$ and an $[r(G), d(G)]$-LDF g, we define a class $\mathcal{A}(G, g)$ of assignments for $\mathcal{R}$, based on $\mathcal{A}'$. The elements of $\mathcal{A}(G, g)$ are the assignments for $\mathcal{R}$ that assign g to G, and that are equal to $\mathcal{A}'$ on all domains of $\mathcal{R}$ which are not active. Active domains of $\mathcal{R}$ other than G may be assigned arbitrarily. A local-reader L of $\mathcal{R}$ with $\mathrm{Dom}_\star(L) = G$, may be either ρ-erroneous with respect to *all* assignments in $\mathcal{A}(G, g)$, or with respect to *none*, because the assignments in $\mathcal{A}(G, g)$ are all equal on the variables of L (recall that L cannot have variables from active domains other than G).

Consider a local-reader $L \circ M$ that is core-erroneous. The evaluators of M yield values consistent with an LDF g, which is ρ'-permissible with respect to the assignment of $G \doteq \mathrm{Dom}_\star(L)$. It follows that as a local-reader of $\mathcal{R}$, L is ρ-erroneous with respect to the assignments in $\mathcal{A}(G, g)$ – these assignments yield the same values for the variables of G as the evaluators of M, and give the same values as $\mathcal{A}'$ to all the other variables of L.

Core-erroneous local-readers. Let G be an active domain of $\mathcal{R}$. Denote by $\alpha(G, g)$ the fraction among $\mathcal{R}$-local-readers, of local-readers whose active domain is G and which are ρ-erroneous with respect to the assignments in $\mathcal{A}(G, g)$. Denote by $\alpha(G)$ the maximum over all $\alpha(G, g)$.

Let $\mathcal{A}$ be the assignment obtained from $\mathcal{A}'$ by assigning to each active domain G of $\mathcal{R}$ an LDF g such that $\alpha(G, g)$ is maximized. Then for every G, the fraction of $\mathcal{R}$-local-readers whose active domain is G, and which are ρ-erroneous with respect to $\mathcal{A}$ is $\alpha(G)$. The parameters of $\mathcal{R}$ imply that the total fraction of ρ-erroneous local-readers is bounded by ϵ, hence $\sum_G \alpha(G) \leq \epsilon$.

For an active domain G of $\mathcal{R}$ we denote by $\gamma(G)$ the fraction of local-readers L in $\mathcal{R}$ whose active domain is G, and for which there exists a local-reader M in $\mathcal{R}_L$ where $L \circ M$ is core-erroneous. We have seen that for an $\mathcal{R}$-local-reader L to be accounted in $\gamma(G)$, it must be ρ-erroneous with respect to the assignments in $\mathcal{A}(G, g)$, where g is ρ'-permissible. $\gamma(G)$ is therefore bounded by the sum $\sum \alpha(G, g)$ taken over all permissible LDFs g, and so by $\alpha(G)$ times the number of ρ'-permissible LDFs. By Proposition 3.5 the number of ρ'-permissible LDFs is less than $2/\rho'$, hence $\gamma(G) < (2/\rho')\alpha(G)$, and we obtain that

$$\sum_G \gamma(G) < (2/\rho') \sum_G \alpha(G) \leq 2\epsilon/\rho' = 2\epsilon^{1/4}$$

Namely the fraction of $\mathcal{R}$-local-readers L for which there exists a core-erroneous local-reader $L \circ M$ is bounded by $O(\epsilon^{1/4})$.

We show that this also bounds the total fraction of core-erroneous local-readers: Note that there is the same number of local-readers in every LDF-reader of the form $\mathcal{R}_L$ – this follows from the definition of a constructor, together with the fact that all active domains of $\mathcal{R}$ have the same degree parameters. Hence for each local-reader L of $\mathcal{R}$ there is the same number of local-readers of the form $L \circ M$ in $\mathcal{R}'$. A simple counting-argument now shows that the fraction of core-erroneous local-readers is also bounded by $O(\epsilon^{1/4})$, as desired.

4.5 The CR Constructor

It is now the time to describe the actual CR constructor, proving the Composition-Recursion Constructor Lemma (Lemma 3.3). Let F be a domain, and let $(x_1, \ldots, x_k)$ be a k-tuple of points in $\mathcal{F}^{d(F)}$ (where, as in Lemma 3.3, k is a constant). We assume, under the notation as specified in Lemma 3.3, that $d(F) = O(\log^{1-\beta} n)$, $s(F) \leq |\mathcal{F}|^{c_1}$, and $r(F) \geq |\mathcal{F}|^{c_2}$. For simplicity we reset $s(F)$ and $r(F)$ so that the latter inequalities hold as equalities – note that a (ρ, ϵ)-LDF-reader with respect to the new degree parameters remains a (ρ, ϵ)-LDF-reader if $s(F)$ is reduced and $r(F)$ is increased to their original values.

The CR constructor generates an (unrestricted) LDF-reader $\mathcal{R}$ evaluating $(x_1, \ldots, x_k)$ in F. First, it generates a sequence $\mathcal{R}_0, \ldots, \mathcal{R}_K$, where $K = O(\frac{1}{1-\beta})$ is a constant that will be chosen later, of restricted LDF-readers. The transformation of each $\mathcal{R}_i$ into $\mathcal{R}_{i+1}$ is accomplished in two steps. At first a restricted LDF-reader $\mathcal{R}'_i$ is generated by applications of an extension-procedure to the active domains of $\mathcal{R}_i$, as described in the extension proposition (Proposition 4.16). The degree parameters of $\mathcal{R}'_i$ are decreased with respect to $\mathcal{R}_i$ but the dimension is increased. $\mathcal{R}_{i+1}$ is then generated by applying the composition procedure to $\mathcal{R}'_i$, thus the degree parameters remains the same while the active dimension parameter becomes constant. Finally $\mathcal{R}_K$ has a constant active dimension and both of its active degree parameters are 1, hence in a good or feasible assignment each active domain of $\mathcal{R}_K$ is assigned a constant-dimensional linear function. The final unrestricted LDF-reader is obtained by replacing each active domain of $\mathcal{R}_K$ by a constant number of variables that represent the coefficients of a linear function over it.

We now fully describe the generation of the sequence $\mathcal{R}_0, \ldots, \mathcal{R}_K$, and the transformation of $\mathcal{R}_K$ into $\mathcal{R}$. We then show that the CR constructor has the properties required by Lemma 3.3.

Generating $\mathcal{R}_0$. To generate the first restricted LDF-reader, $\mathcal{R}_0$, the CR constructor applies the SP constructor to the domain F and the tuple $(x_1, \ldots, x_k)$.

Generating $\mathcal{R}_1, \ldots, \mathcal{R}_{K-1}$. From $\mathcal{R}_0$ the CR constructor continues to iteratively generate restricted LDF-readers as follows. Having generated $\mathcal{R}_i$, the constructor transforms it into a restricted LDF-reader $\mathcal{R}'_i$ by applying the power-substitution extension-procedure to each active domain of $\mathcal{R}_i$ with parameter

$$b = \max\left\{ \left(s_\star(\mathcal{R}_i) + 1\right)^{1/\log^{1-\beta} n} \ , \ 2 \right\}$$

and taking these extensions to be the active domains of $\mathcal{R}'_i$. The constructor then generates $\mathcal{R}_{i+1}$ by applying the composition procedure to $\mathcal{R}'_i$. The CR constructor iteratively generates LDF-readers as described above until finally an LDF-reader $\mathcal{R}_{K-1}$ is generated such that

$$\binom{s_\star(\mathcal{R}_{K-1}) + d_\star(\mathcal{R}_{K-1})}{d_\star(\mathcal{R}_{K-1})} \leq \log^{1-\beta} n$$

As proven below, this occurs for a constant K.

Generating $\mathcal{R}_K$. The transformation of $\mathcal{R}_{K-1}$ into $\mathcal{R}_K$ is carried similarly to the previous transformations described above, only that $\mathcal{R}'_{K-1}$ is generated using the linearization extension-procedure instead of the power-substitution extension-procedure. Note that for the linearization extension-procedure to be applicable the active lower-degree parameter of $\mathcal{R}_{K-1}$ must not be greater than its active upper-degree. We show below that this indeed holds.

Generating $\mathcal{R}$. The constructor now transforms $\mathcal{R}_K$ into the final CR LDF-reader. Having used the linearization extension-procedure to produce $\mathcal{R}'_{K-1}$, we gather that the active lower-degree of $\mathcal{R}_K$ (which equals that of $\mathcal{R}'_{K-1}$) is 1. Its active dimension, $d \doteq d_\star(\mathcal{R}_K)$, is constant since $\mathcal{R}_K$ is generated by the composition procedure. A good assignment to an active domain G of $\mathcal{R}_K$ is thus a *linear* LDF f, that can be represented using a constant number of coefficient γ_i by

$$\forall \ (u_1, \ldots, u_d) \in \mathcal{F}^d \qquad g(u_1, \ldots, u_d) \doteq \sum_{i=1}^{d} \gamma_i u_i$$

The CR constructor adds d variables to the representation, $G_1, \ldots, G_d$, for each active domain G of $\mathcal{R}_K$, to represent the coefficients $\gamma_1, \ldots, \gamma_d$ above. It then goes over all the local-readers and replaces every term $G[(u_1, \ldots, u_d)]$, where G is an active-LDF, by $\sum_{i=1}^{d} G_i u_i$. It is now possible to deactivate or even remove the active domains altogether (their variables no longer appear anywhere), thus completing the generation of $\mathcal{R}$.

4.6 The CR Constructor Works

Below it is proven that the CR constructor above has the properties stated in Lemma 3.3. We show that it stops after a constant number of iterations as stated above, and that it takes polynomial time. It is then shown that although each transformation of $\mathcal{R}_i$ into $\mathcal{R}_{i+1}$ consumes some of the lower-degree to upper-degree gap, the active upper-degree of $\mathcal{R}_{K-1}$ is not smaller than the active lower-degree (hence linearization extension-procedure is correctly used by the CR constructor). We conclude by showing that for an appropriate constant $c > 0$, the constructor generates $(\rho, O(\rho^c))$-LDF-readers for all ρ's such that $\rho > (r/|\mathcal{F}|)^{c_g} d$.

First of all observe that as noted in the description of the properties of the composition procedure, for every constant i both the number of variables and the number of conjunctions in each local-reader are bounded by a constant.

The number of iterations is constant.

It should be shown that for some constant $K = O(\frac{1}{1-\beta})$, the parameters of the $(K-1)$'th element in the sequence $\mathcal{R}_0, \mathcal{R}_1, \ldots$ of LDF-readers satisfy

$$\binom{s_\star(\mathcal{R}_{K-1}) + d_\star(\mathcal{R}_{K-1})}{d_\star(\mathcal{R}_{K-1})} \leq \log^{1-\beta} n \tag{4.2}$$

To see this we examine the parameters of the LDF-readers in the sequence.

Parameters of the $\mathcal{R}_i$'s. Consider an LDF-reader $\mathcal{R}_i$ in the sequence, and assume $s_\star(\mathcal{R}_i) + 1 > 2^{\log^{1-\beta} n}$. The power-substitution extension-procedure is applied to its active domains using the parameter $b = (s_\star(\mathcal{R}_i) + 1)^{1/\log^{1-\beta} n}$, hence the parameter t used within the extension-procedure is $t = \log^{1-\beta} n$ (see Proposition 4.17). Since the active dimension of $\mathcal{R}_i$ is constant, Proposition 4.17 implies that

$$s_\star(\mathcal{R}_{i+1}) = s_\star(\mathcal{R}'_i) = d_\star(\mathcal{R}_i)t(b-1) = polylog(n) s_\star(\mathcal{R}_i)^{1/\log^{1-\beta} n} \tag{4.3}$$

As $\mathcal{R}_0$ is generated by the SP constructor, its active lower-degree parameter equals $s(F)$, so $s_\star(\mathcal{R}_0) = |\mathcal{F}|^{c_1} = 2^{\Theta(\log^\beta n)}$. By inductively using Equation 4.3 one easily sees that as long as $\beta - i(1-\beta) > 0$,

$$s_\star(\mathcal{R}_i) = 2^{\Theta(\log^{\beta - i(1-\beta)} n)} \tag{4.4}$$

(the poly-logarithmic factor is absorbed in the exponent).

Parameters of $\mathcal{R}_{i_o}$. Fix $i_o \doteq \lceil \beta/(1-\beta) \rceil$, and note that it is constant (it depends only on β, which remains constant throughout the proof). We have

$$1 - \beta \geq \beta - (i_o - 1)(1-\beta) > 0$$

hence by Equation 4.4,

$$s_\star(\mathcal{R}_{i_o-1}) = 2^{\Theta(\log^{\beta-(i_o-1)(1-\beta)} n)}$$

$\mathcal{R}'_{i_o-1}$ is generated by applying the extension-procedure with parameter

$$b = \max\left\{ \left(2^{\Theta(\log^{\beta-i_o(1-\beta)} n)}\right) ,\ 2 \right\} = O(1)$$

since $\beta - i_o(1-\beta) \leq 0$. The parameter t used is poly-logarithmic in n, specifically $t = O(\log^{\beta-(i_o-1)(1-\beta)} n) \leq O(\log^{1-\beta} n)$. It hence follows from Proposition 4.17 that $s_\star(\mathcal{R}_{i_o}) = s_\star(\mathcal{R}'_{i_o-1})$ is poly-logarithmic in n.

Parameters of $\mathcal{R}_{i_o+1}$. The power-substitution extension-procedure is applied with parameter $b = 2$ to generate $\mathcal{R}'_{i_o}$ from $\mathcal{R}_{i_o}$. Since $s_\star(\mathcal{R}_{i_o})$ is poly-logarithmic, t is poly-log-logarithmic in n, and therefore $s_\star(\mathcal{R}'_{i_o}) = s_\star(\mathcal{R}_{i_o+1})$ is also poly-log-logarithmic in n. Since $d_\star(\mathcal{R}_{i_o+1})$ is constant, it follows that

$$\binom{s_\star(\mathcal{R}_{i_o+1}) + d_\star(\mathcal{R}_{i_o+1})}{d_\star(\mathcal{R}_{i_o+1})} \leq \log^{1-\beta} n \tag{4.5}$$

Setting $K \doteq i_o + 2$, we have that $K = O(1/(1-\beta))$ is constant and that Inequality 4.2 clearly holds for $\mathcal{R}_{K-1} = \mathcal{R}_{i_o+1}$.

The lower − upper-degree gap remains.

Going from $\mathcal{R}_{K-1}$ to $\mathcal{R}'_{K-1}$, the CR constructor applies the linearization extension-procedure to each active domain of $\mathcal{R}_{K-1}$. This procedure is only applicable to domains where the lower-degree is not greater than the upper-degree, hence we must show that $s_\star(\mathcal{R}_{K-1}) \leq r_\star(\mathcal{R}_{K-1})$.

Let us compute how $s_\star(\mathcal{R}_{i+1})/r_\star(\mathcal{R}_{i+1})$ behaves with respect to $s_\star(\mathcal{R}_i)/r_\star(\mathcal{R}_i)$ for $0 \leq i < K-1$. Let b_i denote the b-parameter with which the power-substitution extension-procedure is applied to the active domains of $\mathcal{R}_i$ to obtain $\mathcal{R}'_i$, and let t_i denote the associated t-parameter. According to Proposition 4.17,

$$\frac{s_\star(\mathcal{R}_{i+1})}{r_\star(\mathcal{R}_{i+1})} = \frac{s_\star(\mathcal{R}'_i)}{r_\star(\mathcal{R}'_i)} \leq \frac{s_\star(\mathcal{R}_i) \cdot d_\star(\mathcal{R}_i) t_i (b_i - 1)}{r_\star(\mathcal{R}_i)} = O\left(\frac{s_\star(\mathcal{R}_i)}{r_\star(\mathcal{R}_i)} \cdot t_i b_i\right)$$

hence the ratio between the active lower-degree and the active upper-degree is consumed by a factor of up to $O(t_i b_i)$ in the transition from $\mathcal{R}_i$ to $\mathcal{R}_{i+1}$.

Let us bound $t_i b_i$. By the choice of the parameters b_i it follows that $t_i \leq \log^{1-\beta} n$ for all i. According to the above computations of $s_\star(\mathcal{R}_i)$, for $1 \leq i < K-3$

$$b_i = (s_\star(\mathcal{R}_i) + 1)^{1/\log^{1-\beta} n} = \left(2^{\Theta(\log^{\beta-i(1-\beta)} n)}\right)^{1/\log^{1-\beta} n} = 2^{\Theta(\log^{\beta-(i+1)(1-\beta)} n)} = 2^{O(\log^{\beta-(1-\beta)} n)}$$

and therefore $t_i b_i = 2^{O(\log^{\beta-(1-\beta)} n)}$. For $i = K-3$ or $i = K-2$, $b_i = O(1)$ so in these cases $t_i b_i$ is poly-logarithmic in n.

The initial lower-degree upper-degree fraction is $s_\star(\mathcal{R}_0)/r_\star(\mathcal{R}_0) = |\mathcal{F}|^{c_1-c_2} = 2^{-\Theta(\log^\beta n)}$. This fraction is consumed in each of the constant number of iterations by either $2^{O(\log^{\beta-(1-\beta)} n)}$ or a poly-logarithm, hence $s_\star(\mathcal{R}_{K-1})/r_\star(\mathcal{R}_{K-1}) = 2^{-\Theta(\log^\beta n)}$ and in particular $s_\star(\mathcal{R}_{K-1}) < r_\star(\mathcal{R}_{K-1})$, as desired.

The CR constructor takes polynomial time

We need to show that the CR constructor takes polynomial time in $|\mathcal{F}|^{d(F)}$. Since

$$|\mathcal{F}|^{d(F)} = \left(2^{\log^\beta n}\right)^{\Theta(\log^{1-\beta} n)} = n^{\Theta(1)}$$

this is equivalent to showing that it takes polynomial time in n. The proof is by showing that the generation of each LDF-reader $\mathcal{R}_i$ in the sequence $\mathcal{R}_0, \mathcal{R}'_0, \mathcal{R}_1, \mathcal{R}'_1, \ldots, \mathcal{R}_K$ takes polynomial time in n and in the size of the predecessor of $\mathcal{R}_i$ (clearly the time it takes to generate the final LDF-reader from $\mathcal{R}_K$ is polynomial in the size of $\mathcal{R}_K$). This implies that the CR constructor indeed takes polynomial time.

Generating $\mathcal{R}_0$. The CR constructor generates $\mathcal{R}_0$ using the SP constructor, which does take time polynomial in $|\mathcal{F}|^{d(F)}$.

Generating $\mathcal{R}_i$ for $0 < i \leq K$. The CR constructor generates $\mathcal{R}_i$ by applying the composition procedure to $\mathcal{R}'_{i-1}$. As mentioned in Section 4.4, this takes polynomial time in the size of $\mathcal{R}'_{i-1}$.

Generating $\mathcal{R}'_i$ for $i < K-1$. The CR constructor generates $\mathcal{R}'_i$ by applying the power-substitution extension-procedure to all active domains of $\mathcal{R}_i$. The time it takes is bounded by the size of $\mathcal{R}_i$ times the time needed for each application of the extension-procedure. By the definition of extension-procedures, each such application takes polynomial time in $|\mathcal{F}|^{d_\star(\mathcal{R}'_i)}$. According to Proposition 4.17, $d_\star(\mathcal{R}'_i) = d_\star(\mathcal{R}_i)t_i = O(t_i) \leq O(\log^{1-\beta} n)$ where t_i is as denoted in the degree-gap computation, hence $|\mathcal{F}|^{d_\star(\mathcal{R}'_i)} = n^{O(1)}$. Therefore each application of the extension procedure takes polynomial time in n, as needed.

Generating $\mathcal{R}'_{K-1}$. The difference between the generation of $\mathcal{R}_K$ and that of the other $\mathcal{R}'_i$'s, is that the linearization extension-procedures is applied to each active domain instead of the power-substitution extension-procedure. Each such application still takes polynomial time in $|\mathcal{F}|^{d_\star(\mathcal{R}_{K-1})}$ but here $d_\star(\mathcal{R}_{K-1})$ is calculated according to Proposition 4.18,

$$d_\star(\mathcal{R}_{K-1}) = \binom{s_\star(\mathcal{R}_{K-1}) + d_\star(\mathcal{R}_{K-1})}{d_\star(\mathcal{R}_{K-1})} \leq \log^{1-\beta} n$$

$|\mathcal{F}|^{d_\star(\mathcal{R}_{K-1})}$ is therefore still polynomial.

(ρ, ϵ)-parameters of the CR constructor.

We now show that $\mathcal{R}$ has parameters $(\rho, \rho^{4^{-K}/3})$ for all ρ's that satisfy $\rho > (r/|\mathcal{F}|)^{c_g} d$. We first prove by induction that for all i, $\mathcal{R}_i$ is a restricted $(\rho, \rho^{4^{-i}/3})$-LDF-reader: For $\mathcal{R}_0$ it follows directly from Lemma 4.2. Assume now that $\mathcal{R}_{i-1}$ is a restricted $(\rho, \rho^{4^{-i+1}/3})$-LDF-reader. The extension proposition (Proposition 4.16) implies that $\mathcal{R}'_{i-1}$ has the same parameters. Since $\mathcal{R}_i$ is generated from $\mathcal{R}'_{i-1}$ by the composition procedure, Proposition 4.19 yields that $\mathcal{R}_i$ is a restricted $(\rho, \rho^{4^{-i}/3})$-LDF-reader, as desired. Note that the requirement over ϵ in Proposition 4.19 holds here, since we apply it with $\epsilon^{3/4} = \rho^{4^{-i}} \geq \rho > (r/|\mathcal{F}|)^{c_g} d$.

By the above induction, $\mathcal{R}_K$ is a restricted (ρ, ρ^c)-LDF-reader for $c \doteq 4^{-K}/3$. To show that $\mathcal{R}$ has the same parameters, we define for each assignment $\mathcal{A}$ of $\mathcal{R}$ a *feasible* assignment $\mathcal{A}'$ for $\mathcal{R}_K$, such that an $\mathcal{R}_K$-local-reader is ρ-erroneous with respect to $\mathcal{A}'$ if and only if the $\mathcal{R}$-local-reader generated from it is ρ-erroneous with respect to $\mathcal{A}$. This would imply that $\mathcal{R}$ has the same (ρ, ϵ) parameters as $\mathcal{R}_K$.

$\mathcal{A}'$ differs from $\mathcal{A}$ only on active domains of $\mathcal{R}_K$ – for an active domain G and a variable $G[(u_1, \ldots, u_d)]$ in it we define

$$\mathcal{A}'(\, G[(u_1, \ldots, u_d)]\,) \doteq \sum_{i=1}^{d} \mathcal{A}(G_i) u_i$$

where the G_i's are the new variables added in the generation of the final CR constructor. $\mathcal{A}'$ assigns to each active domain G a linear LDF represented by the assignment of the G_i's, and is hence feasible. It is clear from the construction of $\mathcal{R}$ that a local-reader of $\mathcal{R}_K$ is ρ-erroneous with respect to $\mathcal{A}'$ if and only if the $\mathcal{R}$-local-reader obtained from it is ρ-erroneous with respect to $\mathcal{A}$.

Chapter 5

The Sum-Check

In this chapter we prove the Sum-Check Lemma, Lemma 3.1. A reduction algorithm is shown that given a system Ψ of n quadratic-equations, where there are up to n variables in each equation, generates a system Ψ_{sc} whose variables belong to domains, and where every equation accesses only a constant number of variables. The reduction of Ψ into Ψ_{sc} is gap-preserving in the sense that if Ψ is completely satisfiable then Ψ_{sc} can be completely satisfiable by a **good** assignment to its domains; and if there is no assignment for Ψ that satisfies more than a $\frac{2}{|\mathcal{F}|}$ fraction of its equations, then no **feasible** assignment for Ψ_{sc} can satisfy more than a $\frac{2}{|\mathcal{F}|}$ fraction of its equations as well.

The reduction begins with the given system $\Psi_0 \doteq \Psi$, and puts it through a constant number $(O(\frac{1}{1-\beta}))$ of transformations, obtaining a sequence $\Psi_0, \Psi_1, \ldots, \Psi_l$ of equation-systems. The final system Ψ_{sc} is generated by a small alteration of Ψ_l. The intermediate systems Ψ_i share a similar structure and are hence called *restricted equation-systems*, as defined below. However, the number of variables in their equations decreases from up to n in Ψ_0, to a constant in Ψ_l.

A system Ψ_i in the sequence is transformed into Ψ_{i+1} by substituting each equation ψ of Ψ_i with several new equations. The new equations represent ψ in the following sense: if ψ is satisfied by a certain good assignment then the new equations are also satisfied by an extension thereof, and if ψ is not satisfied by a certain feasible assignment then the new equations can not not be satisfied, except for at most an $|\mathcal{F}|^{-1/2}$ fraction. As shown below, this property suffices for the gap to be mostly preserved in the transformation of Ψ_i into Ψ_{i+1}.

We continue as follows. The next subsection defines the structure of a restricted equation-system. The transformation of each restricted equation-system Ψ_i into Ψ_{i+1} (the transformation of Ψ_0 into Ψ_1 is an exception) is performed by an algorithm that is described in Section 5.1. This algorithm is used for the transformation of each intermediate system into the next, but it uses a different *representation-procedure* each time. Section 5.2 describes the properties of the different representation-procedures used, and of the algorithm which transforms Ψ_0 into Ψ_1. The complete reduction of Ψ into Ψ_{sc} is finally described in Section 5.3. The next sections are dedicated to proving the correctness of the reduction (Section 5.4), and to the description and correctness proofs of the product-check and the representation-procedures it uses (Sections 5.5, 5.6, 5.7, 5.8, and 5.9).

Restricted Equation-Systems

All the systems Ψ_i, $i = 1, 2, \ldots, l$, in the sequence generated by the reduction algorithm have a similar structure. The following is an exact definition thereof.

Definition 5.1 (restricted equation-systems). *A* restricted equation-system Ψ*, is a quadratic equation-system where some domains are considered* active. *The dimension, and the upper and lower-degree parameters of the active domains must all be the same. These parameters are called the active dimension, active upper-degree and active lower-degree of* Ψ*, and are denoted by* $d_\star(\Psi)$*,* $r_\star(\Psi)$ *and* $s_\star(\Psi)$ *respectively.*

Each equation $\psi \in \Psi$ *is written in the form "*$\psi_\star = \psi_{\mathbf{c}}$*".* $\psi_\star$ *is called the active part of* ψ *and* $\psi_{\mathbf{c}}$ *is called the core of* ψ*. While* $\psi_{\mathbf{c}}$ *may contain any variable of* Ψ *and can have quadratic as well as linear terms,* $\psi_\star$ *contains only linear terms and the variables in it must all be from one active domain, called the active domain of* ψ *and denoted by* $\mathrm{Dom}_\star(\psi)$*. The variables that appear in* $\psi_\star$ *are called the active variables of* ψ*.*

The equations in all intermediate equation-systems will have only a constant number of variables in their core – all other variables appear in the active part of the equations. It is hence useful to denote the number of variables in the core of an equation and the number of active variables by different names.

Definition 5.2 (active and core-dependency). *Let* Ψ *be a restricted equation-system. The* active-dependency *of an equation* $\psi \in \Psi$*, denoted by* $\mathrm{D}_\star(\psi)$*, is defined as the number of variables in* $\psi_\star$*. The* core-dependency *of* ψ*,* $\mathrm{D}_{\mathbf{c}}(\psi)$*, is defined as the number of variables in* $\psi_{\mathbf{c}}$*. The active-dependency of* Ψ*, denoted by* $\mathrm{D}_\star(\Psi)$*, is the maximum of* $\mathrm{D}_\star(\psi)$ *over all equations* $\psi \in \Psi$*. The core-dependency of* Ψ *is denoted* $\mathrm{D}_{\mathbf{c}}(\Psi)$*, and is defined similarly.*

As mentioned above, the core-dependency parameters of all the restricted equation-systems in the sequence $\Psi_1, \ldots, \Psi_l$ are constants. The active-dependency parameter is decreased gradually until it becomes constant in Ψ_l. The total number of variables in an equation of Ψ_l is therefore constant, as required by Lemma 3.1 (this property is preserved in the final transition from Ψ_l to Ψ_{sc}).

5.1 The Main Transformation-Scheme

The transformation of each restricted equation-system Ψ_i into the next is done by substituting each equation ψ of Ψ_i by a "representation" containing several new equations.

The transformations of Ψ_i into Ψ_{i+1} where $i = 1, \ldots, l-1$ are in fact of a more specific structure, and are carried out by the *system-representation algorithm*. An important part of this algorithm is the application of a *representation-procedure* to each equation in the system (a different representation-procedure is used for different transformations). We now define a representation-procedure, and then describe the system-representation algorithm.

Representation-procedures

A representation-procedure is an algorithm that is applied to an equation ψ and produces a set $\mathcal{E}_\psi$ of *conjunctions* of equations, and a new domain denoted by $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ (the new conjunctions may have variables from $\mathrm{Dom}_\star(\mathcal{E}_\psi)$). The conjunctions of $\mathcal{E}_\psi$ represent ψ in the sense that they are only satisfied by an extension of assignments that also satisfy ψ – otherwise almost none of them can be satisfied.

For $i = 1, \ldots, l$, Ψ_{i+1} is obtained from Ψ_i by applying a representation-procedure to each equation $\psi \in \Psi_i$, generating a system Ψ'_i of conjunctions which is the union of the sets $\{\mathcal{E}_\psi\}_{\psi \in \Psi_i}$. Ψ_{i+1} is then generated by replacing the conjunctions with equations as in Proposition 3.7. If the representation-procedure generates conjunctions with a small number of variables, then the dependency parameter of Ψ_{i+1} will be smaller than that of Ψ_i (eventually the dependency is constant). Also, the active-domains of Ψ_{i+1} are set to be the new domains generated by the representation-procedure, and hence the active parameters of Ψ_{i+1} are changed. Actually the reduction generates domains with different parameters, contrary to a requirement in Lemma 3.1. This is rectified by applying a simple technical method at the end of the reduction, that makes all the domains uniform.

An $[s,d]$-representation-procedure. An $[s,d]$-representation-procedure is an algorithm $\mathbf{A}$ that receives as input an equation ψ in a restricted equation-system Ψ, and generates a set $\mathcal{E}_\psi$ of conjunctions of quadratic-equations that "represent" ψ. It also generates a new domain denoted $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ – the conjunctions in $\mathcal{E}_\psi$ may have variables from $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ in addition to any variables of Ψ. For a conjunction $\chi \in \mathcal{E}_\psi$ we define the active domain of χ to be $\mathrm{Dom}_\star(\chi) \doteq \mathrm{Dom}_\star(\mathcal{E}_\psi)$. The variables of χ that are associated with $\mathrm{Dom}_\star(\chi)$ are called the active variables of χ.

The parameters $r(\mathrm{Dom}_\star(\psi))$, s and d determine the parameters of the new domain, namely $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ must satisfy $r(\mathrm{Dom}_\star(\mathcal{E}_\psi)) = r(\mathrm{Dom}_\star(\psi))$, $s(\mathrm{Dom}_\star(\mathcal{E}_\psi)) = s$, and $d(\mathrm{Dom}_\star(\mathcal{E}_\psi)) = d$. The running time of $\mathbf{A}$ should be polynomial in $|\mathcal{F}|^d = |\mathrm{Dom}_\star(\mathcal{E}_\psi)|$ and the size of ψ.

Extension and restriction properties. For the conjunctions in $\mathcal{E}_\psi$ to properly represent ψ, it is required that $\mathbf{A}$ has the following extension and restriction properties:

- Extension Property: For every good assignment $\mathcal{A}$ for Ψ that satisfies ψ there should be an s-degree LDF such that if it is assigned to $\mathrm{Dom}_\star(\mathcal{E}_\psi)$, all the conjunctions in

$\mathcal{E}_\psi$ are satisfied.

- Restriction Property: If a feasible assignment for Ψ and for $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ satisfies at least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions in $\mathcal{E}_\psi$, then ψ is satisfied as well.

Uniformity. It is required that the parameters s and d be functions of Ψ alone, so that the parameters of $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ are the same for all equations $\psi \in \Psi$ to which **A** is applied. The number of conjunctions in $\mathcal{E}_\psi$ should also be independent of ψ (and be a function of Ψ alone). The number of equations in each conjunction of $\mathcal{E}_\psi$ should all be the same, and they must be independent of ψ as well. In addition we require that the number of equations in each conjunction is bounded by $O(d)$.

Conjunction-structure. The conjunctions of $\mathcal{E}_\psi$ should have the following structure. $\psi_{\mathbf{c}}$ may appear at most once in at most one equation of each conjunction $\chi \in \mathcal{E}_\psi$. Except for the terms in this copy of $\psi_{\mathbf{c}}$, all terms must be linear and the number of terms not associated with the domain $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ must be bounded by a constant (that is, a number which is independent of ψ and Ψ).

The system-representation algorithm.

Let us now describe how a restricted equation-system Ψ_i is transformed into Ψ_{i+1} using a representation-procedure **A**.

1. First, **A** is applied to every equation $\psi \in \Psi_i$.

2. A system Ψ_i' of conjunctions is constructed by taking the union of the sets $\{\mathcal{E}_\psi\}_{\psi\in\Psi_i}$. Note that the number of equations in each conjunction of Ψ_i' is the same, and that each equation of Ψ_i results in the same number of conjunctions in Ψ_i'.

3. Ψ_{i+1} is generated by replacing each conjunction $\chi \in \Psi_i'$ by all linear-combinations of its equations (with multiplicities, if the same equation occurs more than once). The active domain of each equation is set to be the same as that of the conjunction from which it originated. The active variables of such an equation are thus the same as the active variables of the originating conjunction.

4. For each equation $\xi \in \Psi_{i+1}$, $\xi_\star$ and $\xi_{\mathbf{c}}$ are defined as follows. The variables associated with $\mathrm{Dom}_\star(\xi)$ are moved to the left-hand side of the equation, and the other variables to the right-hand side (it follows from the properties of **A** that variables associated with $\mathrm{Dom}_\star(\xi)$ only appear in linear terms). The left-hand side of ξ is then defined to be the active part $\xi_\star$ of ξ, and the right-hand side is set to be its core, $\xi_{\mathbf{c}}$.

Let us examine some of the properties of the system-representation algorithm.

The parameters of Ψ_{i+1}. Since the upper-degree parameter of the domains produced by the representation-procedure $\mathbf{A}$ are the same as the active upper-degree of Ψ_i, we have $r_\star(\Psi_{i+1}) = r_\star(\Psi_i)$. If $\mathbf{A}$ is an $[s, d]$-representation-procedure, then the active domains of Ψ_{i+1} will all have lower-degree s and dimension d, hence $s_\star(\Psi_{i+1}) = s$ and $d_\star(\Psi_{i+1}) = d$.

Time. The system-representation algorithm takes polynomial time in the size of Ψ_i and $|\mathcal{F}|^{d_\star(\Psi_{i+1})}$. Especially note that step 3 is applicable in polynomial time in $|\mathcal{F}|^{d_\star(\Psi_{i+1})}$ and in the size of Ψ_i – the uniformity property requires that the number of equations in each conjunction be bounded by $O(d_\star(\Psi_{i+1}))$, hence the number of equations produced for each conjunction is $|\mathcal{F}|^{O(d_\star(\Psi_{i+1}))}$.

Core-dependency. Note that the core-dependency of Ψ_{i+1} is larger by at most a constant than that of Ψ_i. Consider an equation $\psi' \in \Psi_{i+1}$ whose origin is an equation $\psi \in \Psi_i$. It has the variables of $\psi_{\mathbf{c}}$, and at most a constant number of variables not associated with $\mathrm{Dom}_\star(\mathcal{E}_\psi)$. The other variables are associated with $\mathrm{Dom}_\star(\mathcal{E}_\psi)$, and are hence active, so the core-dependency of ψ' is larger by only a constant than that of ψ.

The gap. The extension and restriction properties of the representation-procedure $\mathbf{A}$ that is used by the system-representation algorithm, ensure that the fraction of satisfiable equations in Ψ_i with respect to good or feasible assignments is close to the satisfiable fraction in Ψ_{i+1}. Here is a precise definition of this property.

Definition 5.3 (gap-preserving algorithm). *An algorithm that transforms a given equation-system Ψ_i into an equation-system Ψ_{i+1} is said to be gap-preserving if it has the following properties:*

- *Completeness: If Ψ_i can be completely satisfied by a good assignment then so can Ψ_{i+1}.*
- *Soundness: If a feasible assignment for Ψ_{i+1} can satisfy a γ-fraction of its equations, then there exists a feasible assignment for Ψ_i satisfying at least a $\gamma - O(|\mathcal{F}|^{-1/2})$ fraction of its equations.*

Note that even in a gap-preserving algorithm, the gap is actually consumed by an $O(|\mathcal{F}|^{-1/2})$ fraction. The reduction deals with this by applying a simple gap amplification technique in the transformation from the system Ψ_l into the final system Ψ_{sc}.

Proposition 5.4. *The system-representation algorithm is gap-preserving.*

Proof. Assume that the system-representation algorithm is applied to a restricted equation-system Ψ_i using a representation-procedure $\mathbf{A}$, and outputs Ψ_{i+1}.

The completeness property is implied from the extension property of $\mathbf{A}$ as follows. Suppose Ψ_i is satisfied by an assignment $\mathcal{A}$. According to the extension property, $\mathcal{A}$ can be extended to assign for each ψ an $s(\mathrm{Dom}_\star(\mathcal{E}_\psi))$ degree LDF to $\mathrm{Dom}_\star(\mathcal{E}_\psi)$ such that the conjunctions in $\mathcal{E}_\psi$ are satisfied. The system of conjunctions Ψ_i', generated by the system-representation algorithm, is hence satisfied by this extended assignment, and therefore Ψ_{i+1} is satisfied as well by Proposition 3.7.

Let us now prove the soundness property. Assume that Ψ_{i+1} is γ-satisfiable by a feasible assignment $\mathcal{A}$. Then by Proposition 3.7, Ψ_i' is at least $\gamma - |\mathcal{F}|^{-1}$ satisfied by $\mathcal{A}$. Recall that the number of conjunctions in the set $\mathcal{E}_\psi$ is the same for every $\psi \in \Psi_i$. Hence for at least a $\gamma - 2|\mathcal{F}|^{-1/2}$ fraction of the sets $\mathcal{E}_\psi$, a $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions are satisfied: Otherwise the fraction of satisfied conjunctions in Ψ_{i+1} would be less than $\gamma - 2|\mathcal{F}|^{-1/2} + (1 - \gamma + 2|\mathcal{F}|^{-1/2})|\mathcal{F}|^{-1/2} < \gamma - |\mathcal{F}|^{-1}$.

By the restriction property of $\mathbf{A}$, it follows that at least a $\gamma - 2|\mathcal{F}|^{-1/2}$ fraction of the equations in Ψ_i are satisfied by $\mathcal{A}$. The proof is thus completed, noting that the restriction of $\mathcal{A}$ to the variables of Ψ_i is feasible. ■

5.2 The Representation-Procedures

We now state the properties of the representation-procedures that are utilized in reducing Ψ to Ψ_{sc}. Only the properties that are needed for the reduction are discussed – the actual representation-procedures and the proofs of their stated properties appear later.

Product-check.

The product-check algorithm is actually not a representation-procedure. Its properties are stated here since it is used to transform Ψ_0 into Ψ_1.

Lemma 5.5 (product-check). *Let Ψ be a system of n quadratic equations over a field $\mathcal{F}$, $|\mathcal{F}| = 2^{\log^\beta n}$, where there are at most n variables in each equation. There exists a gap-preserving polynomial time algorithm that given such a system, constructs a restricted equation-system Ψ_* that has the following properties:*

- $\mathrm{D}_{\mathbf{c}}(\Psi_*)$ *is bounded by a constant.*
- Ψ_* *has exactly one domain F which is the active domain of all of its equations. The parameters of F, that also determine the active degree and dimension parameters of Ψ_*, are $r(F) = |\mathcal{F}|^{1/4}$, $s(F) = |\mathcal{F}|^{1/8}$, and $d(F) = \Theta(\log^{1-\beta} n)$.*

Interpolation.

Applying the system-representation algorithm to a system using the interpolation representation-procedure generates a system of *condensed* equations – an equation is condensed if its active variables are associated with points from the *principal cube* of its active domain, defined shortly below. The reduction uses the interpolation representation-procedure to transform Ψ_1 into Ψ_2 since the arithmetization representation-procedure, applied to Ψ_2, can only be applied to condensed equations.

Next we formally define condensed equations in order to state the properties of the interpolation representation-procedure. A more detailed explanation of the use of condensed equations appears in Section 5.7, which describes the arithmetization representation-procedure.

The principal cube of a domain. To define the principal cube we assume that for each non-negative number $s < |\mathcal{F}|$ an arbitrary subset $\mathcal{H}_s$ of $\mathcal{F}$ is fixed, of size $s+1$. The principal cube of a domain F is defined to be the subset $\left(\mathcal{H}_{s(F)}\right)^{d(F)} \subseteq \mathcal{F}^{d(F)}$.

Definition 5.6 (condensed equations). *Let ψ be an equation or a conjunction with an active domain F, in a restricted equation-system. ψ is said to be condensed if all of its active variables are associated with points in the principal cube of F. A restricted equation-system where all of its equations are condensed is called condensed.*

Note that for a domain F, the value of an $[s(F), d(F)]$-LDF at any point can be interpolated by a linear combination of its values on the principal cube of F, where the coefficients are independent of the LDF. Utilizing this, when applied to an equation ψ the interpolation representation-procedure generates a representation $\mathcal{E}_\psi$ containing only condensed conjunctions. It follows that when the system-representation algorithm is applied to a restricted equation-system with the interpolation as a procedure, the generated equation-system has condensed equations.

Lemma 5.7 (interpolation). *Let Ψ be a restricted equation-system satisfying $r_\star(\Psi) < |\mathcal{F}|^{1/2}$.*

There exists an $[s_\star(\Psi), d_\star(\Psi)]$-representation-procedure applicable to the equations of such a system called **interpolation***, that generates only condensed conjunctions.*

Arithmetization.

The arithmetization representation-procedure uses a technique from [BFL91] to generate systems with a reduced active-dependency parameter. Given a condensed equation ψ, it produces a representation $\mathcal{E}_\psi$ where the number of active variables in each conjunction is a function of the degree and dimension parameters of $\mathrm{Dom}_\star(\psi)$. If these parameters are small enough, then the active-dependency is decreased. Note that the degree and dimension parameters themselves are not decreased, hence an iterative application of the arithmetization representation-procedure would not further reduce the dependency.

Lemma 5.8 (arithmetization). *Let Ψ be a restricted equation-system satisfying $s_\star(\Psi)d_\star(\Psi) + r_\star(\Psi) < |\mathcal{F}|^{1/2}$, and where all the equations are condensed.*

There exists a $[2d_\star(\Psi)s_\star(\Psi)\,,\; d_\star(\Psi)+1]$*-representation-procedure applicable to the equations of such systems called* **arithmetization***, that generates conjunctions with at most* $2d_\star(\Psi)\cdot s_\star(\Psi)$ *active variables.*

Curve-extension.

When applied to equations with a small active-dependency parameter, the curve-extension representation-procedure generates domains with small degree and dimension parameters. The active dependency is not reduced (in fact it increases somewhat), but then the system-representation algorithm is applied to the resulting system using the arithmetization representation-procedure, and the decrease in the degree and dimension parameters is utilized to reduce the active dependency as well. By applying the system-representation algorithm using the curve-extension and the arithmetization representation-procedures alternately, the reduction gradually reduces the active-dependency, the active degree and the dimension parameters of the intermediate systems.

Lemma 5.9 (curve-extension). *Let* Ψ *be a restricted equation-system such that* $s_\star(\Psi)\mathrm{D}_\star(\Psi)$ *and* $r_\star(\Psi)\mathrm{D}_\star(\Psi)$ *are smaller than* $|\mathcal{F}|^{1/2}$*. There exists an* $[s,d]$*-representation-procedure called* **curve-extension** *applicable to the equations of such systems, for*

$$d \doteq \min\left\{d_\star(\Psi), \log_2\left(s_\star(\Psi)\cdot \mathrm{D}_\star(\Psi)\right)\right\}$$

$$\textit{and} \quad s \doteq d\cdot\max\left\{\left(s_\star(\Psi)\cdot\mathrm{D}_\star(\Psi)\right)^{\frac{1}{d_\star(\Psi)}}, 2\right\}$$

that generates only condensed conjunctions.

Linearization.

Applying the system-representation algorithm using the linearization representation-procedure obtains a system with constant active-dependency, as desired. However it is applicable in polynomial time only to systems where the active degree and dimension parameters are very small (the running time of a representation-procedure is polynomial in the size of the newly generated domains, which may become very large in the case of linearization). Hence the reduction generates a sequence of intermediate equation-systems where the active parameters are gradually reduced, until they finally become suitable for the linearization representation-procedure to be applied.

Lemma 5.10 (linearization). *Let* Ψ *be a system such that* $r_\star(\Psi)\mathrm{D}_\star(\Psi)$ *and* $s_\star(\Psi)\mathrm{D}_\star(\Psi)$ *are smaller than* $(|\mathcal{F}|^{1/2})/2$*.*

There exists a $[1, s_\star(\Psi)\mathrm{D}_\star(\Psi)]$*-representation-procedure applicable to such systems called* **linearization***, that generates conjunctions with at most* 4 *active variables.*

Note that as mentioned above, for the linearization representation-procedure to be applicable within the reduction, $s_\star(\Psi)\mathrm{D}_\star(\Psi)$ should be in fact considerably smaller than the above bound of $|\mathcal{F}|^{1/2}$.

5.3 The Reduction Algorithm of Ψ Into Ψ_{sc}

We now state the reduction algorithm that transforms Ψ into Ψ_{sc}, as claimed by Lemma 3.1. This algorithm is mostly a concatenation of the algorithms that were discussed above. Starting with $\Psi = \Psi_0$, the reduction algorithm applies the product-check algorithm to obtain Ψ_1, and from there it continues to use the system-representation algorithm, applying it a constant ($O(\frac{1}{1-\beta})$) number of times with different representation-procedures. This yields a sequence of equation-systems $\Psi_2, \ldots, \Psi_l$. Ψ_{sc} is then obtained from Ψ_l by a simple transformation.

We next give the sequence of transformations and representation-procedures used to obtain Ψ_l from Ψ_0, and then describe how Ψ_{sc} is obtained from Ψ_l. In Section 5.4 it is shown that this reduction takes polynomial time in n, and that the generated system Ψ_{sc} has the desired properties. Section 5.4 also shows that although each representation-procedures is applicable only to systems with certain parameters, the reduction algorithm does use them correctly.

The sequence of systems.

First, the reduction applies the product-check algorithm to Ψ_0 and obtains Ψ_1. The system-representation algorithm is then applied to Ψ_1 with the interpolation representation-procedure to obtain Ψ_2. The next $\frac{2\beta}{1-\beta}$ systems*, $\Psi_3, \ldots, \Psi_{\frac{2\beta}{1-\beta}+2}$ are generated, by applying the system-representation algorithm with the arithmetization and the curve-extension representation-procedures alternately (the arithmetization is used first). The system-representation algorithm is then applied once more to $\Psi_{\frac{2\beta}{1-\beta}+2}$ using the arithmetization representation-procedure, and then finally it is applied once again using the linearization representation-procedure. The outcome is Ψ_l, where $l \doteq \frac{2\beta}{1-\beta} + 4$. Apart from a simple transformation that is described shortly below, Ψ_l is the outcome of the reduction.

Properties of Ψ_l.

Before we describe how Ψ_l is transformed into Ψ_{sc}, let us overview its main properties.

Constant dependency. Ψ_l has the desired dependency parameter, namely a constant. Since it is generated using the linearization representation-procedure, it follows from Lemma 5.10 that its active-dependency parameter is constant. As for the core-dependency, Ψ_1 is generated using the product-check algorithm and therefore by Lemma 5.5 its core-dependency is constant. Since the other systems in the sequence $\Psi_2, \ldots, \Psi_l$, are generated using the system-representation algorithm, the core-dependency increases only by a constant throughout the sequence (recall that the sequence is of constant length).

*For simplicity of exposition, we assume here that $\beta/(1-\beta)$ is an integer.

Completeness, and soundness. Since each of the intermediate transformations that were applied so far are gap-preserving, it follows immediately that the transformation from $\Psi = \Psi_0$ into Ψ_l is gap preserving as well. Hence Ψ_l has the following properties:

- Completeness: If Ψ can be completely satisfied by a good assignment then so can Ψ_l.
- Weakened Soundness: If Ψ is no more than $\frac{2}{|\mathcal{F}|}$-satisfiable then Ψ_l cannot be more than $O(|\mathcal{F}|^{-1/2})$-satisfied by a feasible assignment.

From Ψ_l to Ψ_{sc}.

Ψ_l fails to comply with two requirements of Lemma 3.1: The parameters of its domains are not all the same, and it has only a weakened soundness property, which is less than what is required in Lemma 3.1. The reduction hence transforms Ψ_l into Ψ_{sc} in two steps. First it resets the degree and dimension parameters of its domains without changing any of the other properties, and then it applies a simple technique to amplify the soundness property.

Parameter uniformization. First note that the upper-degree parameter is the same for all the domains of Ψ_l since Ψ_1 has only one domain, and the representation-procedures generate domains with the same upper-degree as the active domain of the equation to which they are applied. Denote this upper-degree by $r(\Psi_{sc})$, and fix $s(\Psi_{sc})$ to be the maximum over all lower-degrees of domains in Ψ_l, and $d(\Psi_{sc})$ to be the maximum over all the dimension parameters. As shown in Section 5.4, $s(\Psi_{sc})$ is smaller than $r(\Psi_{sc})$.

The reduction replaces each domain F of Ψ_l by a new domain F' with $r(F') = r(\Psi_{sc})$, $s(F') = s(\Psi_{sc})$ and $d(F') = d(\Psi_{sc})$. Each variable $F[x]$ which appears in an equation of Ψ_l is then replaced by the variable $F'[x']$, where x' is obtained from x by padding it with the appropriate number of zeros (in case the dimension parameter of F' is larger than that of F).

Note that the completeness and weakened soundness properties of Ψ_l are not affected by the uniformization step. Resetting the lower-degree parameter maintains the completeness property since the lower-degree parameters may only be increased, and it has no effect on the soundness. The dimension enlargement also preserves the completeness property, as an LDF that was assigned to a domain before the change of dimension extends naturally to the larger domain maintaining the same degree, and thus a satisfying assignment can be translated through the uniformization step. Similarly, a feasible assignment to a domain with an enlarged dimension translates to a feasible assignment to the original domain by restriction, thus preserving the values of the variables appearing in the equations, and therefore the weakened soundness property is also maintained.

Soundness amplification. To amplify the soundness of Ψ_l the reduction first generates all conjunctions of three (not necessarily distinct) equations from Ψ_l. It then replaces each such conjunction with the set of all linear-combinations over its equations. The set of equations of Ψ_{sc} is thus

$$\{ \sum_{i=1}^{3} \lambda_i \psi_i \ : \quad \forall\, i \quad \lambda_i \in \mathcal{F}\,,\ \psi_i \in \Psi_l \}$$

Completeness and soundness for Ψ_{sc}. Since it is simple to observe that the completeness property is maintained by the soundness amplification step, let us verify that Ψ_{sc} has the soundness property. Assume then that Ψ is no more than $2/|\mathcal{F}|$-satisfiable. As mentioned above, a feasible assignment for Ψ_l cannot satisfy more than an $O(|\mathcal{F}|^{-1/2}) < |\mathcal{F}|^{-1/3}$ fraction of its equations, and this remains true when the domain-parameters of Ψ_l are reset. The fraction of conjunctions of three equations that can be satisfied by a feasible assignment is hence less than $\left(|\mathcal{F}|^{-1/3}\right)^3 = 1/|\mathcal{F}|$. It then follows from Proposition 3.7 that Ψ_{sc} cannot be more than $2/|\mathcal{F}|$-satisfiable by a feasible assignment (Proposition 3.7 discusses general assignments but it is easily extendable to feasible assignments).

5.4 The Reduction Works

Based on the stated properties of the representation-procedures, we now verify that the reduction algorithm described above is applicable, and that the generated system Ψ_{sc} has the required parameters. The completeness and soundness properties of Ψ_{sc} have already been verified. From the properties of Ψ_l and the construction of Ψ_{sc} it is obvious that the number of variables in the equations of Ψ_{sc} is bounded by a constant and that the parameters of its domains are all the same.

We now compute the active parameters of all the intermediate systems $\Psi_1, \ldots, \Psi_l$, and at the same time verify that all representation-procedures are correctly used by the reduction. The computation will also imply that the parameters of the domains of Ψ_{sc} are as required by Lemma 3.1, and that the reduction takes polynomial time. For simplicity, we use O and Θ notations in the computation, where any function that depends solely on β is regarded as constant.

The active parameters of the intermediate systems

As mentioned above the domains of Ψ_{sc}, as well as the domains in all the intermediate systems, all have the same upper-degree parameter, namely $r(\Psi_{sc})$. It also equals the active upper-degree of Ψ_1, hence $r(\Psi_{sc}) = |\mathcal{F}|^{1/4}$. Let us consider the other parameters of the intermediate systems.

The parameters of Ψ_1 and Ψ_2. Ψ_1 is generated from Ψ_0 using the product-check algorithm (see Lemma 5.5), hence it has the parameters

- $s_\star(\Psi_1) = |\mathcal{F}|^{1/8}$
- $d_\star(\Psi_1) = \Theta(\log^{1-\beta} n)$

Ψ_2, generated from Ψ_1 by applying the system-representation algorithm with the interpolation representation-procedure, has the same parameters. Note that the interpolation representation-procedure is indeed applicable to the equations of Ψ_1 under these parameters.

The parameters of Ψ_3. Ψ_3 is obtained from Ψ_2 using the arithmetization representation-procedure. Note that the parameters of Ψ_2 are such that the arithmetization representation-procedure is applicable. The parameters of Ψ_3, as follows from the arithmetization lemma, are

- $s_\star(\Psi_3) = 2d_\star(\Psi_2)s_\star(\Psi_2) = \Theta(|\mathcal{F}|^{1/8}\log^{1-\beta} n) = 2^{\Theta(\log^\beta n)}$
- $\mathrm{D}_\star(\Psi_3) = 2d_\star(\Psi_2)s_\star(\Psi_2) = \Theta(|\mathcal{F}|^{1/8}\log^{1-\beta} n) = 2^{\Theta(\log^\beta n)}$
- $d_\star(\Psi_3) = d_\star(\Psi_2) + 1 = \Theta(\log^{1-\beta} n)$

The active parameters of $\Psi_4, \Psi_5, \ldots, \Psi_{l-5}$ (recall that $l - 5 = \frac{2\beta}{1-\beta} - 1$) are given by the following proposition.

Proposition 5.11. *For i such that $4 \leq 2i \leq l-6$, the active parameters of Ψ_{2i} (generated using the curve-extension representation-procedure) are*

- $s_\star(\Psi_{2i}) = 2^{\Theta(\log^{\beta - i(1-\beta)} n)}$
- $d_\star(\Psi_{2i}) = \Theta(\log^{1-\beta} n)$

and for i such that $5 \leq 2i+1 \leq l-5$, the parameters of Ψ_{2i+1} (that is generated using the arithmetization representation-procedure) are

- $s_\star(\Psi_{2i+1}) = 2^{\Theta(\log^{\beta - i(1-\beta)} n)}$
- $\mathrm{D}_\star(\Psi_{2i+1}) = 2^{\Theta(\log^{\beta - i(1-\beta)} n)}$
- $d_\star(\Psi_{2i+1}) = \Theta(\log^{1-\beta} n)$

Proof. The proposition is obtained by induction over i, calculating the parameters of an equation-system according to the parameters of the previous system and the properties of the appropriate representation-procedure. We omit the calculation. ∎

Note that the systems Ψ_{2i+1} have parameters such that the curve-extension representation-procedure is applicable, and that the arithmetization representation-procedure is applicable for the Ψ_{2i} systems, hence the sequence of transformation is valid up to and including Ψ_{l-5}. From the computations below it is also implied that the representation-procedure used for generating $\Psi_{l-4}, \ldots, \Psi_l$ are also applicable.

Parameters of Ψ_{l-4}. Setting $2i+1 = l-5 = 2(\frac{\beta}{1-\beta}) - 1$ in the above proposition we obtain that $s_\star(\Psi_{l-5}) = \mathrm{D}_\star(\Psi_{l-5}) = 2^{\Theta(\log^{1-\beta} n)}$, and that $d_\star = \Theta(\log^{1-\beta} n)$. Hence according to the Curve-Extension Lemma (Lemma 5.9),

- $s_\star(\Psi_{l-4}) = \Theta(\log^{1-\beta} n) \cdot \Theta(1) = \Theta(\log^{1-\beta} n)$
- $d_\star(\Psi_{l-4}) = \Theta(\log^{1-\beta} n)$

Parameters of Ψ_{l-3}. The active parameters of this system, that is obtained using the arithmetization representation-procedure, are

- $s_\star(\Psi_{l-3}) = \Theta(\log^{2(1-\beta)} n)$
- $\mathrm{D}_\star(\Psi_{l-3}) = \Theta(\log^{2(1-\beta)} n)$
- $d_\star(\Psi_{l-3}) = \Theta(\log^{1-\beta} n)$

Parameters of Ψ_{l-2}. The system Ψ_{l-2}, generated using the curve-extension representation-procedure, has parameters

- $s_\star(\Psi_{l-2}) = \Theta(\log\log n)$
- $d_\star(\Psi_{l-2}) = \Theta(\log\log n)$

Parameters of Ψ_{l-1}. This system, obtained via the arithmetization representation-procedure, is the last before the linearization representation-procedure is applied. Its parameters are

- $s_\star(\Psi_{l-1}) = \Theta(\log\log^2 n)$
- $\mathrm{D}_\star(\Psi_{l-1}) = \Theta(\log\log^2 n)$
- $d_\star(\Psi_{l-1}) = \Theta(\log\log n)$

Parameters of Ψ_l. Ψ_{l-1} obviously satisfies the conditions of the Linearization Lemma (Lemma 5.10). According to the lemma, the parameters of Ψ_l are

- $s_\star(\Psi_l) = 1$
- $\mathrm{D}_\star(\Psi_l) \leq 4$
- $d_\star(\Psi_l) = \Theta(\log\log^4 n)$

The parameters of Ψ_{sc}. By the above computations it is possible to deduce the parameters of the domains of Ψ_{sc}. Noting that $s_\star(\Psi_3)$ is the highest active low-degree parameter of all intermediate systems it follows that $s(\Psi_{sc}) = s_\star(\Psi_3) = \Theta(|\mathcal{F}|^{1/8}\log^{1-\beta} n)$. Since $r(\Psi_{sc}) = |\mathcal{F}|^{1/4}$, it follows that the requirements over s and r in Lemma 3.1 hold. The above computations also imply that the active dimension of all intermediate systems is bounded by $O(\log^{1-\beta} n)$, and hence $d(\Psi_{sc}) = \Theta(\log^{1-\beta} n)$ as required.

Polynomial time. Since Ψ_{sc} was shown to satisfy all the requirements of Lemma 3.1, it is only left to verify that it is obtained from Ψ_0 in polynomial time. Ψ_1 is obtained in polynomial time, as stated in lemma 5.5. The other intermediate systems $\Psi_2, \ldots, \Psi_l$, are obtained by applying the system-representation algorithm. As stated in Section 5.1, an application of the system-representation algorithm to a system Ψ_{i-1} takes polynomial time in the size of Ψ_{i-1} and in $|\mathcal{F}|^{d_\star(\Psi_i)}$.

According to the computations above $d_\star(\Psi_i) = O(\log^{1-\beta} n)$ for all i, so $|\mathcal{F}|^{d_\star(\Psi_i)}$ is polynomial in n. By induction it is therefore easy to verify that all intermediate systems are produced in polynomial time in n. The transformation of Ψ_l into Ψ_{sc} obviously takes polynomial time in the size of Ψ_l, so the entire reduction takes polynomial time in n.

5.5 The Product-Check Lemma

In this section we prove the product-check lemma. We show an algorithm that transforms a given quadratic-equation system into a restricted equation-system with one domain, which has a relatively small (with respect to the size of the field) dimension parameter.

The product-check algorithm actually disposes of all the variables of Ψ, substituting them by the variables of the new domain F. Each variable of Ψ and each product of two such variables is replaced by a variables of the form $F[x]$ that represent it. This is done so that for every assignment of Ψ there is a *good* assignment to F, where the value of each variable $F[x]$ is equal to the value of the corresponding term in Ψ.

However, not every feasible assignment to F indeed represents an assignment of Ψ. Consider two variables of Ψ that are represented by $F[x_1]$ and $F[x_2]$ in F. There is no guarantee that the value of the variable $F[x]$ that represents their product is indeed the product of the values of $F[x_1]$ and $F[x_2]$. Each equation of Ψ is hence replicated several times in Ψ_*, where a "product-test" is added in conjunction to each copy to verify the correctness of the assignment.

The product-check algorithm

Generating F. Let $h \doteq |\mathcal{F}|^{1/9}$, and choose $\mathcal{H} \subseteq \mathcal{F}$ to be an arbitrary set of size h. Let $d \doteq \lceil \log_h(n+1) \rceil$ (note that $d = O(\log^{1-\beta} n)$). The procedure constructs a new domain F with lower-degree parameter $s(F) = |\mathcal{F}|^{1/8}$, upper-degree $r(F) = |\mathcal{F}|^{1/4}$, and dimension $d(F) = 2d$.

Representing terms. The procedure chooses an arbitrary *injection* $v \to x_v$, associating every variable of Ψ with a point in $\mathcal{H}^d \subseteq \mathcal{F}^d$ (such an injection exists). The procedure chooses another distinct point $x_I \in \mathcal{H}^d$ to represent the value 1. Writing points in $\mathcal{F}^{2d}$ as pairs (x_1, x_2) of points in $\mathcal{F}^d$, each variable v of Ψ is represented in Ψ_* by $F[(x_v, x_I)]$, and the product of two variables u, v is represented by $F[(x_u, x_v)]$.

Generating conjunctions. The procedure replaces each equation ψ of Ψ by a set $\mathcal{E}_\psi$ of conjunctions as follows. Given ψ, it produces one conjunction in $\mathcal{E}_\psi$ for every point $(x_1, x_2) \in \mathcal{F}^{2d}$, consisting of the following equations:

1. ψ itself, where every product $u \cdot v$ is replaced by $F[(x_u, x_v)]$ and every variable v in a linear term is replaced by $F[(x_v, I)]$.
2. The product-test equation $F[(x_1, x_I)] \cdot F[(x_2, x_I)] = F[(x_1, x_2)]$.
3. The equation $F[(x_I, x_I)] = 1$, which verifies that x_I indeed represents the value 1.

From conjunctions to equations. Let Ψ' denote the system of conjunctions, containing the union of all the sets $\mathcal{E}_\psi$ where $\psi \in \Psi$. The system Ψ_* is generated from Ψ' by replacing each conjunction with all linear combinations of its equations, as described in Proposition 3.7. For every $\chi \in \Psi_*$ we set $\mathrm{Dom}_\star(\chi)$ to be F.

Observing the construction of the conjunctions and of Ψ_*, one notes that there is at most one quadratic term in each equation $\chi \in \Psi_*$. This quadratic term and the constant term of each equation χ are moved, if they exist, to the right-hand side of ψ and are set to be the core of ψ. The other terms are moved to the left-hand side, which is set to be the active part of ψ.

Proof of correctness

It is easy to observe that the product-check algorithm indeed takes polynomial time. The generated system Ψ_* has one domain F, with parameters as stated by Lemma 5.5. As also required by the lemma, the core dependency of Ψ_* is bounded by 2. It is left to show that the product-check algorithm is gap-preserving.

Completeness. Suppose Ψ is satisfiable by a good assignment $\mathcal{A}$. We show a good assignment $\mathcal{A}'$ for F which represents it, namely that

- $\mathcal{A}'(F[(x_I, x_I)]) = 1$, and for every variable v of Ψ, $\mathcal{A}'(F[(x_v, x_I)]) = \mathcal{A}(v)$.
- $F[(x_1, x_I)] \cdot F[(x_2, x_I)] = F[(x_1, x_2)]$ for every $x_1, x_2 \in \mathcal{F}^d$.

It is easy to observe that an assignment $\mathcal{A}'$ with the above properties will satisfy Ψ_*.

We define an LDF $f : \mathcal{F}^d \to \mathcal{F}$ and then use it to define $\mathcal{A}'$. For points $x_v \in \mathcal{H}^d$ associated with a variable v of Ψ we set $f(x_v) \doteq \mathcal{A}(v)$, and we also set $f(x_I) \doteq 1$. For points $x \in \mathcal{H}^d$ not associated with variables, we arbitrarily set $f(x) \doteq 0$. We extend f over $\mathcal{F}^d$ by the unique extension to an LDF of degree $h-1$ in each variable. The total degree of f is therefore $(h-1)d = O(|\mathcal{F}|^{1/9} \log n)$. $\mathcal{A}'$ will assign to F the LDF g, defined by $g(x_1, x_2) \doteq f(x_1) f(x_2)$. This is a good assignment since g is of total-degree $O(|\mathcal{F}|^{1/9} \log n) < |\mathcal{F}|^{1/8}$. The other stated properties of $\mathcal{A}'$ are easy to verify.

Soundness. The next proposition is the first step in proving the soundness property. It shows that in order for Ψ_* to be $|\mathcal{F}|^{-5/8}$-satisfiable by a feasible assignment $\mathcal{A}'$, $\mathcal{A}'$ must be consistent with an assignment $\mathcal{A}$ for Ψ. After proving the proposition we show that in that case $\mathcal{A}$ must satisfy almost the same (up to $\mathcal{F}^{-1}$) fraction of the equations in Ψ as $\mathcal{A}'$ does for Ψ_*.

Proposition 5.12. *Let $\mathcal{A}'$ be an assignment of an $r(F)$-degree LDF g to F. If it satisfies at least an $|\mathcal{F}|^{-5/8}$ fraction of the equations in Ψ_*, then there is an assignment $\mathcal{A}$ for Ψ such that for every variable v of Ψ, $\mathcal{A}(v) = g(x_v, x_I)$, and for every two variables u, v of Ψ $\mathcal{A}(u)\mathcal{A}(v) = g(x_u, x_v)$.*

Proof. Consider an assignment $\mathcal{A}'$ as above, that assigns an LDF g to F and satisfies at least a $|\mathcal{F}|^{-5/8}$ fraction of the equations of Ψ_*. We define an $[r(F), d]$-LDF f by $f(x) \doteq g(x, x_I)$, and set an assignment $\mathcal{A}$ for every variable v of Ψ by $\mathcal{A}(v) \doteq f(x_v)$ (hence the first stated property of $\mathcal{A}$ holds).

By Proposition 3.7, if $\mathcal{A}'$ satisfies more than an $|\mathcal{F}|^{-5/8}$ fraction of the equations of Ψ_*, then it satisfies an $\Omega(|\mathcal{F}|^{-5/8})$ fraction of the conjunctions in Ψ'. Then, for at least one of the equations $\psi \in \Psi$, the fraction of satisfied conjunctions in $\mathcal{E}_\psi$ is at least $\Omega(|\mathcal{F}|^{-5/8})$. By observing the product-test in each conjunction of $\mathcal{E}_\psi$, we obtain that for an $\Omega(|\mathcal{F}|^{-5/8})$ fraction of the points $(x_1, x_2) \in \mathcal{F}^{2d}$,

$$f(x_1)f(x_2) = \mathcal{A}'(F[(x_1, x_I)])\mathcal{A}'(F[(x_2, x_I)]) = \mathcal{A}'(F[(x_1, x_2)]) = g(x_1, x_2) \tag{5.1}$$

In both sides of the equation we have LDFs of degree at most $2r(F) = O(|\mathcal{F}|^{1/4})$. Different LDFs of such parameters may only agree on an $O(|\mathcal{F}|^{1/4}/|\mathcal{F}|) = O(|\mathcal{F}|^{-3/4})$ fraction of the points, however the LDFs in Equation 5.1 agree on an $\Omega(|\mathcal{F}|^{-5/8})$ fraction and are hence equal. We therefore have

$$\forall\,(x_1, x_2) \in \mathcal{F}^{2d} \quad g(x_1, x_2) = f(x_1)f(x_2)$$

and specifically

$$\forall\, v, u \quad \mathcal{A}(u)\mathcal{A}(v) = f(x_u)f(x_v) = g(x_u, x_v)$$

as required. ■

We now return to the soundness proof of the product-check procedure. Assume that Ψ_* is γ-satisfiable by a feasible assignment $\mathcal{A}'$, and let us show an assignment $\mathcal{A}$ satisfying a $\gamma - O(|\mathcal{F}|^{-1/2})$ fraction of the equations in Ψ. We may assume that $\gamma > |\mathcal{F}|^{-1/2}$ (otherwise there is nothing to show), and hence there exists an assignment $\mathcal{A}$ for Ψ that corresponds to $\mathcal{A}'$ as in Proposition 5.12.

The fraction of conjunctions in Ψ' that are satisfied by $\mathcal{A}'$ is, by Proposition 3.7, at least $\gamma - |\mathcal{F}|^{-1}$. Hence for the same fraction of equations ψ of Ψ, there is at least one conjunction $\chi \in \mathcal{E}_\psi$ which is satisfied by $\mathcal{A}'$. One of the equations in such a conjunction χ is a copy of ψ where certain terms are replaced. According to Proposition 5.12 the replaced terms have the same value as the replacing terms, and therefore ψ is satisfied by $\mathcal{A}$. This implies that at least a $\gamma - |\mathcal{F}|^{-1} > \gamma - |\mathcal{F}|^{-1/2}$ fraction of the equations of Ψ' are satisfied by $\mathcal{A}$.

5.6 The Interpolation Lemma

We now show the interpolation representation-procedure that given an equation ψ, generates a set $\mathcal{E}_\psi$ of condensed conjunctions. This is achieved utilizing the fact that the value of a good assignment to a domain F can be computed as a linear combination of its evaluations at points of its principal cube. The coefficients of the combination are independent of the LDF, and only depend on the point for which an evaluation is needed.

Let us describe the running of the algorithm for a given equation ψ in a system Ψ. Recall that ψ is of the form $\psi_\star = \psi_{\mathbf{c}}$, where $\psi_\star$ is a linear combination of variables associated with a domain $E \doteq \text{Dom}_\star(\psi)$. For shortness, let us denote $r \doteq r(E)$, $s \doteq s(E)$, $d \doteq d(E)$. Then the principal cube of E is ${\mathcal{H}_s}^d$.

Outline of the algorithm. The procedure generates the new domain $F \doteq \text{Dom}_\star(\mathcal{E}_\psi)$ with the same degree and dimension parameters as E, and hence with the same principal cube. Each element of $\mathcal{E}_\psi$ is a conjunction of two equations. One is an equation ψ', derived from ψ by replacing each of its active variables with a linear combination of variables from the principal cube of F. The replacement is done so that if F is given the same assignment as E, then the value of each active variable of ψ is the same as that of the linear combination which replaces it.

The algorithm also produces a set of equations called a *consistency-verifier*, which are not satisfied unless the assignment of F is the same as that of E (it is actually a bit more subtle, as stated in Proposition 5.14). $\mathcal{E}_\psi$ is generated by taking all the conjunctions of ψ' and a consistency-verifier equation. Hence to satisfy a large fraction of the conjunctions of $\mathcal{E}_\psi$, F and E must be given the same assignments, and therefore it is easy to conclude that ψ is satisfied as well.

The interpolation representation-procedure

We now give the details of the algorithm. First, it generates the domain $\mathrm{Dom}_\star(\mathcal{E}_\psi)$, denoted for shortness by F, with the same parameters as E, namely $s(F) = s$, $r(F) = r$, and $d(F) = d$.

Generating the consistency-verifier. The next step is the construction of the consistency-verifier, but before that we need the following claim. It states that the interpolation of an s-degree LDF from its values on the principal cube is possible.

Claim 5.13 (interpolation). *Let $s < |\mathcal{F}|$. Then there exists a polynomial algorithm that receives as input a point $x \in \mathcal{F}^d$ and outputs a* coefficient function $\kappa_x : {\mathcal{H}_s}^d \to \mathcal{F}$ *with the following properties:*

- *For every function $f' : {\mathcal{H}_s}^d \to \mathcal{F}$, the function $f : \mathcal{F}^d \to \mathcal{F}$ defined by $f(x) \doteq \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) f'(y)$ is an $[s,d]$-LDF.*
- *For any $[s,d]$-LDF f, $f(x) = \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) f(y)$.*

Proof. Each point $y \in \mathcal{F}^d$ determines an evaluation functional over the set of $[s,d]$-LDFs, the value of which on an LDF f is defined to be $f(y)$. Consider the set $\mathcal{L}$ of all evaluation functionals determined by points in ${\mathcal{H}_s}^d$. Since an $[s,d]$-LDF for which all these functionals yield zero must be the zero LDF, it follows that $\mathcal{L}$ spans all the functionals over the set of $[s,d]$-LDFs.

In particular, the evaluation functional determined by any point x can be obtained as a linear combination of functionals from $\mathcal{L}$. Let $\kappa_x(y)$ denote the coefficient, in such a linear combination, of the evaluation functional determined by y. It is easy to verify that κ_x has the desired properties, and that it can be found in polynomial time. ∎

The consistency-verifier is a set containing one equation $\chi[x]$ for each point $x \in \mathcal{F}^d$. $\chi[x]$ verifies that an evaluation of an LDF at x using κ_x (as in the claim) and the assignment of F, yields the same value as the assignment of $E[x]$:

$$\chi[x] \ : \qquad \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) F[y] = E[x]$$

Before continuing with the algorithm, we state the properties of the consistency-verifier in the following proposition.

Proposition 5.14. *Let $\mathcal{A}$ be a feasible assignment for E and F. Let f be the s-degree LDF defined by $f(x) \doteq \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) \mathcal{A}(F[y])$, as in Claim 5.13. Then either E is assigned f, in which case all of the consistency-verifier equations are satisfied, or E is not assigned F, in which case all but less than an $|\mathcal{F}|^{-1/2}$ fraction of the equations are not satisfied.*

Proof. Note that an equation $\chi[x]$ of the consistency-verifier is satisfied iff $E[x]$ is assigned $f(x)$. It is thus obvious that these equations will all be satisfied if E is assigned f. If E is not assigned f then it is assigned another LDF of degree at most $r < |\mathcal{F}|^{1/2}$. Since two different $[r, d]$-degree LDFs differ on at least a $1 - \frac{r}{|\mathcal{F}|}$ fraction of the points, $f(x)$ will equal the assignment of $E[x]$ for at most an $r/|\mathcal{F}| < |\mathcal{F}|^{-1/2}$ of the points. Therefore all except for less than an $|\mathcal{F}|^{-1/2}$ fraction of the consistency-verifier equations are not satisfied. ∎

Generating ψ'. The procedure now generates an equation ψ' from ψ by replacing each of the active variables by a linear combination of variables from the principal cube of F. A variable $E[x]$ in $\psi_\star$ is replaced by $\sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) F[y]$. The equation ψ' obtained "simulates" ψ in the case where E is assigned the interpolation of the LDF assigned to F, as is stated in the following claim.

Claim 5.15. *Let $\mathcal{A}$ be an assignment for Ψ and for F. Let f be the s-degree LDF defined by $f(x) \doteq \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) \mathcal{A}(F[y])$, as in Claim 5.13, and assume that E is assigned f. In that case ψ is satisfied by $\mathcal{A}$ if and only if ψ' is satisfied by it.*

Proof. Since E is assigned f, for every $x \in \mathcal{F}^d$, $\mathcal{A}(E[x])$ equals $f(x) = \sum_{y \in \mathcal{H}^d} \kappa_x(y) \mathcal{A}(F[y])$. Therefore the evaluations of $\psi_\star$ and ${\psi_\star}'$ are equal, hence the claim. ∎

Combining the consistency-verifier and ψ'. The last step of the interpolation representation-procedure is to generate $\mathcal{E}_\psi$. $\mathcal{E}_\psi$ will just be the set of conjunctions of ψ' and a consistency-verifier equation.

Proof of correctness

The domain F generated by the interpolation representation-procedure has the parameters required by Lemma 5.7, and the conjunctions it generates are condensed as required. It is also easy to see that the running time of the interpolation representation-procedure is polynomial in the size of ψ and in $|F|$. To complete the proof of Lemma 5.7 it only remains to show that $\mathcal{E}_\psi$ has the extension and restriction properties. The other stated properties of the interpolation representation-procedure are obvious.

- Extension: Let $\mathcal{A}$ be a good assignment for Ψ, satisfying ψ. Denote by f the (s-degree) LDF assigned to E. Extend $\mathcal{A}$ to F by assigning f to it, and let us show that this satisfies both ψ' and the consistency-verifier equations, and hence the conjunctions of $\mathcal{E}_\psi$.

  By Claim 5.13 we have

$$f(x) = \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) f(y) = \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) \mathcal{A}(F[y]) \tag{5.2}$$

  where κ_x is the coefficient function for the parameter s. Hence from Claim 5.15 it follows that the extended $\mathcal{A}$ satisfies ψ'. Equation 5.2 also implies that the consistency-verifier equations are satisfied by $\mathcal{A}$.

- Restriction: Let $\mathcal{A}$ be a feasible assignment for Ψ, and for F. We define an $[s, d]$-LDF f by

$$f(x) = \sum_{y \in {\mathcal{H}_s}^d} \kappa_x(y) \mathcal{A}(F[y])$$

  where the κ_x's are coefficient functions as stated in Claim 5.13.

  Now assume that at least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions in $\mathcal{E}_\psi$ are satisfied by $\mathcal{A}$. Then ψ' is satisfied and at least an $|\mathcal{F}|^{-1/2}$ fraction of the consistency-verifier equations are satisfied as well. By Proposition 5.14 it follows that E is assigned f. Therefore from Claim 5.15 it follows that ψ is satisfied by $\mathcal{A}$, thus proving the restriction property.

5.7 The Arithmetization Representation-Procedure

In this section we show the arithmetization representation-procedure. When applied to an equation ψ whose active LDF has small degree and dimension parameters, this procedure produces a representation $\mathcal{E}_\psi$ with small active-dependency. In essence, the arithmetization representation-procedure utilizes the *sum-check* technique, from previous PCP proofs (see [BFL91]).

We describe the running of the arithmetization representation-procedure over a given *condensed* equation ψ in a restricted equation-system Ψ. For shortness we denote $E \doteq \mathrm{Dom}_\star(\psi)$, $r \doteq r(E)$, $s \doteq s(E)$, and $d \doteq d(E)$.

Outline of the algorithm. The equation ψ has the form $\psi_\star = \psi_{\mathbf{c}}$. Since ψ is condensed, $\psi_\star$ can be written as a sum

$$\sum_{y \in {\mathcal{H}_s}^d} \kappa(y) E[y] \qquad (\psi_\star)$$

where $\kappa : {\mathcal{H}_s}^d \to \mathcal{F}$ is a coefficient function. The assignment of the domain $F \doteq \mathrm{Dom}_\star(\mathcal{E}_\psi)$, that is generated by the arithmetization representation-procedure, encodes the summands in $\psi_\star$ and also many of the partial sums. If the assignment of F is a correct encoding then it is possible to evaluate $\psi_\star$ by accessing only a few variables of F, since their values are evaluations of large partial sums of $\psi_\star$.

Each conjunction of $\mathcal{E}_\psi$ tests whether the assignment of F is a correct encoding, and assuming that the encoding is indeed correct, verifies that ψ holds. This is accomplished by accessing only a small number of variables of F. To satisfy more than an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions in $\mathcal{E}_\psi$ the assignment of F must encode the summands in the active part of ψ correctly, and also ψ must be satisfied.

The assignment for F that correctly encodes the summands of ψ is called the sum-check LDF, and is defined below. Values of the sum-check LDF at some points are evaluations of certain partial sums of $\psi_\star$, and values of the sum-check LDF at other points are used in testing the correctness of the encoding. It is necessary to define how the assignment of F should encode the summands in $\psi_\star$ before the construction of $\mathcal{E}_\psi$ can be understood, so we first describe the sum-check LDF and only then continue to the construction.

The sum-check LDF

We define the sum-check LDF of ψ with respect to a given good assignment $\mathcal{A}$ for E. First, we extend κ to an LDF of degree ds over $\mathcal{F}^d$ – such an extension exists and is computable in polynomial time in $|\mathcal{F}|^d < |F|$, and hence it is possible to compute the extension within the representation-procedure. We now define d LDFs that encode different partial sums of $\psi_\star$. The sum-check LDF is constructed from these LDFs below.

Definition 5.16 (the sum-check tree). *For $k = 1, 2, \ldots, d$, we define a function $g_k : \mathcal{F}^k \to \mathcal{F}$ by*

$$\forall x \in \mathcal{F}^k \quad g_k(x) \doteq \sum_{y \in {\mathcal{H}_s}^{(d-k)}} \kappa(x, y) \mathcal{A}(E[x, y])$$

where "x, y" means the concatenation of the vector x and the vector y. The sequence $g_1, \ldots, g_d$ is called the sum-check tree with respect to $\mathcal{A}(E)$.

For an $x \in {\mathcal{H}_s}^k$ the value of $g_k(x)$ is a partial sum of $\psi_\star$. The value of g_d at a point $x \in \mathcal{F}^d$ is just $\kappa(x)\mathcal{A}(E[x])$, and hence g_d is an LDF of degree at most $ds + s = (d+1)s$. It follows from the above definition that the other g_k's have degree at most $(d+1)s$ as well.

The LDFs $g_1, \ldots, g_d$ form a tree of partial sums in the following sense. Consider a tree of depth d, where every non-leaf node has $|\mathcal{F}|$ offsprings, and every node of depth $k > 0$ is labeled by a point evaluation of g_k. We label the root by $\sum_{y \in \mathcal{H}_s{}^d} \kappa(y)\mathcal{A}(E[y])$, which is the evaluation of $\psi_\star$. The root has an offspring labeled by $g_1(z)$, for each $z \in \mathcal{F}$. Note that for $z \in \mathcal{H}_s$, $g_1(z)$ is a partial sum of $\psi_\star$, and in fact the root-label is the sum of labels of its offsprings that are assigned $g_1(z)$ for $z \in \mathcal{H}_s$.

For a non-leaf node that has been labeled $g_k(x)$, we label one of its offsprings by $g_{k+1}(x, z)$ for every $z \in \mathcal{F}$. From the definition of the g_k's it follows that for every $k < d$ and $x \in \mathcal{F}^k$,

$$g_k(x) = \sum_{z \in \mathcal{H}_s} g_{k+1}(x, z) \tag{5.3}$$

Hence the label of each node labeled $g_k(x)$ in the tree is the sum of labels of its $s + 1$ offsprings that are assigned $g_{k+1}(x, z)$ for $z \in \mathcal{H}_s$.

The sum-check LDF. We now incorporate all the LDFs $g_1, \ldots, g_k$ into a single LDF of degree at most $(d + 1)s + d \leq 2ds$, called the sum-check LDF. For this purpose, let $\mathcal{H}_{d-1} = \{a_1, \ldots, a_d\}$ be an arbitrary subset of size d in $\mathcal{F}$. The sum-check LDF, denoted by f, will satisfy

$$f(a_k, x_1, \ldots, x_k, 0, \ldots, 0) = g_k(x_1, \ldots, x_k) \tag{5.4}$$

for every $1 \leq k \leq d$, and every $x = (x_1, \ldots, x_k) \in \mathcal{F}^k$. There exists such an f – for example it can be defined by

$$f(x_0, x_1, \ldots, x_d) \doteq \sum_{k=1}^{d} \left(\prod_{i=1, i \neq k}^{d} \frac{x_0 - a_i}{a_k - a_i} \right) \cdot g_k(x_1, .., x_k)$$

Properties of the sum-check LDF. From Equation 5.4 and the discussion above it follows that the sum-check LDF has the following properties:

- $\sum_{z \in \mathcal{H}_s} f[(1, z, 0, \ldots, 0)]$ is the evaluation of $\psi_\star$, as follows from the explanation after Definition 5.16.

- For $k = 1, 2, \ldots, (d - 1)$ and every $(x_1, \ldots, x_k) \in \mathcal{F}^k$

  $$f[(a_k, x_1, \ldots, x_k, 0, \ldots, 0)] = \sum_{z \in \mathcal{H}_s} f[(a_{k+1}, x_1, \ldots, x_k, z, 0, \ldots, 0)]$$

  as follows from Equation 5.3.

- For every $(x_1, \ldots, x_d) \in \mathcal{F}^d$,

  $$f[(a_d, x_1, \ldots, x_d)] = \kappa(x_1, \ldots, x_d)\mathcal{A}(E[x_1, \ldots, x_d])$$

  as follows from the explanation after Definition 5.16.

The arithmetization representation-procedure

We now give the details of the arithmetization representation procedure. At first the representation-procedure produces a new domain $F = \mathrm{Dom}_\star(\mathcal{E}_\psi)$ with parameters as stated in Lemma 5.8, namely $r(F) = r$, $s(F) = 2ds$ and $d(F) = d + 1$. The procedure generates conjunctions that can only be satisfied if F is assigned the sum-check f. For each $x = (x_1, \ldots, x_d) \in \mathcal{F}^d$ the procedure generates one conjunction, denoted by $\chi[x]$, consisting of the following $d + 1$ equations:

- The *root equation*:
$$\sum_{z \in \mathcal{H}_s} F[a_1, z, 0, \ldots, 0] = \psi_{\mathbf{c}}$$
- The $d - 1$ *path equations* for $k = 1, 2, \ldots, (d - 1)$:
$$F[a_k, x_1, \ldots, x_k, 0, \ldots, 0] = \sum_{z \in \mathcal{H}_z} F[a_{k+1}, x_1, \ldots, x_k, z, 0, \ldots, 0]$$
- The *leaf* equation:
$$F[a_d, x_1, \ldots, x_d] = \kappa(x_1, \ldots, x_d) E[x_1, \ldots, x_d]$$

Proof of correctness

Let us show that the arithmetization representation-procedure has the required properties. It is easy to verify that it runs in polynomial time, and that it generates a domain $F = \mathrm{Dom}_\star(\psi)$ with parameters as required. As to the number of active variables in each conjunction, there are $s+1$ variables associated with F in the root equation, $s+2$ variables in each of the $d-1$ path equations, and one variable in the leaf equation. The total number is therefore $s + 1 + (d - 1)(s + 2) + 1 = ds + 2d \leq 2ds$ as required. It is left only to verify the extension and restriction properties.

Extension. Let $\mathcal{A}$ be a good assignment for the variables of Ψ. Extend $\mathcal{A}$ to F by assigning the sum-check LDF f to it (f is of degree less than $s(F)$). From the properties of f stated above, it easily follows that if ψ is satisfied by $\mathcal{A}$ then all the conjunctions of $\mathcal{E}_\psi$ are also satisfied by the extension of $\mathcal{A}$.

Restriction. Let $\mathcal{A}$ be a feasible assignment for the variables of Ψ and for F, and assume that at least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions in $\mathcal{E}_\psi$ are satisfied. We define the sum-check tree $g_1, \ldots, g_d$ and the sum-check LDF f with respect to the assignment of E, as in Definition 5.16 and Equation 5.4 above. Since now the degree of the LDF assigned to E may be up to r, the degree of the g_k's can be up to $sd + r < |\mathcal{F}|^{1/2}$. We claim that F must be assigned f, as shown in the following claim.

Claim 5.17 (sum-check). *Suppose that at least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions in $\mathcal{E}_\psi$ are satisfied by a feasible assignment. Then for every k, $1 \leq k \leq d$, and every $x = (x_1, \ldots, x_k) \in \mathcal{F}^k$,*

$$\mathcal{A}(F[a_k, x_1, \ldots, x_k, 0, \ldots, 0]) = g_k(x_1, \ldots, x_k)$$

Before proving the claim we show how it implies the restriction property. Note that the root equation is common to all the conjunctions in $\mathcal{E}_\psi$, and hence it must be satisfied. So together with the claim we have that the evaluation of $\psi_{\mathbf{c}}$ equals $\sum_{z\in\mathcal{H}_s} f(a_1, z, 0, \ldots, 0)$, which by the properties of the sum-check LDF equals the evaluation of $\psi_\star$. Therefore ψ is satisfied, as required.

Proof of the sum-check claim. For every k, $1 \leq k \leq d$, we define an $[r, k]$-degree LDF g'_k by

$$g'_k(x_1, \ldots, x_k) \doteq \mathcal{A}(F[a_k, x_1, \ldots, x_k, 0, \ldots, 0])$$

For the sake of contradiction, assume that $g'_k \neq g_k$ for some k, and choose k to be the highest for which this inequality holds. We distinguish between two cases for k:

- $k = d$: At least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions of $\mathcal{E}_\psi$ are satisfied, and therefore at least the same fraction of the leaf equations are satisfied. So for at least an $|\mathcal{F}|^{-1/2}$ fraction of the points $x \in \mathcal{F}^d$, $g'_d(x) = \mathcal{A}(F[a_d, x]) = \kappa(x)\mathcal{A}(E[x]) = g_d(x)$. But according to our assumption $g'_d \neq g_d$ and therefore their evaluations can not be equal on more than an $\frac{sd+r}{|\mathcal{F}|} < |\mathcal{F}|^{-1/2}$ fraction of the points, a contradiction.

- $1 \leq k < d$: At least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions of $\mathcal{E}_\psi$ are satisfied, and therefore in at least the same fraction of them the k'th path equation is satisfied. It follows that for at least an $|\mathcal{F}|^{-1/2}$ fraction of the points $x = (x_1, \ldots, x_k) \in \mathcal{F}^k$,

  $$\begin{aligned} g'_k(x) &= \mathcal{A}(F[a_k, x_1, \ldots, x_k, 0, \ldots, 0]) = \\ &= \sum_{z\in\mathcal{H}_s} \mathcal{A}(F[a_{k+1}, x_1, \ldots, x_k, z, 0, \ldots, 0]) = \\ &= \sum_{z\in\mathcal{H}_s} g'_{k+1}(x_1, \ldots, x_k, z) \end{aligned}$$

  By the maximality of k we have that $g'_{k+1} = g_{k+1}$, hence for at least an $|\mathcal{F}|^{-1/2}$ fraction of the points x,

  $$g'_k(x) = \sum_{z\in\mathcal{H}_s} g_{k+1}(x_1, \ldots, x_k, z) = g_k(x) \quad \text{(by Equation 5.3)}$$

  This is a contradiction to our assumption that $g'_k \neq g_k$, since they are both of degree at most $sd + r$ and therefore our assumption implies that they can be equal on at most an $\frac{sd+r}{|\mathcal{F}|} < |\mathcal{F}|^{-1/2}$ fraction of the points.

5.8 The Curve-Extension Representation-Procedure

In this section we show the curve-extension representation-procedure. If it is applied to an equation with a small enough active-dependency, then the new generated domain has a small active lower-degree parameter, and for equations with even smaller active-dependency the active dimension parameter becomes small as well. The conjunctions that are generated by the procedure are all condensed.

Let us describe the running of the curve-extension representation-procedure over a given equation ψ. For shortness we denote $E \doteq \mathrm{Dom}_\star(\psi)$, $r \doteq r(E)$, $s \doteq s(E)$, $d \doteq d(E)$, and $D \doteq \mathrm{D}_\star(\psi)$.

The principle of the algorithm. Denote the active variables of ψ by $E[x_1], \ldots, E[x_D]$. We define below a polynomial vector function of small degree $\Gamma : \mathcal{F} \rightarrow \mathcal{F}^d$, that goes through the points $x_1, \ldots, x_D$. The assignment of the domain $F \doteq \mathrm{Dom}_\star(\mathcal{E}_\psi)$, generated by the curve-extension representation-procedure, encodes the restriction of the assignment of E to the points of the curve Γ.

Variables in F associated with certain points in its principal cube have, in a correct encoding, the values of the assignment of E at certain points on Γ. The values at other points on Γ can be computed by interpolation over these variables of F, making use of the fact that Γ has a small degree, and hence restricting the assignment of E to its points yields an LDF of small degree as well. The conjunctions of $\mathcal{E}_\psi$ use the variables of F to evaluate $\psi_\star$ and verify that ψ is satisfied, and they also test whether F is indeed given a correct encoding.

The curve-extension algorithm

At first the representation-procedure produces a new domain $F = \mathrm{Dom}_\star(\mathcal{E}_\psi)$ with parameters as stated in Lemma 5.9, that is

$$r(F) = r, \quad d(F) = \min\{d, \log_2(s\mathrm{D})\}, \quad \text{and} \quad s(F) = d(F) \cdot \max\left\{\left(s \cdot \mathrm{D}\right)^{1/d}, 2\right\}$$

Each element of $\mathcal{E}_\psi$ will be a conjunction of two condensed equations. One is an equation ψ', derived from ψ by replacing each of its active variables with a variable of F that "encodes" it. The other equation is taken from a set of equations called a *curve-verifier*. These equations are not satisfied unless the assignment of F is a correct encoding. Before the construction of these equations, we define the curve Γ and describe how the assignment of F encodes the restriction of the assignment of E to the points of Γ.

Definition 5.18 (the curve Γ). *Let $\mathcal{H}_{sD-1}$ be an arbitrary subset of $\mathcal{F}$ of size sD, and denote its elements by $a_1, \ldots, a_{sD}$. $\Gamma : \mathcal{F} \rightarrow \mathcal{F}^d$ is defined to be the $(D-1)$-degree polynomial vector function satisfying*

$$\forall\, 1 \le i \le D \qquad \Gamma(a_i) = x_i$$

where $E[x_1], \ldots, E[x_D]$ are the active variables of ψ. Γ can clearly be computed in polynomial-time.

Associating points with $a_1, \ldots, a_{sD}$. Let $\left(\mathcal{H}_{s(F)}\right)^{d(F)}$ be the principal cube of F. The procedure chooses an arbitrary subset $\mathcal{H} \subseteq \mathcal{H}_{s(F)}$ of size $s(F)/d(F) = \max\left\{\left(s \cdot \mathrm{D}\right)^{1/d}, 2\right\}$, and associates to each point a_i in $\mathcal{H}_{sD-1}$ a distinct point y_i in $\mathcal{H}^{d(F)}$ (note that $\mathcal{H}^{d(F)}$ is a subset of the principal cube of F and that it contains at least sD points). Each of the variables $F[y_i]$ will encode the value of $E[\Gamma(a_i)]$. The active variables of the conjunctions in $\mathcal{E}_\psi$ will all be of the form $F[y_i]$, so the conjunctions of $\mathcal{E}_\psi$ are condensed. It is important to note that any assignment to the variables $F[y_i]$ can be extended by interpolation to a good assignment for F, as is shown below.

Generating the curve-verifier. Suppose E is assigned an LDF g. Then a correct encoding assigns to $F[y_i]$ the value of g at $\Gamma(a_i)$. Since Γ is of degree at most $D-1$, if g is of degree s then $g \circ \Gamma$ is of degree less than $sD-1$. The value of g at any point on the curve Γ can hence be evaluated by interpolation over its values at $\Gamma(a_1), \ldots, \Gamma(a_{sD})$ or, if F is assigned a correct encoding, by interpolation over the variables $F[y_1], \ldots, F[y_{sD}]$. This is stated precisely in the following claim, which is the one-dimensional equivalent of Claim 5.13.

Claim 5.19 (curve-interpolation). *Let s and D be such that $sD < |\mathcal{F}|$. Then there exists a polynomial (in $|\mathcal{F}|$) algorithm that receives as input a point $x \in \mathcal{F}$ and outputs a* coefficient function *$\kappa_x : \{a_1, \ldots, a_{sD}\} \to \mathcal{F}$ with the following property: Every function $f' : \{a_1, \ldots, a_{sD}\} \to \mathcal{F}$ has a unique extension to an $[sD-1, 1]$-LDF f over $\mathcal{F}$, and f satisfies*

$$\forall\, x \in \mathcal{F} \qquad f(x) = \sum_{i=1}^{sD} \kappa_x(a_i) f'(a_i)$$

The curve-verifier will have one equation $\chi[x]$ for each point $x \in \mathcal{F}$. $\chi[x]$ verifies that the interpolation over $F[y_1], \ldots, F[y_{sD}]$ using the κ_x from Claim 5.19 yields the value of $E[\Gamma(x)]$, as it should if F is assigned the encoding of a good assignment to E:

$$\chi[x] \;: \qquad \sum_{i=1}^{sD} \kappa_x(a_i) F[y_i] = E[\Gamma(x)]$$

The next proposition shows that the curve-verifier equations cannot be satisfied unless F is indeed assigned a correct encoding.

Proposition 5.20. *Let $\mathcal{A}$ be a feasible assignment for E and F. Let f be the $[sD-1, 1]$-LDF defined by $f(x) = \sum_{i=1}^{sD} \kappa_x(a_i) \mathcal{A}(F[y_i])$, as in Claim 5.19. Then either $\mathcal{A}(E) \circ \Gamma = f$, in which case all of the curve-verifier equations are satisfied, or less than an $|\mathcal{F}|^{-1/2}$ fraction of the curve-verifier equations are satisfied.*

Proof. Note that an equation $\chi[x]$ of the curve-verifier is satisfied if and only if $E[\Gamma(x)]$ is assigned $f(x)$. It is thus obvious that these equations will all be satisfied if $\mathcal{A}(E) \circ \Gamma = f$. If this is not the case, then $\mathcal{A}(E) \circ \Gamma$ and f are in particular two different $[rD, 1]$-LDFs. Since $rD < |\mathcal{F}|^{1/2}$ it follows that their evaluations differ on all but less than an $|\mathcal{F}|^{1/2}/|\mathcal{F}| \leq |\mathcal{F}|^{-1/2}$ fraction of the points. Hence if $\mathcal{A}(E) \circ \Gamma \neq f$, then less than an $|\mathcal{F}|^{-1/2}$ fraction of the curve-verifier equations can be satisfied. ∎

Generating ψ'. The procedure generates an equation ψ' by replacing each active variable $E[x_i]$ in $\psi_\star$ with the variable $F[y_i]$. If F is assigned a correct encoding then ψ' simulates ψ, as stated in the following claim.

Claim 5.21. *Let $\mathcal{A}$ be an assignment for Ψ and for F. Let f be the $[sD-1, 1]$-degree LDF defined by $f(x) = \sum_{i=1}^{sD} \kappa_x(a_i)\mathcal{A}(F[y_i])$, as in Claim 5.19, and assume that $\mathcal{A}(E) \circ \Gamma = f$. In that case ψ is satisfied by $\mathcal{A}$ if and only if ψ' is satisfied by it.*

Proof. According to the definition of Γ, $\Gamma(a_i) = x_i$ for $i = 1, \ldots, D$. Hence it follows from the assumption that $\mathcal{A}(E[x_i]) = \mathcal{A}(E[\Gamma(a_i)]) = f(a_i)$ for every i, $1 \leq i \leq D$. But according to Claim 5.19 $f(a_i) = \mathcal{A}(F[y_i])$ for every i. Therefore the assignment of every active variable $E[x_i]$ equals the assignment of $F[y_i]$. The claim immediately follows. ∎

Generating $\mathcal{E}_\psi$. The set $\mathcal{E}_\psi$ is composed of all the conjunctions of ψ' and an equation $\chi[x]$ of the curve-verifier.

Proof of correctness

The domain F that is generated by the curve-extension representation-procedure has the parameters required by Lemma 5.9, and the conjunctions of $\mathcal{E}_\psi$ are all condensed. It is also easy to verify that the curve-extension representation-procedure takes polynomial time in the size of ψ and in $|F|$. To complete the proof of Lemma 5.9 it remains to show that $\mathcal{E}_\psi$ has the extension and restriction properties. The other properties required of a representation-procedure are obvious.

- Extension: Let $\mathcal{A}$ be a good assignment for the variables of Ψ that satisfies ψ. We extend $\mathcal{A}$ by assigning an $s(F)$-degree LDF to F such that all the conjunctions of $\mathcal{E}_\psi$ are satisfied. The LDF g, to be assigned to F, is defined as follows. First, let $g(y_i) \doteq \mathcal{A}(E[\Gamma(a_i)])$ for $i = 1, \ldots, sD$. Since all the y_i's are contained in $\mathcal{H}^{d(F)}$, there exists an extension of g to an LDF over $\mathcal{F}^{d(F)}$ of degree at most $s(F)/d(F)$ in each variable. The total degree of this g is hence at most $s(F)$. We assign g to F. Then $\mathcal{A}(F[y_i]) = \mathcal{A}(E[\Gamma(a_i)])$ for every i, and so Claim 5.21 implies that ψ' is satisfied.

Let f be the $[sD-1,1]$-LDF defined by $f \doteq \mathcal{A}(E) \circ \Gamma$. Then by Claim 5.19,

$$\forall\, x \in \mathcal{F} \qquad f(x) = \sum_{i=1}^{sD} \kappa_x(a_i) f(a_i) = \sum_{i=1}^{sD} \kappa_x(a_i) \mathcal{A}(E[\Gamma(a_i)])$$
$$= \sum_{i=1}^{sD} \kappa_x(a_i) \mathcal{A}(F[y_i])$$

where the coefficients $\kappa_x(a_i)$ are as in Claim 5.19. It hence follows that the curve-verifier equations are all satisfied by the extended $\mathcal{A}$. Since $\mathcal{E}_\psi$ consists of conjunctions of ψ' and equations of the curve-verifier, we have that all of its conjunctions are satisfied.

- Restriction: Let $\mathcal{A}$ be a feasible assignment for the variables of Ψ and for F, and assume that at least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions in $\mathcal{E}_\psi$ are satisfied by $\mathcal{A}$. Since ψ' appears in every conjunction of $\mathcal{E}_\psi$, ψ' is satisfied, and at least an $|\mathcal{F}|^{-1/2}$ fraction of the curve-verifier equations are satisfied as well.

  Define an $(sD-1)$-degree LDF f by

$$\forall\, x \in \mathcal{F} \qquad f(x) \doteq \sum_{i=1}^{sD} \kappa_x(a_i) \mathcal{A}(F[y_i])$$

  where the coefficients $\kappa_x(a_i)$ are as in Claim 5.19. It follows from Proposition 5.20 that $\mathcal{A}(E) \circ \Gamma = f$. Since ψ' is satisfied, it then follows from Claim 5.21 that ψ is satisfied as well, thereby proving the restriction property.

5.9 The Linearization Representation-Procedure

In this section we show the Linearization representation-procedure. It is the final representation-procedure used in the sequence of transformations, resulting in a system of a constant active-dependency parameter.

The linearization representation-procedure is similar to the curve-extension. When applied to an equation ψ, it uses the newly generated domain to encode the restriction of the assignment of $\mathrm{Dom}_\star(\psi)$ to a curve that contains the active variables of ψ. The curve-extension representation-procedure encoded directly the assignment at only some points of the curve; to obtain other evaluations it applied interpolation by computing an appropriate linear-combinations over the encoded values.

The linearization representation-procedure applies a method of [ALM$^+$98], using the newly generated domain to encode all linear-combinations of these values. Hence each curve-verifier equation requires just one active variable of the new domain. Also, since the active part of ψ is a linear-combination of variables associated with points on the curve, $\psi_\star$ can also be evaluated using one access to the new domain.

The linearization representation-procedure

We now describe the linearization representation-procedure. We fix the notations $E \doteq \mathrm{Dom}_\star(\psi)$, $r \doteq r(E)$, $s \doteq s(E)$, $d \doteq d(E)$, and $D \doteq \mathrm{D}_\star(\psi)$. The linearization representation-procedure first generates a new domain F with parameters as stated in Lemma 5.10, that is

$$r(F) = r,\ s(F) = 1, \quad \text{and} \quad d(F) = sD$$

In each conjunction in $\mathcal{E}_\psi$ there will be an equation ψ', that is derived by replacing the active part of ψ with a variable of F that encodes it. Another equation in each conjunction is taken from a set of equations called a *linearization-verifier*, that are not satisfied unless F is assigned a homogeneous linear-LDF. As in the curve-extension representation-procedure, the last equation in each conjunction is taken from a set called the *curve-verifier*, whose equations are not satisfied unless the assignment of F is a correct encoding.

Generation of the linearization-verifier. The linearization-verifier has one equation $\chi[y,t]$ for every $y \in \mathcal{F}^{d(F)}$ and $t \in \mathcal{F}$:

$$\chi[y,t]\ : \qquad tF[y] = F[ty]$$

The equations of the linearization-verifier are not satisfied unless F is assigned a linear homogeneous LDF. To prove it we need the following observation.

Claim 5.22. *Let f be an $[r(F), d(F)]$-LDF which is not linear homogeneous. For every point $y \in \mathcal{F}^{sD}$ we define an $[r(F), 1]$-LDF ϕ_y by*

$$\forall\, t \in \mathcal{F} \qquad \phi_y(t) \doteq f(ty)$$

Then ϕ_y is linear homogeneous for less than an $(|\mathcal{F}|^{-1/2})/2$ fraction of the points y.

Proof. If f is linear but not homogeneous then obviously none of the ϕ_y's are homogeneous, so we assume that $m \doteq \deg(f) > 1$. Then f can be written as a sum $f = f_1 + f_2$ where f_1 is a homogeneous[†] function of degree m, and $\deg(f_2) < m$. Then $\phi_y(t) = f_1(y)t^m + f_2(ty)$. The second term in this sum is of degree less than m as an LDF over t. The first term is an LDF of degree exactly m in t (and in particular it is not linear), unless $f_1(y) = 0$.

It follows that ϕ_y is not linear (and in particular not linear-homogeneous) unless $f_1(y) = 0$, which occurs for at most an $\frac{m}{|\mathcal{F}|} \le \frac{r(F)}{|\mathcal{F}|} \le |\mathcal{F}|^{-1/2}/2$ fraction of the points y, as we needed to show. ■

The next proposition shows that the linearization-verifier equations are indeed satisfied only if F is assigned a linear-homogeneous function.

[†] f is homogeneous of degree m iff $f(ty) = t^m f(y)$ for every y and t.

Proposition 5.23. *Let $\mathcal{A}$ be a feasible assignment for F. Then less than an $|\mathcal{F}|^{-1/2}$ fraction of the linearization-verifier equations are satisfied unless F is assigned a linear-homogeneous function, in which case all of the equations are satisfied.*

Proof. It is clear that the linearization-verifier equations are all satisfied in the case that F is assigned a linear-homogeneous function. We therefore suppose that $\mathcal{A}(F)$ is not linear homogeneous, and prove that less than an $|\mathcal{F}|^{-1/2}$ fraction of the equations are satisfied. Define for every $y \in \mathcal{F}^{d(F)}$ the $[r(F), 1]$-LDF ϕ_y as in the proof of Claim 5.22:

$$\forall\, t \in \mathcal{F} \qquad \phi_y(t) \doteq \mathcal{A}(F[ty])$$

As proven there, ϕ_y is linear-homogeneous for less than an $(|\mathcal{F}|^{-1/2})/2$ fraction of the points y.

Consider a point y for which ϕ_y is not linear-homogeneous. Since $t\mathcal{A}(F[y])$ is linear-homogeneous as a function of t, $t\mathcal{A}(F[y]) \neq \mathcal{A}(F[ty])$ for all but at most an $\frac{r(F)}{|\mathcal{F}|} \leq (|\mathcal{F}|^{-1/2})/2$ fraction of the t's, and hence at most an $(|\mathcal{F}|^{-1/2})/2$ fraction of the equations $\chi[y, t]$ are satisfied.

We have shown that for at least a $1 - (|\mathcal{F}|^{-1/2})/2$ fraction of the y's ϕ_y is not linear-homogeneous, and that for such y's $\chi[y, t]$ is satisfied for less than an $(|\mathcal{F}|^{-1/2})/2$ fraction of the t's. It follows that less than an $|\mathcal{F}|^{-1/2}$ fraction of the equations are satisfied, as we needed to show. ■

The curve Γ. Write the active part of ψ as

$$\psi_\star \quad : \qquad \sum_{i=1}^{D} \alpha_j E[x_i]$$

As in the curve-extension representation-procedure, we define a curve $\Gamma : \mathcal{F} \to \mathcal{F}^d$ which goes through the points associated with the active variables of ψ.

Definition 5.24 (the curve of ψ). *Let $\mathcal{H}_{sD-1} = \{a_1, \ldots, a_{sD}\}$ be an arbitrary subset of $\mathcal{F}$. Define $\Gamma : \mathcal{F} \to \mathcal{F}^d$ to be the vector of $(D-1)$-degree polynomial functions satisfying*

$$\forall\, 1 \leq i \leq D \qquad \Gamma(a_i) = x_i$$

Given an assignment $\mathcal{A}$ for E, the assignment of F is used as an encoding of $\mathcal{A}(E) \circ \Gamma$. Unlike in the curve-extension representation-procedure, the correct encoding here is a linear-homogeneous LDF.

The encoding. The procedure generates a curve-verifier, whose equations are only satisfied if the assignment of F is the correct encoding of $\mathcal{A}(E) \circ \Gamma$. To define what the correct encoding is, suppose E is assigned an s-degree LDF g. The LDF $g \circ \Gamma : \mathcal{F} \to \mathcal{F}$, which is to be encoded by the assignment of F, has degree at most $sD - 1$. Its encoding is the following linear-homogeneous LDF, L_g:

$$\forall\ (x_1, \ldots, x_{sD}) \in \mathcal{F}^{sD} \qquad L_g(x_1, \ldots, x_{sD}) \doteq \sum_{i=1}^{sD} x_i g(\Gamma(a_i))$$

The next claim shows how $g \circ \Gamma$ can be reconstructed, given L_g.

Claim 5.25 (linearizing-interpolation). *For $i = 1, \ldots, sD$ let γ_i be the $[sD-1, 1]$-LDF satisfying $\gamma_i(a_i) = 1$ and $\gamma_i(a_j) = 0$ for every $j \neq i$.*

Then the polynomial vector function $\widehat{\gamma} = (\gamma_1, \ldots, \gamma_{sD})$ satisfies $L_g \circ \widehat{\gamma} = g \circ \Gamma$.

Proof. L_g is linear, hence $L_g \circ \widehat{\gamma}$ is of degree at most $sD - 1$. Since it follows from the definition of L_g that $L_g \circ \widehat{\gamma}(a_i) = g(\Gamma(a_i))$ for $i = 1, \ldots, sD$, we obtain that $L_g \circ \widehat{\gamma} = g \circ \Gamma$ (also recall that $g \circ \Gamma$ is of degree at most $sD - 1$). ■

Generating the curve-verifier. It follows from Claim 5.25 that if an assignment $\mathcal{A}$ assigns a good LDF to E and assigns its encoding to F, then $\mathcal{A}(F[\widehat{\gamma}(x)]) = \mathcal{A}(E[\Gamma(x)])$ for every $x \in \mathcal{F}$. To verify that F is assigned a correct encoding, the representation-procedure generates one equation $\chi[x]$ in the curve-verifier for every $x \in \mathcal{F}$ as follows:

$$\chi[x] : \qquad F[\widehat{\gamma}(x)] = E[\Gamma(x)]$$

where the vector-function $\widehat{\gamma}$ is as defined in Claim 5.25.

The following proposition shows that indeed the curve-verifier equations are not satisfied unless the assignment for F is a correct encoding, in the sense that $\mathcal{A}(F) \circ \widehat{\gamma} = \mathcal{A}(E) \circ \Gamma$. It is assumed that F is assigned a linear LDF, since otherwise the linearization-verifier equations cannot be satisfied.

Proposition 5.26. *Let $\mathcal{A}$ be a feasible assignment for E and F, assigning a linear LDF to F. Then less than an $|\mathcal{F}|^{-1/2}$ fraction of the curve-verifier equations are satisfied unless $\mathcal{A}(F) \circ \widehat{\gamma} = \mathcal{A}(E) \circ \Gamma$, in which case all of the equations are satisfied.*

Proof. If $\mathcal{A}(F) \circ \widehat{\gamma} = \mathcal{A}(E) \circ \Gamma$ then for every $x \in \mathcal{F}$, $\mathcal{A}(F[\widehat{\gamma}(x)]) = \mathcal{A}(E[\Gamma(x)])$ and hence $\chi[x]$ is satisfied.

Now assume that at least an $|\mathcal{F}|^{-1/2}$ fraction of the equations $\chi[x]$ are satisfied. According to this assumption, $\mathcal{A}(F[\widehat{\gamma}(x)]) = \mathcal{A}(E[\Gamma(x)])$ for at least an $|\mathcal{F}|^{-1/2}$ fraction of the x's. Since $\mathcal{A}(F) \circ \widehat{\gamma}$ is an LDF of degree at most $sD - 1 < |\mathcal{F}|^{-1/2}$ and $\mathcal{A}(E) \circ \Gamma$ is an LDF of degree at most $r(D-1) < |\mathcal{F}|^{-1/2}$, this implies that $\mathcal{A}(F) \circ \widehat{\gamma} = \mathcal{A}(E) \circ \Gamma$. ■

Generating ψ'. The procedure now generates the equation ψ' such that when F is assigned a correct encoding, ψ' is satisfied if and only if ψ is satisfied. ψ' is obtained from ψ by removing $\psi_\star$ and replacing it with one variable $F[y_*]$, where y_* is as specified by the following claim.

Claim 5.27. *There exists a point $y_* \in \mathcal{F}^{d(F)}$ such that for any feasible assignment $\mathcal{A}$ for E and F, for which $\mathcal{A}(F)$ is a correct encoding of $\mathcal{A}(E)$, $\mathcal{A}(F[y_*])$ equals the evaluation of $\psi_\star$. Moreover, y_* can be found, given ψ, in time polynomial in the size of ψ and in $|F|$.*

Proof. For every assignment $\mathcal{A}$ satisfying the above conditions $\mathcal{A}(F)$ is linear-homogeneous and hence $\mathcal{A}(F) \circ \widehat{\gamma}$ is of degree at most $sD-1$. $\mathcal{A}(E) \circ \Gamma$ is therefore an $(sD-1)$-degree LDF as well. According to the curve-interpolation claim (Claim 5.19), for every $x \in \mathcal{F}$ one can find in polynomial time (and independently of $\mathcal{A}$) a coefficient function κ_x such that

$$\mathcal{A}(E[\Gamma(x)]) = \sum_{i=1}^{sD} \kappa_x(a_i)\mathcal{A}(E[\Gamma(a_i)]) = \sum_{i=1}^{sD} \kappa_x(a_i)\mathcal{A}(F[\widehat{\gamma}(a_i)]) =$$

$$= \mathcal{A}\left(F\big[\sum_{i=1}^{sD} \kappa_x(a_i)\widehat{\gamma}(a_i)\big]\right)$$

where the last equality occurs because F is assigned a linear-homogeneous LDF.

Denote $y_x \doteq \sum_{i=1}^{sD} \kappa_x(a_i)\Gamma(a_i)$ for every $x \in \mathcal{F}$, and recall that $\psi_\star$ has the form $\sum_{j=1}^{D} \alpha_j E[x_j]$. We have from the equation above that $\mathcal{A}(E[x_j]) = \mathcal{A}(F[y_{x_j}])$ for every j, and therefore since $\mathcal{A}(F)$ is linear-homogeneous we conclude that the evaluation of $\psi_\star$ equals the assignment of $F[y_*]$ for $y_* \doteq \sum_{j=1}^{D} \alpha_j y_{x_j}$ ■

Generating $\mathcal{E}_\psi$. The linearization representation-procedure constructs the set of conjunctions $\mathcal{E}_\psi$ as follows. For each triple (x, y, t) where $x, t \in \mathcal{F}$ and $y \in \mathcal{F}^{d(F)}$, $\mathcal{E}_\psi$ will have the conjunction of ψ', the curve-verifier equation $\chi[x]$, and the linearization-verifier equation $\chi[y, t]$.

Correctness of the algorithm.

The domain F that is generated by the linearization representation-procedure has the required parameters, and it is easy to verify that the running time is polynomial in $|F|$. To complete the proof of Lemma 5.7 let us show that $\mathcal{E}_\psi$ has the extension and restriction properties, as the other required properties are obvious.

- Extension: Let $\mathcal{A}$ be a good assignment for the variables of Ψ that satisfies ψ. Let g be the $[s, d]$-LDF assigned to E, and extend $\mathcal{A}$ to F by assigning L_g to it. We need to show that the extended $\mathcal{A}$ satisfies the conjunctions of $\mathcal{E}_\psi$. According to the construction, it is enough to show that ψ' is satisfied and that the curve-verifier and linearization-verifier equations are satisfied as well.

Since $\mathcal{A}(F) = L_g$ is a linear-homogeneous LDF, the linearization-verifier equations are satisfied by Proposition 5.23. g is an s-degree LDF, hence by Claim 5.25 $L_g \circ \widehat{\gamma} = g \circ \Gamma$, which implies that the curve-verifier equations are all satisfied by Proposition 5.26. The only difference between ψ and ψ' is the replacement of $\psi_\star$ with $F[y_*]$. But by Claim 5.27 the evaluations of $\psi_\star$ and $F[y_*]$ are equal (the conditions of the claim hold by the above discussion), and therefore since ψ is satisfied, ψ' is satisfied as well.

- Restriction: Let $\mathcal{A}$ be a feasible assignment for the variables of Ψ and for F. We assume that these assignments satisfy at least an $|\mathcal{F}|^{-1/2}$ fraction of the conjunctions in $\mathcal{E}_\psi$. This implies that ψ' is satisfied, and that at least an $|\mathcal{F}|^{-1/2}$ fraction of the curve-verifier equations are satisfied, as well as an $|\mathcal{F}|^{-1/2}$ fraction of the linearization-verifier equations. Let us prove that ψ is satisfied.

  Since at least an $|\mathcal{F}|^{-1/2}$ fraction of the linearization-verifier are satisfied, we gather from Proposition 5.23 that F is assigned a linear-homogeneous LDF. Proposition 5.26 then implies that $\mathcal{A}(F) \circ \widehat{\gamma} = \mathcal{A}(E) \circ \Gamma$. The conditions of Claim 5.27 are thus satisfied, hence $\psi_\star$ has the same evaluation as $F[y_*]$. Since ψ' is satisfied, this implies that ψ is satisfied as well.

Part II

Testing Juntas

Chapter 6

Introduction to Part II

A property P is said to be *ϵ-testable using q queries*, or simply (ϵ, q)-testable, if there exists a probabilistic algorithm that makes at most q queries on any given input f (it is assumed that the input is accessed using an oracle), such that

- if f satisfies P, then the algorithm accepts it with probability at least 2/3, and
- if f is ϵ-far from P, that is, if it must be changed in more than an ϵ-fraction of the places in order to make it satisfy P, then the algorithm rejects it with probability at least 2/3.

A testing algorithm is said to be 1-*sided* if it accepts with probability 1 any input that satisfies P. A testing algorithm that determines all its queries in advance, and uses the answers only in deciding whether to accept the input (and not in planning some of the queries) is called a *non-adaptive test*.

Boolean functions, which are the focus of this part of the thesis, were given particular consideration from the point of view of property testing, and especially properties related to monotonicity [GGL$^+$00, DGL$^+$99, FLN$^+$02]. Perhaps the work most closely related to the results in this part of the thesis is [PRS01]. It presents testing algorithms that perform $O(1/\epsilon)$ queries for the following properties of boolean functions: Being a singleton function (a function of a single variable), being a J-monomial (a conjunction of at most J literals), and being a monotone DNF function with a bounded number of terms.

Boolean functions and juntas

In this part of the thesis we consider properties of boolean functions over n variables, namely functions over n variables that admit only two values. It will be convenient for us to assume that the values of boolean functions range in $\{-1, 1\}$.

While some of our results consider functions over boolean variables, other results apply to functions over variables that range in general domains. When the boolean function f

being discussed is known, we denote the range of the i'th variable of f by Ω_i (in the case of boolean variables, $\Omega_i = \{0,1\}$). Denoting

$$\mathcal{P}([n]) \doteq \prod_{i=1}^{n} \Omega_i \ ,$$

we have that all boolean functions dealt with herein can be written in the form $\mathsf{f} : \mathcal{P}([n]) \to \{-1,1\}$, and that any input x for such a function is a vector $(x_1, \ldots, x_n)$, where $x_i \in \Omega_i$ for every i.

Juntas. The main property of boolean functions we focus on, is that of depending on only J (or less) of its variables.

Definition 6.1 (juntas, dominating sets). *A boolean function* $\mathsf{f} : \mathcal{P}([n]) \to \{-1,1\}$ *is called a* J-junta *if there exists a set* $\mathcal{J} \subseteq [n]$ *of size at most* J*, such that* $\mathsf{f}(x) = \mathsf{f}(y)$ *for every two inputs* $x, y \in \mathcal{P}([n])$ *that agree on* $\mathcal{J}$*, namely that satisfy* $x_i = y_i$ *for all* $i \in \mathcal{J}$*. In this case it is said that* f *is* dominated *by* $\mathcal{J}$*. Somewhat abusing notation,* $\mathcal{J}$ *is also referred to as* the junta which dominates f*.*

Preview of Results

Knowing that a function depends on only a small number of variables can be especially useful in the context of learning. For various functions classes there exist algorithms that are attribute efficient (*cf.* [Lit87, BHL95, BL96, UTW97]). That is, they have polynomial dependence on the number of relevant variables of the function being learned but only logarithmic dependence on the total number of variables. One should also mention here the work of [MOS02] concerning computationally efficient learning of such functions when the algorithm is restricted to uniform samples.

As part of this effort, [DT99] presented an algorithm that for any input function f over boolean variables, uses $O(J(\log(J+1)/\epsilon + \log n))$ queries to completely determine a J-junta that dominates a function f' which is ϵ-close to f, if such a J-junta exists. In particular, their algorithm can be used to test for the property of being a J-junta. We show here the existence of a test for being a J-junta, for functions over arbitrary product spaces, whose number of queries does not depend on n at all.

Theorem 6.2 (the main result). *For every fixed* J *the property of being a* J*-junta is* $(\epsilon, \mathrm{poly}(J)/\epsilon))$*-testable for any given* ϵ*.*

Almost juntas

Let us review the definition of testable properties, with respect to the property of being a J-junta. To prove that this property is ϵ-testable, a test is to be shown, that distinguishes between J-juntas, and functions that must be changed in more than an 'ϵ-fraction' of the places in order to become J-juntas. This is made more formal and somewhat more general using the following definition, of a function that is ϵ-close to being a junta. Instead of just counting the number of values of f that need to be changed in order to make it a J-junta, giving the same weight to the value at every input, we allow weighing the inputs using a product probability-measure.

Definition 6.3 (($\boldsymbol{\epsilon, J}$)-juntas). *Let* $\mathsf{f} : \mathcal{P}([n]) \to \{-1, 1\}$ *be a boolean function, and assume that the range* Ω_i *of each variable of* f *is equipped with a probability measure* μ_i. *This determines a probability measure* $\mu^{[n]} = \prod_{i=1}^{n} \mu_i$ *over* $\mathcal{P}([n])$.

f *is said to be a,* (ϵ, J)-junta *if there exists a boolean* J*-junta* $\mathsf{g} : \mathcal{P}([n]) \to \{-1, 1\}$ *such that for a random input* $x \in \mathcal{P}([n])$ *(chosen according to* $\mu^{[n]}$*),*

$$\Pr\left[\mathsf{f}(x) \neq \mathsf{g}(x)\right] \geq 1 - \epsilon$$

In terms of the above definition, an (ϵ, q)-test for the property of being a J-junta is given as input a function f, and a product measure $\mu^{[n]}$ on its domain, and uses q queries to distinguish between the case where the input function is a J-junta, and the case where it is not an (ϵ, J)-junta. We require that the number of queries made be entirely independent of $\mu^{[n]}$.

Note that the above definition includes the standard case where f is defined over boolean variables – one should just take $\Omega_i = \{0, 1\}$ for every i, and μ_i to be the uniform measure over Ω_i. However, by supplying a biased measure μ_i for every i, a J-junta test can distinguish, using the same number of queries, between the case where a given f is a junta, and the case where it must be changed on a set of $\mu^{[n]}$-measure ϵ in order to become junta. Such a test can also be used to test functions that range over non-boolean variables.

Junta tests

In order to establish Theorem 6.2 we describe three testing algorithms. The first algorithm is non-adaptive, requires $O(J^4 \ln(J+1)/\epsilon)$ queries, and is stronger in that it is 1-sided. We also provide an adaptive variant of this algorithm which requires only $O(J^3 \ln^2(J+1)/\epsilon)$ queries. The third algorithm presented here, is a non-adaptive variant of the first algorithm that has a 2-sided error, but requires just $O(J^2 \ln^2(J+1)/\epsilon)$ queries.

Lower bound

On the other hand, at least with regards to non-adaptive algorithms, we show that the query complexity has to be a power of J (the tilde notation in the following is used to hide polylogarithmic factors), even if the test is restricted to functions over boolean variables with respect to the uniform measure.

Theorem 6.4. *For every $\alpha > 0$, a non-adaptive $\left(\frac{1}{2} - \alpha, q\right)$-test for the property of being a J-junta requires at least $q \geq \tilde{\Omega}\left(\sqrt{J}\right)$ queries, even if restricted to functions over boolean variables and equipped with the uniform measure over their domain.*

Recently, Chockler and Gutfreund [CG02] have proven a better lower bound, which holds for adaptive testing algorithms as well. However, the proof given here may have significance beyond the lower bound itself, since during its course we prove a result about random walks on the group $\mathbb{Z}_2^q$ that may be of an independent interest.

Random walks

Given any (finite) group G and a distribution P on G, a *random walk on G with step distribution P* starts with the identity element, and at each step t, denoting its current position by X_t, picks a random element ξ_t of G according to P and goes to $X_{t+1} = \xi_t X_t$. This definition of a random walk generalizes the more familiar notion of a random walk on a Cayley graph of a group, which is obtained by setting P to be a uniform distribution on the elements of a generating set for G.

A fundamental result of Markov [Mar06] from 1906 (see also [AD86]) states that this random walk converges to the uniform distribution on G, unless P is concentrated on a coset. A more recent question of interest is to estimate the rate of convergence of the random walk to its limit distribution. It is easy to see that this rate depends on the step distribution P, and therefore all the results in this direction concentrate on particular families of distributions for which good bounds can be obtained.

Here we ask a different question: Given a distance parameter $\delta > 0$, we ask when do the distributions of X_t and X_{t+c} (for an appropriate constant c), become δ-close to each other. Here we give a bound for the group $\mathbb{Z}_2^q$ (and $c = 2$), that does not depend on the step distribution P.

Theorem 6.5. *Let P be a distribution on $\mathbb{Z}_2^q$, and let X be the random walk on $\mathbb{Z}_2^q$ with step distribution P. Let P_t be the distribution of X at step t. There is an absolute constant C, such that for every $\delta > 0$, if $t \geq C \cdot \frac{\log \frac{1}{\delta}}{\delta} \cdot q^2 \log^2(q+1)$ then $\|P_t - P_{t+2}\| \leq \delta$.*

Testing whether f is a permutation of h

Finally, we consider the question of testing that a function f is identical to a fixed function h up to a permutation of its variables. We only consider functions over boolean variables here, whose domains are equipped with the uniform measure. Similar questions were given consideration already in [PRS01]. Here we construct a test for any function h which is a J-junta that is given in advance.

Some notation about restrictions and permutations of vectors is needed for the exact formulation of this result: Suppose that $\mathcal{J} = \{j_1, \ldots, j_J\}$ is some subset of $[n]$, whose elements are given in ascending order, $j_1 < \cdots < j_J$. For any permutation $\sigma : [J] \rightarrow [J]$ and every vector $x = (x_1, \ldots, x_n) \in \{0,1\}^n$, we denote by $x|_{\sigma(\mathcal{J})}$ the vector $x = (x_{j_{\sigma(1)}}, \ldots, x_{j_{\sigma(J)}}) \in \{0,1\}^J$.

Theorem 6.6. *Let* $\mathsf{g} : \{0,1\}^J \rightarrow \{-1,1\}$ *be a function. The property, that* $\mathsf{f}(x) = \mathsf{g}(x|_{\sigma(\mathcal{J})})$ *for some* $\mathcal{J} \subset [n]$ *of size* J *and some permutation* $\sigma : [J] \rightarrow [J]$*, is* $(\epsilon, \mathrm{poly}(\epsilon, J))$*-testable for every* ϵ*.*

Organization of this Part

We start with Chapter 7, where we give some preliminaries and notation required for the subsequent chapter, and introduce the notion of the variation of a function f on a set I of coordinates.

Chapter 8 presents our first junta test, called the size test. It randomly partitions the coordinates of a given function f, and applies a simple test to each subset in the partition, to discover whether f depends on any of its coordinates. The size test is non-adaptive, and it has 1-sided error. In Chapter 9, we present two variants of the size test, which achieve better query complexity. One of these variants has a 1-sided error but is adaptive, and the other is non-adaptive but has a 2-sided error.

We then provide the lower bound for non-adaptive junta testing in Chapter 10, deriving it from the result concerning random walks in $\mathbb{Z}_2^q$ that is also proven there. Finally, in Chapter 11 we show how to test a function f for the property of being identical to a permutation of a given function h.

Chapter 7

Preliminaries

First, let us deal with a notational issue that will simplify the following exposition.

Partial inputs. Suppose that $\mathsf{f} : \mathcal{P}([n]) \to \{-1, 1\}$ is a boolean function, where $\mathcal{P}([n]) = \prod_{i=1}^{n} \Omega_i$, and each set Ω_i is equipped with a probability measure μ_i. Each element $x \in \mathcal{P}([n])$ is thus an assignment to the variables of f, where the i'th coordinate of x determines the value of the i'th variable. To easily specify assignments for only some of the variables of f, we define for each set $I \subseteq [n]$ of coordinates,

$$\mathcal{P}(I) \doteq \prod_{i \in I} \Omega_i$$

and equip it with the probability measure $\mu^I \doteq \prod_{i \in I} \mu_i$. An element $w \in \mathcal{P}(I)$ is thus a partial assignment for the variables of f. Whenever an element $w \in \mathcal{P}(I)$ is chosen randomly, it is chosen with respect to μ^I unless stated otherwise.

Input manipulation. If $w \in \mathcal{P}(I)$ and $z \in \mathcal{P}(H)$ are two partial inputs, and I and H are disjoint, let $w \sqcup z \in \mathcal{P}(I \cup H)$ denote the partial input whose i'th coordinate is w_i if $i \in I$, and z_i, if $i \in H$. For a set $I \subseteq [n]$ of coordinates and an input $x \in \mathcal{P}([n])$, it is possible to obtain a partial input by restricting x to the coordinates of I, obtaining $x|_I \in \mathcal{P}(I)$. For simplicity we somewhat abuse notation, writing $x \cap I$ instead of $x|_I$. Similarly, we let $x \setminus I \in \mathcal{P}([n] \setminus I)$ denote the partial assignment obtained from x by taking the coordinates from $[n] \setminus I$.

Variation

We now turn to define a measure of dependency of a boolean function f on a given set of coordinates (variables). The *variation* of f on a set I is proportional to the probability that f does not yield the same values, given two random inputs that differ only on coordinates from I.

Definition 7.1 (variation). *Let* $\mathsf{f} : \mathcal{P}([n]) \to \{-1,1\}$ *be a boolean function, and fix a set* $I \subseteq [n]$ *of coordinates. Let* $w \in \mathcal{P}([n] \setminus I)$ *and let* $z_1, z_2 \in \mathcal{P}(I)$ *be chosen independently at random. Then the variation of* f *on* I *is defined by*

$$\mathsf{Vr}_{\mathsf{f}}(I) \doteq 2\Pr[\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)]$$

The variation is monotone and sub-additive, as stated in the next proposition.

Proposition 7.2 (monotonicity and sub-additivity). *Let* $\mathsf{f} : \mathcal{P}([n]) \to \{-1,1\}$ *be a boolean function, and let* A *and* B *be subsets of* $[n]$*. Then*

$$\mathsf{Vr}_{\mathsf{f}}(A) + \mathsf{Vr}_{\mathsf{f}}(B) \geq \mathsf{Vr}_{\mathsf{f}}(A \cup B) \geq \mathsf{Vr}_{\mathsf{f}}(B)$$

Proof. The proof of Proposition 7.2 is elementary, however somewhat tedious. Note that for the case where $\mathcal{P}([n])$ is the discrete cube equipped with the uniform measure, the proposition follows directly from the Fourier-analytic formula for the variation, given in Proposition 7.4.

We begin by proving the monotonicity property. Let $A, B \subseteq [n]$, and suppose without loss of generality that A and B are disjoint. Now let w be a random element of $\mathcal{P}([n] \setminus (A \cup B))$, let u_1, u_2 be random elements in $\mathcal{P}(A)$, and let v_1, v_2 be random elements in $\mathcal{P}(B)$, and assume that w, u_1, u_2, v_1, v_2 are all independent. Then according to Definition 7.1, we have

$$\begin{aligned}\mathsf{Vr}_{\mathsf{f}}(A \cup B) &= 2\Pr\left[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_2 \sqcup v_2)\right] = \\ &= 2\mathbb{E}_w\left[\Pr\left[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_2 \sqcup v_2)\right]\right]\end{aligned}$$

And similarly,

$$\mathsf{Vr}_{\mathsf{f}}(B) = 2\mathbb{E}_w\left[\Pr\left[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_1 \sqcup v_2)\right]\right]$$

Let us fix an arbitrary $w \in \mathcal{P}([n] \setminus (A \cup B))$, leaving u_1, u_2, v_1, v_2 random, and conclude by showing that

$$\Pr\left[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_2 \sqcup v_2)\right] \geq \Pr\left[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_1 \sqcup v_2)\right] \qquad (*)$$

For every $u \in \mathcal{P}(A)$, define

$$\alpha_u \doteq \Pr\left[\mathsf{f}(w \sqcup u \sqcup v_1) = 1\right]$$

Then

$$\begin{aligned}\Pr\left[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_2 \sqcup v_2)\right] &= \mathbb{E}_{u_1,u_2}[\alpha_{u_1}(1-\alpha_{u_2}) + \alpha_{u_2}(1-\alpha_{u_1})] = \\ &= \mathbb{E}_{u_1,u_2}[\alpha_{u_1} + \alpha_{u_2} - 2\alpha_{u_1}\alpha_{u_2}] =\end{aligned}$$

(α_{u_1} and α_{u_2} are functions of independent variables, and therefore are independent)

$$\begin{aligned}
&= 2\mathbb{E}_{u_1}[\alpha_{u_1}] - 2\big(\mathbb{E}_{u_1}[\alpha_{u_1}]\big)^2 \geq \\
&\geq 2\mathbb{E}_{u_1}[\alpha_{u_1}] - 2\mathbb{E}_{u_1}\left[\alpha_{u_1}^2\right] = \\
&= 2\mathbb{E}_{u_1}[\alpha_{u_1}(1-\alpha_{u_1})] = \\
&= \Pr\left[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_1 \sqcup v_2)\right]
\end{aligned}$$

We have (*), and therefore we have completed the proof of the monotonicity property.

Let us now continue with the sub-additivity of the variation. Let $A, B \subseteq [n]$. From the monotonicity property it follows that we may assume that A and B are disjoint. Let w, u_1, u_2, v_1, v_2 be randomly chosen as above. Then

$$\begin{aligned}
\mathsf{Vr_f}(A \cup B) &= 2\Pr[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_2 \sqcup v_2)] \leq \\
&\leq 2\Pr[\mathsf{f}(w \sqcup u_1 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_2 \sqcup v_1)] + 2\Pr[\mathsf{f}(w \sqcup u_2 \sqcup v_1) \neq \mathsf{f}(w \sqcup u_2 \sqcup v_2)] = \\
&= \mathsf{Vr_f}(A) + \mathsf{Vr_f}(B)
\end{aligned}$$

■

Norms, Distances, and Inner-Products

Although our main concern here is the set of boolean functions over $\mathcal{P}([n])$, it is useful to consider such functions as elements in the space of real-valued functions $\mathsf{f} : \mathcal{P}([n]) \to \mathbb{R}$. For such a function f, and any parameter $1 \leq q < \infty$, the *normalized ℓ_q-norm* of f is defined by

$$\|\mathsf{f}\|_q \doteq \mathop{\mathbb{E}}_{x \in \mathcal{P}([n])} [|\mathsf{f}(x)|^q]^{1/q}$$

(x is randomly chosen in $\mathcal{P}([n])$ according to $\mu^{[n]}$). The *inner-product* between two functions $\mathsf{f}, \mathsf{g} : \mathcal{P}([n]) \to \mathbb{R}$, is defined by

$$\langle \mathsf{f}, \mathsf{g} \rangle \doteq \mathop{\mathbb{E}}_{x \in \mathcal{P}([n])} [\mathsf{f}(x)\mathsf{g}(x)]$$

This inner-product is related to the ℓ_2 norm, satisfying $\langle \mathsf{f}, \mathsf{f} \rangle = \|\mathsf{f}\|_2^2$ for every real-valued function f.

We also define another norm, that is used in Chapter 10 to measure the distance between two probability measures $P, Q : \{0,1\}^n \to \mathbb{R}$ over the discrete cube. The *variation distance* between two such measure is defined by $|P - Q| \doteq \frac{1}{2} \sum_{x \in \{0,1\}^n} |P(x) - Q(x)|$ (this is not related to the variation discussed above).

Harmonic Analysis

Let us now focus on functions defined over the discrete cube $\{0,1\}^n$, equipped with the uniform measure. Real-valued functions defined over this domain can be expressed by their Fourier expansion as follows.

Definition 7.3 (characters and weights). *Let $S \subseteq [n]$. The* character χ_S *is the function over $\{0,1\}^n$ defined by $\chi_S(x) \doteq \prod_{i \in S} (-1)^{x_i}$.*

Given a function $\mathsf{f} : \{0,1\}^n \to \mathbb{R}$, its expansion as a linear combination of characters

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

is called the Fourier expansion *of* f *(such an expansion always exists, since the set of characters forms a linear basis for the set of real functions over $\{0,1\}^n$).*

Properties of characters. The set of all characters forms an orthonormal basis for the space of real-valued functions over $\{0,1\}^n$, with respect to the inner-product defined above. In addition, every character χ_S satisfies $\chi_S(x \oplus y) = \chi_S(x)\chi_S(y)$ for any points $x, y \in \{0,1\}^n$, where '$x \oplus y$' denotes the coordinate-wise addition of x and y in $\mathbb{Z}_2^n$.

Variation and fourier expansion. The variation of a boolean function f, defined over the discrete cube, can be written in terms of its Fourier expansion as follows.

Proposition 7.4. *Let $\mathsf{f} : \{0,1\}^n \to \{-1,1\}$ be a boolean function, where $\{0,1\}^n$ is equipped with the uniform measure, and let $I \subseteq [n]$ be a set of coordinates. Then*

$$\mathsf{Vr}_{\mathsf{f}}(I) = \sum_{S \cap I \neq \emptyset} \widehat{\mathsf{f}}^2(S)$$

Note that Proposition 7.4 directly implies Proposition 7.2 above for functions over the discrete cube (with the uniform measure).

Proof of Proposition 7.4: Let w, z_1, z_2 be chosen randomly as in Definition 7.1. Then

$$\begin{aligned}
1 - \mathsf{Vr}_{\mathsf{f}}(I) &= \Pr[\mathsf{f}(w \sqcup z_1) = \mathsf{f}(w \sqcup z_2)] - \Pr[\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)] = \\
&= \mathbb{E}_{w,z_1,z_2}[\mathsf{f}(w \sqcup z_1)\mathsf{f}(w \sqcup z_2)] = \\
&= \mathbb{E}_{w,z_1,z_2}\left[\left(\sum_S \widehat{\mathsf{f}}(S)\chi_S(w \sqcup z_1)\right)\left(\sum_T \widehat{\mathsf{f}}(T)\chi_T(w \sqcup z_2)\right)\right] = \\
&= \sum_{S,T} \widehat{\mathsf{f}}(S)\widehat{\mathsf{f}}(T)\mathbb{E}_{w,z_1,z_2}[\chi_S(w \sqcup z_1)\chi_T(w \sqcup z_2)]
\end{aligned}$$

It is straightforward to verify that unless $S = T$ and $S \cap I = T \cap I = \emptyset$,

$$\mathbb{E}_{w,z_1,z_2}[\chi_S(w \sqcup z_1)\chi_T(w \sqcup z_2)] = 0$$

and that otherwise the above expectaion equals 1. Hence we have

$$1 - \mathsf{Vr}_{\mathsf{f}}(I) = \sum_{S \cap I = \emptyset} \widehat{\mathsf{f}}(S)^2$$

Since f is boolean, and since the set of characters forms an orthonormal basis, we have $\sum_{S \subseteq [n]} \widehat{\mathsf{f}}(S)^2 = \|\mathsf{f}\|_2^2 = 1$. Putting that into the above equality yields

$$\mathsf{Vr}_{\mathsf{f}}(I) = \sum_{S \cap I \neq \emptyset} \widehat{\mathsf{f}}^2(S)$$

as required. ■

Convolution.

The *convolution* of two distributions or functions $\mathsf{f}, \mathsf{g} : \{0,1\}^n \to \mathbb{R}$ is denoted by $\mathsf{f} * \mathsf{g}$, and is defined by $(\mathsf{f} * \mathsf{g})(y) \doteq \sum_x \big(\mathsf{f}(x) \cdot \mathsf{g}(x \oplus y)\big)$. We will need the following important property of convolution:

$$\widehat{(\mathsf{f} * \mathsf{g})}(S) = 2^n \cdot \widehat{\mathsf{f}}(S) \cdot \widehat{\mathsf{g}}(S).$$

Chapter 8

The Size Test

The size test, shown herein, is a one-sided non-adaptive $(\epsilon, \Theta(J^4 \ln(J+1)/\epsilon))$-test for the property of being a J-junta. The independence test, presented next, is its main component. Given a set I of coordinates, the independence test is used to determine whether a given boolean function f is independent of the coordinates in I. It is a simple two-query test as follows.

The independence test. Choose a random $w \in \mathcal{P}([n] \setminus I)$, and choose $z_1, z_2 \in \mathcal{P}(I)$ randomly and independently. Verify that $\mathsf{f}(w \sqcup z_1) = \mathsf{f}(w \sqcup z_2)$.

Properties of the independence test. It is obvious that the independence test always accepts if f is independent of the coordinates in I, and by Definition 7.1, its rejection probability equals $\frac{1}{2}\mathsf{Vr}_{\mathsf{f}}(I)$.

If f is a J junta, then it clearly has the following property: for every partition $I_1, \ldots, I_r$ of the set of coordinates, all but at most J of them have zero variation. Hence when the independence test is applied to f with respect to all but at most J of the subsets, it must accept. This consideration motivates the following size test.

The size test. The test has two parameters, r and h, that are to be chosen later. The test first chooses a random partition $I_1, \ldots, I_r$ of the set $[n]$ of coordinates. It then identifies on which of the I_j's f has non-negligible variation, using $2rh$ queries, by going over every j from 1 to r and applying h iterations of the independence test to I_j. If f is found to be dependent on more than J subsets, the test rejects, and otherwise it accepts.

Properties of the test. The size test obviously accepts every J-junta, thus having perfect completeness. We show in the next section that, for a proper setting of the parameters r and h, the size test rejects with probability at least $1/2$ unless f is an (ϵ, J)-junta (since the test is 1-sided this can easily be amplified to 2/3). Before we prove this, let us set the r and h parameters.

The parameters of the test. Let us set $r \doteq 16J^2$ and $h \doteq 4er(\ln(J+1)+2)/\epsilon = \Theta(J^2 \ln(J+1)/\epsilon)$. Hence overall the test makes $2rh = \Theta(J^4 \ln(J+1)/\epsilon)$ queries to f, as required.

8.1 Soundness of the Size Test

Assume that f passes the test with probability 1/2. We prove that f must be an (ϵ, J)-junta in two steps. We first take $\mathcal{J}$ to be the set of coordinates on which f has variation larger than some threshold t, and prove that $|\mathcal{J}| \leq J$. Then we show that the total variation of f on coordinates outside $\mathcal{J}$ is bounded by 2ϵ. This implies, by a simple argument, that f is ϵ-close to a junta dominated by $\mathcal{J}$.

Let $t \doteq \frac{2(\ln(J+1)+2)}{h} = \frac{\epsilon}{2er}$, and let $\mathcal{J}$ denote the set of all coordinates i for which $\mathsf{Vr}_{\mathsf{f}}(\{i\}) > t$. We also denote $\bar{\mathcal{J}} \doteq [n] \setminus \mathcal{J}$.

Proposition 8.1. *If the size test succeeds on* f *with probability* $1/2$*, then* $|\mathcal{J}| \leq J$.

Proof. The key observation here is that if a set I of coordinates contains a member of $\mathcal{J}$, then the variation of f on that set is at least t (by Proposition 7.2), and therefore each iteration of the independence test on I detects the dependence with probability at least $t/2$.

Suppose, for the sake of contradiction, that $|\mathcal{J}| > J$. Since $r = 16J^2$, it is easy to verify that with probability at least 3/4 the number of subsets in the partition $I_1, \ldots, I_r$ that contain an element from $\mathcal{J}$ is at least $J+1$. When this occurs, the probability that any of the first $J+1$ subsets which intersect $\mathcal{J}$ will *not* be identified by the size test is bounded by $(J+1)(1-t/2)^h \leq (J+1)e^{-\ln(J+1)-2} < 1/4$, since $h = 2(2+\ln(J+1))/t$. Overall we have that with probability at least 1/2 the size test rejects. ■

Having shown that $|\mathcal{J}| \leq J$, the proof of soundness will be complete by showing that f is ϵ-close to a junta dominated by $\mathcal{J}$. We actually show that $\mathsf{Vr}_{\mathsf{f}}(\bar{\mathcal{J}}) < 2\epsilon$. This is sufficient to complete the proof, according to the following claim.

Proposition 8.2. *Let* $\mathcal{J}$ *be a set of coordinates satisfying* $\mathsf{Vr}_{\mathsf{f}}(\bar{\mathcal{J}}) < 2\epsilon$*. Then there exists a boolean function* h*, that depends only on coordinates from* $\mathcal{J}$*, and agrees with* f *on a set of inputs of measure at least* $(1-\epsilon)$.

Proof. Let z be a random element in $\mathcal{P}(I)$, and define $\mathsf{h} : \mathcal{P}([n]) \to \{-1, 1\}$ by

$$\mathsf{h}(x) = \begin{cases} 1 & \mathbb{E}_z[\mathsf{f}((x \setminus I) \sqcup z)] \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

$\mathsf{h}(x)$ is the majority of the values of f over all inputs that agree with x on the coordinates in $\mathcal{J}$. It is easy to verify that h depends only on the coordinates in $\mathcal{J}$, so to prove the claim, we show that f and h agree on at least a $(1-\epsilon)$-fraction of the inputs.

Fix $w \in \mathcal{P}([n] \setminus I)$, let $z_1, z_2 \in \mathcal{P}(I)$ be chosen randomly and independently, and denote $p(w) \doteq \Pr_{z_1}[\mathsf{f}(w \sqcup z_1) \neq \mathsf{h}(w \sqcup z_1)]$. Then

$$\Pr[\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)] = 2p(w)(1-p(w))$$

and $p(w) \leq 1/2$. Now if x is chosen randomly from $\mathcal{P}([n])$ then $w \doteq (x \setminus I)$ is a random element in $\mathcal{P}([n] \setminus I)$, and we thus have

$$\begin{aligned}\Pr_x[\mathsf{f}(x) \neq \mathsf{h}(x)] &= \Pr_{w,z_1}[\mathsf{f}(w \sqcup z_1) \neq \mathsf{h}(w \sqcup z_1)] = \mathbb{E}_w[p(w)] \leq \\ &\leq \mathbb{E}_w[2p(w)(1-p(w))] = \mathbb{E}_w[\Pr_{z_1,z_2}[\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)]\,] = \\ &= \Pr_{w,z_1,z_2}[\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)] = \frac{1}{2}\mathsf{Vr_f}(\bar{\mathcal{J}}) < \epsilon\end{aligned}$$

■

Bounding $\mathsf{Vr_f}(\bar{\mathcal{J}})$

It is left to show that $\mathsf{Vr_f}(\bar{\mathcal{J}}) < 2\epsilon$. Assume otherwise, and let us prove that the test rejects with probability at least 1/2.

Idea of the proof. The sum $\sum_{j=1}^{r} \mathsf{Vr_f}(I_j \setminus \mathcal{J})$ is never less than $\mathsf{Vr_f}(\bar{\mathcal{J}})$, as follows from the sub-additivity of the variation (see Proposition 7.2). Since we assume that $\mathsf{Vr_f}(\bar{\mathcal{J}}) \geq 2\epsilon$, we have

$$\sum_{j=1}^{r} \mathbb{E}[\mathsf{Vr_f}(I_j \setminus \mathcal{J})] = \mathbb{E}\left[\sum_{j=1}^{r} \mathsf{Vr_f}(I_j \setminus \mathcal{J})\right] \geq 2\epsilon$$

Using the fact that the sets in the partition are equidistributed, it follows that for any fixed j,

$$\mathbb{E}[\mathsf{Vr_f}(I_j \setminus \mathcal{J})] \geq 2\epsilon/r$$

Since I_j is a random set of coordinates, then using the fact that every coordinate can contribute at most t to the variation of $I_j \setminus \mathcal{J}$, we obtain a concentration property for its variation. In fact, we show that $\mathsf{Vr_f}(I_j \setminus \mathcal{J})$ (and therefore $\mathsf{Vr_f}(I_j)$) is with high probability at least a sizable portion of the bound for its expectation. This implies that with high probability, there are many sets I_j in the partition whose variation is relatively high. Since such sets are detected with high probability by the independence test, the size test rejects f with high probability.

Definition 8.3. *A set I_j in the partition is said to be* detectable *if* $\mathsf{Vr_f}(I_j) > \frac{\epsilon}{er}$.

Lemma 8.4. *Fix j, $1 \leq j \leq r$. The probability that I_j is detectable, over the choice of the partition $I_1, \ldots, I_r$, is at least* $3/4$.

Before we prove Lemma 8.4, we show how it completes the proof of the soundness of the size test. Let q denote the probability that the number of detectable subsets in the partition is smaller than $r/4$. Since the number of detectable subsets is bounded by r, Lemma 8.4 implies that

$$\frac{1}{4}rq + r(1-q) \geq \mathbb{E}\left[\,\text{number of detectable } I_j\text{'s}\,\right] \geq \frac{3}{4}r$$

from which we have $q \leq 1/3$. Hence with probability at least $2/3$, there are at least $r/4 = 4J^2 > J+1$ subsets in the partition, whose variation is larger than $\epsilon/er = 2t$. The size test fails in this case with probability at least $15/16$, as follows from an argument similar to that in the proof of Proposition 8.1. Therefore, the size test rejects f with an overall probability at least $1/2$, as required.

It is only left to prove Lemma 8.4.

Proof of Lemma 8.4: As mentioned above, the expectation of the variation of f on $I_j \setminus \mathcal{J}$ is at least $2\epsilon/r$. Lemma 8.4 will follow by showing that with probability at least $3/4$, $\mathsf{Vr}_{\mathsf{f}}(I_j \setminus \mathcal{J}) \geq 2\epsilon/r$.

I_j is a random subset, obtained by going over the coordinates $i \in [n]$ and taking each into I_j with probability $1/r$. We can thus view the random variable $\mathsf{Vr}_{\mathsf{f}}(I_j \setminus \mathcal{J})$ as a sum of the gradual donation of each coordinate,

$$\mathsf{Vr}_{\mathsf{f}}(I_j \setminus \mathcal{J}) = \sum_{i=1}^{n} \Big(\mathsf{Vr}_{\mathsf{f}}\big([i] \cap (I_j \setminus \mathcal{J})\big) - \mathsf{Vr}_{\mathsf{f}}\big([i-1] \cap (I_j \setminus \mathcal{J})\big) \Big)$$

In order to use standard deviation bounds for $\mathsf{Vr}_{\mathsf{f}}(I_j \setminus \mathcal{J})$, we would like the summands on the right-hand side to be independent and bounded by a small number. Note that the i'th summand is zero if $i \in \mathcal{J}$, and if $i \notin \mathcal{J}$ it is bounded by t, as follows from the sub-additivity of the variation (and of course, all the summands are non-negative). The summands are thus indeed bounded by a small number, but they are *not* independent. This is tackled by introducing a technical tool that we call the unique-variation. While related to the variation, the unique-variation of I_j can be written as the sum of independent non-negative bounded random variables.

Definition 8.5 (unique-variation). *Define the unique-variation of every coordinate $i \in [n]$ by*

$$\mathsf{Ur}_{\mathsf{f}}(i) \doteq \mathsf{Vr}_{\mathsf{f}}([i] \setminus \mathcal{J}) - \mathsf{Vr}_{\mathsf{f}}([i-1] \setminus \mathcal{J}),$$

where $[0]$ *denotes the empty set. Now for every set $I \subseteq \mathcal{P}([n]))$ define its unique-variation by*

$$\mathsf{Ur}_{\mathsf{f}}(I) \doteq \sum_{i \in I} \mathsf{Ur}_{\mathsf{f}}(i)$$

The following lemma, the proof of which is deferred to Section 8.2, shows that the unique-variation of a subset I bounds the variation of I from below (note that in the case of the discrete cube with the uniform measure, the lemma follows directly from Proposition 7.2).

Lemma 8.6. *For every set $I \subseteq [n]$ of coordinates, $\mathsf{Ur_f}(I) \leq \mathsf{Vr_f}(I \setminus \mathcal{J})$.*

Therefore, it suffices to show that $\Pr\left[\mathsf{Ur_f}(I_j) \leq \epsilon/er\right] < 1/4$ in order to complete the proof of Lemma 8.4.

Note that the unique-variation of coordinates in $\mathcal{J}$ is zero, and that $\mathsf{Ur_f}(i) \leq \mathsf{Vr_f}(i) \leq t$ for coordinates i outside $\mathcal{J}$, as follows from the sub-additivity property of the variation. The unique-variation of I_j is therefore a sum of independent non-negative random variables, each of which is bounded by t, and its expectation is given by

$$\mathbb{E}\left[\mathsf{Ur_f}(I_j)\right] = \frac{1}{r}\sum_{i\in[n]} \mathsf{Ur_f}(i) = \mathsf{Vr_f}(\bar{\mathcal{J}})/r \geq 2\epsilon/r$$

We can therefore apply standard deviation bounds to it, such as the following Chernoff-like bound, proven in Appendix A.

Proposition 8.7. *Let $X = \sum_{i=1}^{l} X_i$ be a sum of non-negative independent random variables X_i, and denote the expectation of X by α. If each X_i is bounded above by t, then*

$$\Pr[X < \eta\alpha] < \exp\left(\frac{\alpha}{et}(\eta e - 1)\right)$$

for every $\eta > 0$.

Since $\mathbb{E}[\mathsf{Ur_f}(I_j)] \geq 2\epsilon/r$, Proposition 8.7 yields

$$\Pr\left[\mathsf{Ur_f}(I_j) \leq \epsilon/er\right] < \exp\left(-\frac{\epsilon}{ert}\right) = e^{-2} < 1/4$$

as desired. ∎

8.2 Variation Dominates Unique-Variation

In this section we prove Lemma 8.6, showing that the unique-variation of every set $I \subseteq [n]$ of coordinates cannot exceed its variation. For this purpose we need to refine the sub-additivity property of the variation (Proposition 7.2). While the sub-additivity property states that $\mathsf{Vr_f}(A \cup B) - \mathsf{Vr_f}(A) \leq \mathsf{Vr_f}(B)$ for every subsets A, B of coordinates, the property needed for the proof of Lemma 8.6 is the following.

Lemma 8.8 (diminishing marginal variation). *For every sets $A, B, C \subseteq [n]$ of coordinates,*

$$\mathsf{Vr_f}(A \cup B) - \mathsf{Vr_f}(A) \leq \mathsf{Vr_f}(B \cup (A \cap C)) - \mathsf{Vr_f}(A \cap C)$$

Let us first show how this property implies Lemma 8.6.

Proof of Lemma 8.6: In fact we show that the unique-variation of $I \subseteq [n]$ is bounded from above by $\mathsf{Vr}_{\mathsf{f}}(I \setminus \mathcal{J})$.

For $i \in [n]$ denote $A_i \doteq [i-1] \setminus \mathcal{J}$, and $B_i \doteq \{i\} \setminus \mathcal{J}$. It follows from Lemma 8.8 that for every $i \in I$,

$$\begin{aligned} \mathsf{Vr}_{\mathsf{f}}([i] \setminus \mathcal{J}) - \mathsf{Vr}_{\mathsf{f}}([i-1] \setminus \mathcal{J}) &= \mathsf{Vr}_{\mathsf{f}}(A_i \cup B_i) - \mathsf{Vr}_{\mathsf{f}}(A_i) \leq \\ &\leq \mathsf{Vr}_{\mathsf{f}}(B_i \cup (A_i \cap I)) - \mathsf{Vr}_{\mathsf{f}}(A_i \cap I) = \\ &= \mathsf{Vr}_{\mathsf{f}}(([i] \cap I) \setminus \mathcal{J}) - \mathsf{Vr}_{\mathsf{f}}(([i-1] \cap I) \setminus \mathcal{J}) \end{aligned}$$

Hence

$$\begin{aligned} \mathsf{Ur}_{\mathsf{f}}(I) = \sum_{i \in I} \mathsf{Ur}_{\mathsf{f}}(i) &= \sum_{i \in I} \Big(\mathsf{Vr}_{\mathsf{f}}([i] \setminus \mathcal{J}) - \mathsf{Vr}_{\mathsf{f}}([i-1] \setminus \mathcal{J}) \Big) \leq \\ &\leq \sum_{i \in I} \Big(\mathsf{Vr}_{\mathsf{f}}(([i] \cap I) \setminus \mathcal{J}) - \mathsf{Vr}_{\mathsf{f}}(([i-1] \cap I) \setminus \mathcal{J}) \Big) = \\ &= \sum_{i=1}^{n} \Big(\mathsf{Vr}_{\mathsf{f}}(([i] \cap I) \setminus \mathcal{J}) - \mathsf{Vr}_{\mathsf{f}}(([i-1] \cap I) \setminus \mathcal{J}) \Big) = \mathsf{Vr}_{\mathsf{f}}(I \setminus \mathcal{J}) \end{aligned}$$

as required. ■

Proof of Lemma 8.8

Let us now prove Lemma 8.8, in order to conclude the proof of Lemma 8.6. First, we define the averaging projection with respect to a set of coordinates, and show some of its elementary properties. We then give a formula for the variation of f on a set I, using the function obtained by subtracting from f its average with respect to I. Lemma 8.8 is then obtained by using this formula together with the properties of the averaging projection.

Definition 8.9 (averaging projection). *Let $\mathsf{g} : \mathcal{P}([n]) \to \mathbb{R}$ be any real-valued function, and let z be chosen randomly in $\mathcal{P}(I)$. The* average *of g with respect to the set I is the real-valued function over the domain $\mathcal{P}([n])$, defined by*

$$\mathsf{Avg}_I\,[\mathsf{g}]\,(x) \doteq \mathbb{E}_z[\mathsf{g}((x \setminus I) \sqcup z)]$$

Proposition 8.10 (properties of the averaging projection). *Let $A, B \subseteq [n]$ be sets of coordinates. Then*

1. *The transformation that takes a real-valued function $\mathsf{g} : \mathcal{P}([n]) \to \mathbb{R}$ to $\mathsf{Avg}_A\,[\mathsf{g}]$ is an orthogonal projection. Its image is the subspace of functions that do not depend on coordinates from A*

2. *For every function* $\mathsf{g} : \mathcal{P}([n]) \to \mathbb{R}$,

$$\mathsf{Avg}_A\left[\mathsf{Avg}_B\left[\mathsf{g}\right]\right] = \mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]$$

3. *For every function* $\mathsf{g} : \mathcal{P}([n]) \to \mathbb{R}$,

$$\left\|\mathsf{Avg}_A\left[\mathsf{g}\right] - \mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\|_2^2 = \left\|\mathsf{Avg}_A\left[\mathsf{g}\right]\right\|_2^2 - \left\|\mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\|_2^2$$

Proof. It is easy to note that the averaging transformation with respect to A is a projection, and that its image is as stated. To see that it is orthogonal, take $w \in \mathcal{P}([n] \setminus A)$ and $z \in \mathcal{P}(A)$ to be random and independent. The orthogonality of the averaging projection follows since for every function $\mathsf{g} : \mathcal{P}([n]) \to \mathbb{R}$,

$$\begin{aligned}&\left\langle \mathsf{g} - \mathsf{Avg}_A\left[\mathsf{g}\right], \mathsf{Avg}_A\left[\mathsf{g}\right]\right\rangle = \\ &\qquad = \mathbb{E}_w\mathbb{E}_z\left[\big(\mathsf{g}(w\sqcup z) - \mathsf{Avg}_A\left[\mathsf{g}\right](w\sqcup z)\big)\mathsf{Avg}_A\left[\mathsf{g}\right](w\sqcup z)\right] = \mathbb{E}_w[0] = 0\end{aligned}$$

The verification of the second statement is straightforward, and we omit it.

Let us verify the third property. It follows from the second statement that

$$\left\langle \mathsf{Avg}_A\left[\mathsf{g}\right], \mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\rangle = \left\langle \mathsf{Avg}_A\left[\mathsf{g}\right], \mathsf{Avg}_B\left[\mathsf{Avg}_A\left[\mathsf{g}\right]\right]\right\rangle$$

and then from the orthogonality of the averaging projection with respect to B, we have

$$\left\langle \mathsf{Avg}_A\left[\mathsf{g}\right], \mathsf{Avg}_B\left[\mathsf{Avg}_A\left[\mathsf{g}\right]\right]\right\rangle = \left\|\mathsf{Avg}_B\left[\mathsf{Avg}_A\left[\mathsf{g}\right]\right]\right\|_2^2 = \left\|\mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\|_2^2$$

Hence

$$\begin{aligned}&\left\|\mathsf{Avg}_A\left[\mathsf{g}\right] - \mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\|_2^2 = \\ &\qquad = \left\|\mathsf{Avg}_A\left[\mathsf{g}\right]\right\|_2^2 + \left\|\mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\|_2^2 - 2\left\langle \mathsf{Avg}_A\left[\mathsf{g}\right], \mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\rangle = \\ &\qquad\qquad = \left\|\mathsf{Avg}_A\left[\mathsf{g}\right]\right\|_2^2 - \left\|\mathsf{Avg}_{A\cup B}\left[\mathsf{g}\right]\right\|_2^2\end{aligned}$$

■

Let us now express the variation of a set of coordinates in terms of the averaging projection.

Proposition 8.11 (variation vs. average). *For every set* $I \subseteq [n]$ *of coordinates,*

$$\mathsf{Vr}_{\mathsf{f}}(I) = \left\|\mathsf{f} - \mathsf{Avg}_I\left[\mathsf{f}\right]\right\|_2^2$$

Proof. For every $w \in \mathcal{P}([n] \setminus I)$ let α_w be defined by

$$\alpha_w \doteq \Pr_{z \in \mathcal{P}(I)} [\mathsf{f}(w \sqcup z) = 1]$$

Then for every $w \in \mathcal{P}([n] \setminus I)$ and $z \in \mathcal{P}([I])$, $\mathsf{Avg}_I [\mathsf{f}] (w \sqcup z) = 2\alpha_w - 1$. Hence if w is fixed, and $z, z_1, z_2 \in \mathcal{P}(I)$ are chosen randomly and independently, one can easily verify that

$$\mathbb{E}_z\left[\left(\mathsf{f}(w \sqcup z) - \mathsf{Avg}_I [\mathsf{f}] (w \sqcup z)\right)^2\right] = 4\alpha_w(1 - \alpha_w)$$

and that

$$\Pr_{z_1, z_2} [\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)] = 2\alpha_w(1 - \alpha_w)$$

Now if $w \in \mathcal{P}([n] \setminus I)$ and $z, z_1, z_2 \in \mathcal{P}(I)$ are all chosen randomly and independently, we have

$$\begin{aligned}\|\mathsf{f} - \mathsf{Avg}_I [\mathsf{f}]\|_2^2 &= \mathbb{E}_w \mathbb{E}_z\left[\left(\mathsf{f}(w \sqcup z) - \mathsf{Avg}_I [\mathsf{f}] (w \sqcup z)\right)^2\right] = \\ &= \mathbb{E}_w[4\alpha_w(1 - \alpha_w)] = \\ &= 2\mathbb{E}_w[\Pr_{z_1, z_2} [\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)]] = \\ &= 2 \Pr_{w, z_1, z_2} [\mathsf{f}(w \sqcup z_1) \neq \mathsf{f}(w \sqcup z_2)] = \mathsf{Vr}_{\mathsf{f}}(I)\end{aligned}$$

■

We now return to the proof of Lemma 8.8. According to proposition 8.11, it suffices to show that

$$\|\mathsf{f} - \mathsf{Avg}_{A \cup B} [\mathsf{f}]\|_2^2 - \|\mathsf{f} - \mathsf{Avg}_A [\mathsf{f}]\|_2^2 \ \leq \ \|\mathsf{f} - \mathsf{Avg}_{B \cup (A \cap C)} [\mathsf{f}]\|_2^2 - \|\mathsf{f} - \mathsf{Avg}_{A \cap C} [\mathsf{f}]\|_2^2 \qquad (\circ)$$

By applying the third statement in proposition 8.10 to both terms in the left-hand side of $(\circ)$, and noting that $\mathsf{f} = \mathsf{Avg}_\emptyset [\mathsf{f}]$, we have

$$\begin{aligned}&\|\mathsf{f} - \mathsf{Avg}_{A \cup B} [\mathsf{f}]\|_2^2 - \|\mathsf{f} - \mathsf{Avg}_A [\mathsf{f}]\|_2^2 = \\ &\qquad = \|\mathsf{Avg}_A [\mathsf{f}]\|_2^2 - \|\mathsf{Avg}_{A \cup B} [\mathsf{f}]\|_2^2 = \|\mathsf{Avg}_A [\mathsf{f}] - \mathsf{Avg}_{A \cup B} [\mathsf{f}]\|_2^2\end{aligned}$$

Similarly,

$$\|\mathsf{f} - \mathsf{Avg}_{B \cup (A \cap C)} [\mathsf{f}]\|_2^2 - \|\mathsf{f} - \mathsf{Avg}_{A \cap C} [\mathsf{f}]\|_2^2 = \|\mathsf{Avg}_{A \cap C} [\mathsf{f}] - \mathsf{Avg}_{B \cup (A \cap C)} [\mathsf{f}]\|_2^2$$

Now, according to the second statement in Proposition 8.10, and since an orthonormal projection cannot increase the ℓ_2 norm of a vector, we have

$$\begin{aligned}&\|\mathsf{Avg}_A [\mathsf{f}] - \mathsf{Avg}_{A \cup B} [\mathsf{f}]\|_2^2 = \|\mathsf{Avg}_A \left[\mathsf{Avg}_{A \cap C} [\mathsf{f}] - \mathsf{Avg}_{B \cup (A \cap C)} [\mathsf{f}]\right]\|_2^2 \leq \\ &\qquad \leq \|\mathsf{Avg}_{A \cap C} [\mathsf{f}] - \mathsf{Avg}_{B \cup (A \cap C)} [\mathsf{f}]\|_2^2\end{aligned}$$

This implies Inequality $(\circ)$, thus completing the proof of Lemma 8.6.

Chapter 9

Improving the Query Complexity

In this chapter we present two tests for the property of being a J-junta, that obtain improved query complexity with respect to the size test shown in Chapter 8. The first test uses adaptivity in order to reduce the query complexity, and the other is two-sided, namely it may reject a J-junta with probability up to 1/3.

9.1 Improving the Query Complexity Using Adaptivity

The size test applies several iterations of the independence test to every subset in the partition, in order to detect whether it has a non-negligible variation. Here we show how using adaptivity, it is possible to detect all the subsets in the partition that have non-negligible variation using less queries (reducing one power of the dependency on J).

Theorem 9.1. *Set $r = 16J^2$ (as in the size test). Then there exists an adaptive one-sided J-junta test, that uses*

$$\frac{32erJ(1+\log_2 r)\ln(32J(1+\log_2 r))}{\epsilon} = \Theta\big(J^3 \ln^2(J+1)/\epsilon\big)$$

queries.

Proof. The idea of the adaptive test is to speed up the finding of the subsets of the partition with non-negligible variation as follows: Instead of applying the independence test to each subset individually, we apply it to blocks, each of which is a union of several such subsets. If f is not found to depend on a block, then all of its elements can be declared to be 'variation free' at once. When f is found to depend on a block, the algorithm divides the block into two equally sized sub-blocks, for which the process is repeated.

Definition 9.2 (blocks). *Fix a partition $I_1, \ldots, I_r$ of the coordinates. A set B of coordinates is called a block, if it is the union of a positive number of subsets in the partition. The size of the block is the number of subsets in the partition that take part in this union.*

The adaptive test. The adaptive test begins by randomly partitioning the coordinates into subsets $I_1, \ldots, I_r$. The test maintains, throughout its operation, a set $S = \{B_1, \ldots, B_l\}$ of at most J disjoint blocks with respect to this partition. The blocks in S supposedly contain all the sets I_j in the partition that have non-negligible variation. Initially S is set to have only one block which contains all coordinates, namely $S = \{[n]\}$. In each step, the test performs the following.

- If all the blocks in S are of size one, accept (in this case at most J elements of the partition supposedly have non-negligible variation).
- Otherwise, choose a block $B \in S$ that has the largest size. Remove B from S, and partition it arbitrarily into two sub-blocks $B = B' \cup B''$, whose sizes differ by at most 1.
- Apply $\frac{4er\ln(32J(1+\log_2 r))}{\epsilon}$ iterations of the independence test to B'. If f is found to depend on B', then insert B' into S, and otherwise discard it. Apply the same treatment to B''.
- If the size of S is now greater than J, reject (f depends on each of the subsets in S, hence it is not a J-junta). Otherwise continue to the next step.

The adaptive test obviously accepts with probability 1 if f is a J-junta. To bound the number of rounds, we note that if after round T the maximum size of the blocks is m, then clearly after round $T + J$ the maximum size of the blocks is no more than $\lceil \frac{m}{2} \rceil$. This implies that the algorithm terminates after at most $2J(1 + \log_2 r)$ steps, and that each step uses $\frac{16er\ln(32J(1+\log_2 r))}{\epsilon}$ queries. The total number of queries made is therefore as required.

To prove Theorem 9.1, it is left to show that if f passes the test with probability at least 1/2, then it is an (ϵ, J)-junta.

Proposition 9.3 (soundness). *If f passes the adaptive-test with probability $1/2$, then it is an (ϵ, J)-junta.*

Proof. Let $t = \frac{\epsilon}{2er}$ and let $\mathcal{J}$ be defined as the set of coordinates i for which $\mathsf{Vr_f}(\{i\}) > t$ (as in Section 8.1). It suffices to prove that $|\mathcal{J}| \leq J$ and that $\mathsf{Vr_f}(\bar{\mathcal{J}}) \leq 2\epsilon$. Assume on the contrary that this is not the case, and let us prove that the adaptive-test rejects with probability at least 1/2.

According to the proof of Proposition 8.1, if $|\mathcal{J}| > J$ then with probability at least 3/4 there are at least $J + 1$ subsets in the partition $I_1, \ldots, I_r$ whose variation is at least t. Moreover, it is shown in Section 8.1 that if $\mathsf{Vr_f}(\bar{\mathcal{J}}) > 2\epsilon$, then with probability at least 2/3 there are at least $J+1$ subsets in the partition, whose variation is at least $\epsilon/er = 2t$. In both cases, with probability at least $2/3$ there are at least $J+1$ subsets in the partition whose variation is at least t.

To complete the proof we show that if there are at least $J+1$ subsets with variation at least t in the partition $I_1, \dots, I_r$ chosen by the adaptive test, then the probability that it accepts is at most $1/8$. This holds since in order to accept, the test must at some point discard a block whose variation is at least t. The probability of discarding each such block is at most

$$(1-\frac{t}{2})^{\frac{4er\ln(32J(1+\log_2 r))}{\epsilon}} \leq e^{-\ln(32J(1+\log_2 r))} = \frac{1}{32J(1+\log_2 r)}$$

The test encounters two blocks at each step, so summing over all steps bounds the probability that such a block is discarded throughout the test by $1/8$. ■

This concludes the proof of Theorem 9.1. ■

9.2 Improving the Query Complexity Using Two-Sidedness

In this section we present a test with a significantly reduced query complexity. It makes $\Theta(J^2 \ln^2(J+1)/\epsilon)$ queries, removing a J^2 factor from the query complexity of the size test. The test is two-sided, namely we allow it to reject a J-junta with probability at most $1/3$, on condition that it rejects an input that is not an (ϵ, J)-junta with probability at least $2/3$.

Theorem 9.4. *Let* $\epsilon > 0$ *be any positive number, and fix* $r \doteq 16J^2$, $s \doteq 20J(3+\ln r)$, *and* $h \doteq \frac{6er(3+2\ln s)}{\epsilon J}$. *Then there exists a non-adaptive* J*-junta test, which makes* $2sh = \Theta(J^2 \ln^2(J+1)/\epsilon)$ *queries, and satisfies the following.*

- *Every* J*-junta is accepted with probability at least* $2/3$.
- *Any input which is not an* (ϵ, J)*-junta is rejected with probability at least* $2/3$.

Proof. As in the size test, the two-sided test randomly partitions the coordinates into r subsets. In order to reduce the number of queries, the two-sided test finds subsets in the partition that have non-negligible variation by applying the independence test to blocks of such subsets (see Definition 9.2), much like the adaptive test presented above.

The two-sided test. First, the test randomly partitions the coordinates into r subsets $I_1, \dots, I_r$. Then it picks s random subsets $\Lambda_1, \dots, \Lambda_s \subseteq [r]$ independently, each of size J. Each set Λ_l determines a block $B_l \doteq \bigcup_{j \in \Lambda_l} I_j$, to which the test applies h iterations of the independence test.

Acceptance conditions. The test declares a block B_l to be variation-free if none of the independence test iterations applied to it finds f to depend on it. If B_l is declared variation-free, then all the subsets I_j contained in it are declared variation-free on its behalf. The test accepts f if both of the following conditions hold.

- At least half of the blocks $B_1, \ldots, B_s$ are declared variation free
- Except for at most J subsets, every subset in the partition $I_1, \ldots, I_r$ is declared variation-free on behalf of *some* block

Properties of the test. It is obvious that the test performs $2sh$ queries, as required. It is left to show that a J-junta is accepted by the test with probability at least $2/3$, and that an input which is not an (ϵ, J)-junta is rejected with probability at least $2/3$. This is proven in the next two lemmas.

Lemma 9.5 (completeness). *If* f *is a* J*-junta, then it passes the two-sided test with probability at least* $2/3$*.*

Proof. Fix any partition $I_1, \ldots, I_r$. If f is a J-junta, then it is independent of all subsets in the partition, except for at most J of them. Hence for any fixed l, the probability over the selection of the blocks that f is independent of B_l is at least

$$\left(1 - \frac{J}{r - J + 1}\right)^J > 1 - \frac{J^2}{r - J} \geq \frac{14}{15}$$

The probability that f depends on more than half of the blocks is therefore smaller than $\frac{2}{15} < \frac{1}{6}$, using the Markov inequality. Hence with probability at least $1 - \frac{1}{6}$, at least half of the blocks are declared variation-free, and the first acceptance condition holds.

Now fix j such that f does not depend on I_j, and let us bound the probability that it is not declared variation-free. Conditioned on the event that f does not depend on B_l, the probability that in addition B_l contains I_j is at least $J/r = 1/16J$. Hence I_j is declared variation-free on behalf of B_l with probability at least $1/20J$, for every fixed l. The probability that I_j is not declared variation-free is therefore bounded by

$$\left(1 - \frac{1}{20J}\right)^s = \left(1 - \frac{1}{20J}\right)^{20J(3+\ln r)} < \frac{1}{6r}$$

It follows that with probability at least $1 - \frac{1}{6}$, all the subsets in the partition on which f does not depend are declared variation-free (in this case the second acceptance condition is fulfilled). Overall we have that with probability at least $2/3$, both conditions for acceptance are satisfied. ■

Lemma 9.6 (soundness). *If* f *passes the two-sided test with probability higher than* $2/3$*, then it is an* (ϵ, J)*-junta.*

Proof. Let $t = \frac{\epsilon J}{3er}$ and let $\mathcal{J}$ denote the set of all coordinates i for which $\mathsf{Vr_f}(\{i\}) > t$. As shown in Chapter 8, it suffices to prove that $|\mathcal{J}| \leq J$ and that $\mathsf{Vr_f}(\bar{\mathcal{J}}) < 2\epsilon$. Assume on the contrary that this is not the case, and let us prove that the two-sided test rejects with probability at least $2/3$.

First case, $|\mathcal{J}| > J$. As in the proof of proposition 8.1, if $|\mathcal{J}| > J$ then with probability at least $3/4$ there are at least $J+1$ subsets in the partition $I_1, \ldots, I_r$ with variation at least t. To conclude this case, we show that the probability of each such subset to be declared variation-free is bounded by $\frac{1}{12(J+1)}$.

Let I_j be a subset whose variation is at least t, and let B_l be a block that contains it. By the monotonicity of the variation we have $\mathsf{Vr_f}(B_l) > t$, so each iteration of the independence test on B_l detects a dependency of f on B_l with probability at least $t/2$. The probability of B_l to be declared variation-free is therefore bounded by

$$(1 - t/2)^h = (1 - t/2)^{2\cdot(3+2\ln s)/t} < \\ < \frac{1}{12s(J+1)}$$

Since I_j is contained in at most s blocks, the probability of it being declared variation-free is bounded by $1/12(J+1)$, as required.

Second case, $\mathsf{Vr_f}(\bar{\mathcal{J}}) \geq 2\epsilon$. Let us fix one index l, and show that B_l has high variation with very high probability. This will imply that with high-probability, the number of blocks *not* declared variation-free is larger than $s/2$, and the test rejects.

It follows from the procedure of choosing the partition and the blocks, that B_l is in fact a random set of coordinates, independently containing each coordinate $i \in [n]$ with probability J/r. We now consider the unique-variation as in Definition 8.5, only with respect to the set $\mathcal{J}$ as defined here. Then the expectation of $\mathsf{Ur_f}(B_l)$ is given by

$$\mathbb{E}\left[\mathsf{Ur_f}(B_l)\right] = \frac{J}{r}\sum_{i\in[n]} \mathsf{Ur_f}(i) = \frac{J}{r}\mathsf{Vr_f}(\bar{\mathcal{J}}) \geq 2\epsilon J/r$$

Moreover, the unique-variation of B_l is a sum of non-negative independent random variables, each bounded by t. It thus follows from Lemma 8.6 and Proposition 8.7 that

$$\Pr\left[\mathsf{Vr_f}(B_l) < \frac{\epsilon J}{er}\right] \leq \Pr\left[\mathsf{Ur_f}(B_l) < \frac{\epsilon J}{er}\right] < \exp\left(-\frac{\epsilon J}{ert}\right) = e^{-3} < 1/12$$

We say that a block B_l is detectable if its variation is at least $\epsilon J/er$. The expected number of undetectable blocks is therefore smaller than $s/12$. It follows from the Markov inequality that with probability at least $1 - \frac{1}{6}$, there are less that $s/2$ undetectable blocks, and therefore there are more than $s/2$ detectable blocks. The probability of a detectable block to be declared variation-free is bounded by

$$\left(1-\frac{\epsilon J}{2er}\right)^h < \exp\left(-(9+6\ln s)\right) < \frac{1}{6s} \quad ,$$

and therefore with probability at least $1-\frac{1}{6}$, none of the detectable blocks are declared variation-free. Overall we have that with probability at least $2/3$, the number of detectable blocks is more than $s/2$, and none of them is declared variation-free, and therefore the test rejects. ■

This concludes the proof of Theorem 9.4. ■

Chapter 10

Lower Bound, and a Random Walks on $\mathbb{Z}_2^q$

We use Yao's principle, which states that to show a lower bound on the complexity of a randomized test, it is enough to present an input distribution for which any deterministic test with that complexity is likely to fail.

We define distributions D_P, D_N on positive (J-junta) and negative ($\frac{1}{2}$-far from any J-junta) inputs, respectively. Our input distribution first chooses D_P or D_N with equal probability and then draws an input according to the chosen distribution. We show that every deterministic non-adaptive test with $q = \tilde{O}(\sqrt{J})$ queries has error probability larger than 1/3 (with respect to the induced probability on inputs). For this purpose we show that for any set of $q = \tilde{O}(\sqrt{J})$ vertices of the hypercube, the distributions D_P and D_N induced on $\{-1,1\}^q$ by restricting the functions to these q vertices have a variation distance less than $\frac{1}{3}$.

The distributions D_P and D_N are simply uniform distributions over characters χ_S of size J and $J+2$ respectively. We will, however, work with two auxiliary distributions $\tilde{D}_P, \tilde{D}_N$, which are close to D_P and D_N, and which are easier to analyze. To choose a function from $\tilde{D}_P$, we choose a random set $S \subseteq [n]$, $|S| \le J$, by picking J random elements in $[n]$ *with repetition*, and take the character χ_S. The distribution $\tilde{D}_N$ is defined in the same manner, but that we choose $J+2$ elements in $[n]$.

Note that if $|S| > J$, then the character χ_S is $\frac{1}{2}$-far from any J-junta; and that both $\left|D_P - \tilde{D}_P\right|$ and $\left|D_N - \tilde{D}_N\right|$ are bounded by $O\left(\frac{J^2}{n}\right)$.

Now, consider the distributions induced by $\tilde{D}_P$, $\tilde{D}_N$ on $\{-1,1\}^q$. Let $x_1, \ldots, x_q$ be the queries, and let M be a $q \times n$ boolean matrix, with rows $x_1, \ldots, x_q$. To choose an element x of $\{-1,1\}^q$ according to the first distribution, we choose at random, allowing repetitions, J columns of M and sum them up. This gives us an element y of $\{0,1\}^q$. We take $x = (-1)^y$, where the power operation is performed coordinate-wise. The same holds for the second distribution, the only difference being that we choose $J+2$ columns.

For $x \in \mathbb{Z}_2^q$, let $P(x)$ be the probability of choosing x when we pick a column of M at random. Consider a random walk on $\mathbb{Z}_2^q \cong \{-1,1\}^q$, starting at 0, in which at every step we choose an element of the cube according to P and add it to the current location. Let P_t be the distribution induced by this walk after t steps. Note that P_J, P_{J+2} are precisely the distributions induced by $\tilde{D}_P$, $\tilde{D}_N$. Note also that P_t is the distribution of $Y \oplus Y \oplus \ldots \oplus Y$, where we sum t independent copies of a $\mathbb{Z}_2^q$-valued random variable Y, taking every value x with probability $P(x)$.

We want to show that for t sufficiently large compared to q, the distributions P_t, P_{t+2} are close in the variation distance. This is Theorem 6.5, presented in the introduction. Theorem 6.4 (see the introduction) now follows as an immediate corollary.

Theorem 6.5 is proven below. We first give a very brief overview of the proof.Every element x of $\mathbb{Z}_2^q$ defines a partition of the space into a subspace $V_0 = \{y : \langle y, x\rangle = 0\}$ and its complement V_1. x is said to be a *degenerate direction* if the probability of either of these sets is at most $\tilde{O}\left(q^{-1}\right)$. The proof is inductive on the dimension q. We distinguish between two cases: if there are no degenerate directions, then the random walk is exponentially close to being stationary after $\tilde{O}\left(q^2\right)$ steps, and the claim holds. If, on the other hand, there is a degenerate direction x, then the walk 'splits' into two 'independent' walks, one on V_0 and one on V_1, each of which is isomorphic to $\mathbb{Z}_2^{q-1}$, and we can use induction.

Proof of Theorem 6.5

Let us consider the distribution P_t of the walk at time t. Recall that the distribution of the sum of two independent random variables is the convolution of their distributions, $(P * Q)(x) = \sum_y P(y)Q(x \oplus y)$. This implies that P_t is the t-wise convolution of P, which we will denote by P^{*t}.

Now, for any $r \le t$ we have $|P_t - P_{t+2}| = \left|P^{*t} - P^{*(t+2)}\right| = \left|P^{*(t-r)} * \left(P^{*r} - P^{*(r+2)}\right)\right| = \left|P^{*(t-r)} * (P_r - P_{r+2})\right|$. The following fact is well-known and easy: for any two functions f, g on $\mathbb{Z}_2^q$ it holds that $\|\mathsf{f} * \mathsf{g}\|_1 \le 2^q \|\mathsf{f}\|_1 \|\mathsf{g}\|_1$. Taking into account that $P^{*(t-r)}$ is a distribution we deduce

$$|P_t - P_{t+2}| = \left|P^{*(t-r)} * (P_r - P_{r+2})\right| = 2^{q-1} \cdot \|P^{*(t-r)} * (P_r - P_{r+2})\|_1 \le$$

$$2^{q-1} \cdot \|P_r - P_{r+2}\|_1 = |P_r - P_{r+2}|.$$

Therefore, the distance $|P_t - P_{t+2}|$ is monotone non-increasing in t, and we are interested in the first time $t = t(q)$ for which P_t and P_{t+2} are δ-close. We show $t(q) \le O\left(\frac{\log\frac{1}{\delta}}{\delta} \cdot b(q)\right)$, where we set $b(q) \doteq q^2 \log^2(q+1)$.

This is an immediate consequence of the following proposition, where S is the sum of the convergent series $\sum_{J=1}^{\infty} \frac{J}{b(J)}$:

Proposition 10.1. *There exists an absolute constant C such that for any $q \geq 1$, for any distribution P on $\mathbb{Z}_2^q$, and for any $t \geq C^{\frac{\log\frac{1}{\delta}}{\delta}} \cdot b(q)$,*

$$|P_t - P_{t+2}| \leq \frac{\delta}{S} \cdot \sum_{k=1}^{q} \frac{k}{b(k)} < \delta.$$

Proof. The proof is by induction on q.

We will assume, where needed, that C is sufficiently large. We set $t = C^{\frac{\log\frac{1}{\delta}}{\delta}} \cdot b(q)$, assuming this is an integer.

The case $q = 1$ is easy. It is possible to show that for a distribution P on $\mathbb{Z}_2$ with $P(0) = p$ and $P(1) = 1 - p$, we have $|P_t - P_{t+2}| = \frac{1}{2} \cdot \left|(2p-1)^t - (2p-1)^{t+2}\right|$. A simple analysis shows that if $t \geq C^{\frac{\log\frac{1}{\delta}}{\delta}}$, the last expression is at most $\frac{\delta}{S}$.

Assume the claim for holds for $q - 1$. We proceed with simple Fourier analysis, and show that our claim is true if all the non-zero Fourier coefficients of P are relatively small (a nice way to see this, though the actual proof is even simpler, is that this condition on the Fourier coefficients implies that P_t converges rapidly to the uniform distribution U, and $|P_t - P_{t+2}| \leq |P_t - U| + |U - P_{t+2}|$). We have

$$|P_t - P_{t+2}|^2 = 2^{2q-2} \cdot \|P_t - P_{t+2}\|_1^2 \leq 2^{2q-2} \cdot \|P_t - P_{t+2}\|_2^2 =$$

$$2^{2q-2} \cdot \sum_R \left(\widehat{P}_t(R) - \widehat{P_{t+2}}(R)\right)^2 = \frac{1}{4} \cdot \sum_R \left(a^t(R) - a^{t+2}(R)\right)^2, \qquad (10.1)$$

where $a(R) \doteq 2^q \widehat{P}(R)$.

Clearly, $a(\emptyset) = \sum_x P(x) = 1$. Now consider the case in which, for all $R \neq \emptyset$ we have $|a(R)| \leq 1 - \frac{\delta q}{\sqrt{C} b(q)}$. In this case, the right hand side of (10.1) is at most

$$\sum_{R \neq \emptyset} a^{2t}(R) \leq 2^q \cdot \left(1 - \frac{\delta q}{\sqrt{C} b(q)}\right)^{2C^{\frac{\log\frac{1}{\delta}}{\delta}} \cdot b(q)} \leq 2^q \cdot exp\left\{-2\sqrt{C} \cdot \log\frac{1}{\delta} \cdot q\right\}.$$

This is smaller than $\frac{\delta}{S} \leq \frac{\delta}{S} \cdot \sum_{k=1}^{q} \frac{k}{b(k)}$.

It remains to deal with the case where P has large Fourier coefficients. Let R be such that $|a(R)| \geq 1 - \frac{\delta q}{\sqrt{C} b(q)}$.

We make two assumptions for the sake of clarity: we assume that $R = e_1 \doteq (10 \cdots 0)$, and that $a(R) \geq 0$. Both assumptions are easily shown not to lead to loss of generality and we omit the proofs.

Now, $a(e_1) = P\{x : x_1 = 0\} - P\{x : x_1 = 1\}$. It follows that $P\{x : x_1 = 1\} \leq \frac{\delta q}{2\sqrt{C} b(q)}$.

Observe that the direction e_1 is *degenerate*, it partitions the cube $\{0,1\}^q$ into two subcubes $V_0 = \{x : x_1 = 0\}$, and $V_1 = \{x : x_1 = 1\}$, both of which are isomorphic to $\mathbb{Z}_2^{q-1}$.

Because of the degeneracy of e_1, the walk will find it hard to leave the subcube it is in, and we will 'split' it into two walks, on V_0, V_1, and use the induction hypothesis for these walks.

For $i = 0, 1$ and for $r = t, t + 2$ we set $P_r^i \doteq (P_r|V_i)$. All four distributions so obtained can be viewed as distributions on $\mathbb{Z}_2^{q-1}$.

We write P_t as a convex combination $P_t = P_t(V_0) \cdot P_t^0 + P_t(V_1) \cdot P_t^1$, and do the same for P_{t+2}. Note that $\left|P_t(V_0) - P_{t+2}(V_0)\right| \leq \frac{\delta q}{\sqrt{C} b(q)}$. We will show, using the induction hypothesis, that for $i = 0, 1$ we have

$$\left|P_t^i - P_{t+2}^i\right| \leq \frac{\delta}{S} \cdot \left(\sum_{k=1}^{q-1} \frac{k}{b(k)} + \frac{q}{2b(q)}\right).$$

This will conclude the proof, since

$$|P_t - P_{t+2}| \leq 2\left|P_t(V_0) - P_{t+2}(V_0)\right| + \left|P_t(V_0) \cdot \left(P_t^0 - P_{t+2}^0\right) + P_t(V_1) \cdot \left(P_t^1 - P_{t+2}^1\right)\right| \leq$$

$$\frac{\delta}{S} \cdot \left(\sum_{k=1}^{q-1} \frac{k}{b(k)} + \frac{q}{2b(q)}\right) + \frac{2\delta q}{\sqrt{C} b(q)} \leq \frac{\delta}{S} \cdot \sum_{k=1}^{q} \frac{k}{b(k)}.$$

Let $P^0 = (P|V_0)$ and $P^1 = (P|V_1)$. Let N_r be a random variable counting the number of times the walk makes a step in direction x with $x_1 = 1$ during the first r steps.

Let $i = 0$. The other case is treated similarly.

The central (though simple) point of the argument is that for any r and for any even ℓ we have

$$(P_r^0 | N_r = \ell) = (P^1)^{*\ell} * (P^0)^{*(t-\ell)}.$$

This is true because the distribution on the left hand side is the distribution on $\mathbb{Z}_2^{q-1}$ given that the walk makes ℓ 'odd' steps x with $x_1 = 1$ and $r - \ell$ 'even' steps, with $x_1 = 0$. Since the addition in $\mathbb{Z}_2^q$ is commutative, we might as well assume that all the odd steps were made first, giving the right hand side.

Therefore, P_r^0 can be written as a convex combination

$$P_r^0 = \sum_{\ell \leq r, \ell \text{ even}} Pr(N_r = \ell) \cdot (P^1)^{*\ell} * (P^0)^{*(t-\ell)}.$$

Using this, we can bound $\left|P_t^0 - P_{t+2}^0\right|$:

$$\left|P_t^0 - P_{t+2}^0\right| \leq Pr(N_t \neq N_{t+2}) + Pr\left(N_t \geq \sqrt{C} \cdot \log\frac{1}{\delta} \cdot q\right) +$$

$$\sum_{\ell \leq \sqrt{C} \cdot \log\frac{1}{\delta} \cdot q, \ell \text{ even}} Pr(N_r = \ell) \cdot \left|(P^1)^{*\ell} * \left((P^0)^{*(t-\ell)} - (P^0)^{*(t+2-\ell)}\right)\right|. \qquad (10.2)$$

The first summand in (10.2) is equal to the probability that an odd step was made in one of the times $t+1, t+2$, and this is at most $\frac{\delta q}{\sqrt{C} b(q)}$.

As to the second summand, observe that N_t is a binomial random variable with parameters $t = C^{\frac{\log \frac{1}{\delta}}{\delta}} \cdot b(q)$ and $p = \frac{\delta q}{2\sqrt{C} b(q)}$. The probability of the second summand is that of $N_t \geq \sqrt{C} \cdot \log \frac{1}{\delta} \cdot q$, and this, using Chernoff bounds, is at most $exp\left\{-2\sqrt{C} \cdot \log \frac{1}{\delta} \cdot q/27\right\}$.

Thus, the sum of the two first summands is bounded from above by $\frac{\delta}{S} \cdot \frac{q}{2b(q)}$.

It remains to deal with the third summand. For $\ell \leq \sqrt{C} \cdot \log \frac{1}{\delta} \cdot q$ we have $t - \ell \geq C^{\frac{\log \frac{1}{\delta}}{\delta}} \cdot b(q) - \sqrt{C} \cdot \log \frac{1}{\delta} \cdot q \geq C^{\frac{\log \frac{1}{\delta}}{\delta}} \cdot b(q-1)$, and therefore we may use the induction hypothesis to conclude

$$\left|(P^1)^{*\ell} * \left((P^0)^{*(t-\ell)} - (P^0)^{*(t+2-\ell)}\right)\right| \leq \left|(P^0)^{*(t-\ell)} - (P^0)^{*(t+2-\ell)}\right| \leq \frac{\delta}{S} \cdot \sum_{k=1}^{q-1} \frac{k}{b(k)}.$$

Consequently, the third summand in (10.2) is bounded from above by $\frac{\delta}{S} \sum_{k=1}^{q-1} \frac{k}{b(k)}$, and

$$\left|P_t^0 - P_{t+2}^0\right| \leq \frac{\delta}{S} \cdot \left(\sum_{k=1}^{q-1} \frac{k}{b(k)} + \frac{q}{2b(q)}\right),$$

concluding the proof of the proposition and of Theorem 6.5. ■

Chapter 11

Testing that f is a Permutation of a Given h

Given a boolean function $\mathsf{h} : \{0,1\}^n \to \{-1,1\}$, we say that a function f is a *permutation* of h if there exists a permutation $\sigma : [n] \to [n]$, such that for every $x = x_1 x_2 \ldots x_n \in \{0,1\}^n$ we have $\mathsf{f}(x) = \mathsf{h}(\sigma(x))$, where we define (with a slight abuse of notation) $\sigma(x) = x_{\sigma(1)} x_{\sigma(2)} \ldots x_{\sigma(n)}$. We show a test for this property for any h that is a J-junta. We first show a test with a linear dependence in ϵ^{-1} but an exponential one in J, and then show how to change it to a test with a polynomial dependence on ϵ^{-1} and J. On the other hand, a closer look at the proof of Theorem 6.4 shows that it also provides a lower bound on testing that f is a permutation of $\mathsf{h}(x) = \chi_{[J]} = x_1 \oplus \ldots \oplus x_J$ which is a function of J, so the limitation that h has a small junta is essential.

The tests constructed in the following are 2-sided. This is not a coincidence, as the following proposition shows that in some cases one needs a number of queries that depends on n to provide a 1-sided test for being a permutation of a given h. On the other hand, [DT99] in particular provides such a 1-sided test making a number of queries that is logarithmic in n.

Proposition 11.1. *A non-adaptive testing algorithm that makes less than* $\log(n/2)$ *queries on* $\mathsf{f}(x)$*, and accepts any permutation of* $\mathsf{h}(x) = x_1 \wedge x_2$ *with probability* 1*, will necessarily accept some permutation of* $\mathsf{h}'(x) = x_1$ *with probability at least* $\frac{1}{2}$.

Proof. Suppose that we are given a sequence of $l = \log(n/2)$ queries, labeled by $q^{(1)} = (x_1^{(1)}, \ldots, x_n^{(1)}) \ \cdots \ q^{(l)} = (x_1^{(l)}, \ldots, x_n^{(l)})$. We define an equivalence relation over $\{1, \ldots, n\}$ by stating that $i \sim i'$ if for every $1 \le j \le l$ we have $x_i^{(j)} = x_{i'}^{(j)}$. We say that i is *isolated* if its equivalence class is $\{i\}$.

We observe that by the choice of l for every set of l queries there exist a set of at least $\frac{n}{2}$ coordinates that are not isolated. Thus, for every non-adaptive testing algorithm there exists a coordinate i with the property that it is not isolated with probability at least $\frac{1}{2}$.

Now, for every query sequence $q^{(1)}, \ldots, q^{(l)}$ for which i is not isolated, and which is taken with positive probability by the algorithm, let i' be such that $i \sim i'$. Since the algorithm has to accept $\mathsf{f}(x) = x_i \wedge x_{i'}$ with probability 1, the algorithm must accept this function when the sequence $q^{(1)}, \ldots, q^{(l)}$ is chosen. But this means that the algorithm must accept also the function $f'(x) = x_i$ when this sequence is chosen, because these two functions are identical on that whole query sequence. Summing up over all query sequences for which i is not isolated, we conclude that the algorithm must accept $f'(x) = x_i$ with probability at least $\frac{1}{2}$, completing the proof. ■

We now turn to the proof of Theorem 6.6. The constructed tests are adaptive, but they could be made non-adaptive with a penalty of an additional poly(J) factor.

A Test with an Exponential Dependency on J

We assume without loss of generality that g depends on all its variables. In this case, the variation of g on every coordinate is at least 2^{1-J}. We begin by performing the J-junta test given by Theorem 6.2 on f, with $\min\{\frac{1}{4}\epsilon, 2^{-J}\}$ as the approximation parameter and $\frac{7}{8}$ as the detection probability (we use the usual amplification techniques). If the test rejects then we reject the input. If it accepts, we note that with high probability we have sets $U_{i_1}, \ldots, U_{i_l}$ of coordinates such that each of them contains exactly one member of a junta $\mathcal{J}$ of a function f' that is close to f (with $l \leq J$). If $l < J$ we reject the input, so from now on let us assume that $l = J$, and for convenience denote $V_j = U_{i_j}$ for $1 \leq j \leq J$.

We first show how to test for the above property in the special case that g is symmetric with regards to permutations of its variables, and then show how to generalize it to every g.

We check $\mathsf{f}(x)$ at a randomly chosen $x \in \{0,1\}^n$ for equality with $\mathsf{g}(x|_{\mathcal{J}})$, and repeat this $h = 12\epsilon^{-1}$ times so that any $\mathsf{f}(x)$ that is $\frac{1}{4}\epsilon$-far from $\mathsf{g}(x|_{\mathcal{J}})$ will be rejected with probability at least $\frac{7}{8}$. However, since we do not know $\mathcal{J}$ (but only $V_1, \ldots, V_J$), we perform the following.

Let us for the randomly chosen x denote by Z_x the set of its zero coordinates $\{i | x_i = 0\}$. We construct $y \in \{0,1\}^J$ as follows. For every j, we perform $3 \cdot 2^J(\log h + \log(2J) + 3)$ iterations of the independence test for $V_j \cap Z_x$, to know with high probability whether $\mathsf{Vr}_\mathsf{f}(Z_x \cap V_j) \geq 2^{-J}$, and do the same for $V_j \backslash Z_x$ (remember that in a J-junta, every set containing a junta coordinate has variation at least 2^{1-J}). The probability that in any of the h iterations such a dependence will not be detected is bounded by $\frac{1}{8}$.

If only $V_j \cap Z_x$ is found to have variation then we set $y_j = 0$, because this means that the junta coordinate in V_j has the value 0. By the same token, if only $V_j \backslash Z_x$ is found to have variation then we set $y_j = 1$, and in any of the other two cases (for the same j) we reject the input. Having constructed y, we compare $\mathsf{f}(x)$ with $\mathsf{g}(y)$. The construction of y according to the variation tests above ensures that with high probability $x|_{\mathcal{J}} = y$, and thus we are done.

For an asymmetric g we use a similar test as above, performing $h = 12J \log(J+1)\epsilon^{-1}$ iterations of the comparison above, to ensure that for *every* permutation $\sigma : [J] \to [J]$, if $\mathsf{f}(x)$ is ϵ-far from $\mathsf{g}(x|_{\sigma(\mathcal{J})})$ then this is detected with probability at least $1 - 1/8J!$. We use the same queries for every of the $J!$ (or less) possible permutations of g, and so with probability $\frac{7}{8}$ we will detect ϵ-farness from any such permutation for which it exists. The algorithm accepts the input if there was any permutation of g for which this was not detected. Summing up, an input which is ϵ-far from being any permutation of g will be rejected with probability at least $\frac{5}{8}$, and an input which was a permutation of g will be accepted with probability at least $\frac{6}{8}$ (it could only be rejected if the J-junta test did not detect all the junta coordinates, or if in some stage the dependence of f on $Z_x \cap V_j$ or on $V_j \backslash Z_x$ is not correctly detected).

Reducing the Dependency on J

We construct here a test for f being a permutation of g using $O(\epsilon^{-3}(\log J + \log(\epsilon^{-1}))J^3)$ queries. The running time itself is still exponential in J, however.

First, we perform the J-junta test with the approximation parameter $\frac{\epsilon}{4J}$. We denote $U_{i_1}, \ldots, U_{i_l}$ as before. However, after the size test we again use the independence test to distinguish between $\mathsf{Vr}_{\mathsf{f}}(U_{i_j}) \geq \frac{\epsilon}{2J}$ and $\mathsf{Vr}_{\mathsf{f}}(U_{i_j}) \leq \frac{\epsilon}{4J}$ for every j, and discard from $U_{i_1}, \ldots, U_{i_l}$ also the sets whose variation is low. Let us denote the undiscarded sets by $V_1, \ldots, V_m$. Here we allow also for the possibility that $m < J$, as it could be the case that some sets containing junta coordinates were not detected by the size test or were discarded in the next phase.

Given any function $\tilde{\mathsf{g}}$ on m coordinates, that has the additional property that every coordinate has variation at least $\frac{\epsilon}{4J}$ with respect to it, we perform the following procedure. We choose a uniformly random $x \in \{0,1\}^n$, and denote Z_x as before. For every j we use $12\epsilon^{-1}J\log(hk)$ iterations of the independence test to check with probability at least $1 - \frac{1}{8hk}$ whether $V_j \cap Z_x$ or $V_j \backslash Z_x$ have variation at least $\frac{\epsilon}{4J}$ (if both or none of the two sets has it we reject the input). We then construct y as above and check that $\mathsf{f}(x) = \tilde{\mathsf{g}}(y)$.

We perform $h = 32J^2\epsilon^{-2}(4 + J\ln(J+1))$ iterations of the above; we distinguish with probability $1 - 1/8(J+1)!$ between the case that the probability of $\mathsf{f}(x) = \tilde{\mathsf{g}}(y)$ is at least $1 - \frac{1}{4}\epsilon$, and the case that it is at most $1 - \frac{1}{2}\epsilon$ (see [AS00, Appendix A] for the large deviation inequality used here).

In order to accommodate g, we consider every $\tilde{\mathsf{g}}$ which is a restriction of any permutation of g to any subset of its coordinates that includes all those that have variation at least $\frac{\epsilon}{2J}$, and may include any coordinates with variation at least $\frac{\epsilon}{4J}$ as well. With probability at least $\frac{5}{8}$ the above will give the correct answer for all the possible $\tilde{\mathsf{g}}$; we accept the input if at least one of them has the higher probability for $\mathsf{f}(x) = \tilde{\mathsf{g}}(y)$.

Part III

Noise-Resistant Boolean Functions

are

Juntas

Chapter 12

Introduction to Part III

This part of the thesis deals with boolean functions of the form $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1, 1\}$, where $\mathcal{P}([n])$ denotes the power-set of $[n] \doteq \{1, 2, \ldots, n\}$. We regard functions f of this form as boolean functions over n boolean variables (the value of the i'th variable determines whether the argument of f contains i or not). We focus on a specific type of such functions: A boolean function, f, is said to be a J-junta if there is a set of at most J coordinates $\mathcal{J} \subseteq [n]$, such that the value of f on an input $x \in \mathcal{P}([n])$ depends only on $x \cap \mathcal{J}$, that is, the values of f are determined only by the coordinates in $\mathcal{J}$. Somewhat abusing notation, the set $\mathcal{J}$ is sometimes referred to as the junta which dominates f.

The term junta originates from considerations related to social choice, where a boolean function f is seen as an election scheme, and each coordinate in its domain corresponds to a voter. An element $x \in \mathcal{P}([n])$ represents an election where $i \in x$ if the i'th voter votes 'yes' and $i \notin x$ otherwise, and $\mathsf{f}(x)$ is the outcome of the election. In these terms, a J-junta is an election scheme the outcome of which is completely determined by at most J voters. It is now obvious why 1-juntas are called dictatorships.

However, monotone-increasing dictatorships can be also seen as words in a code known as the *long-code* [BGS98]. Specifically, the truth table of a boolean function f is a legal long-code word, if f depends on one coordinate $i \in [n]$, and returns -1 for an input x if $i \in x$ and 1 otherwise. The decoding of a given long-code f, is the coordinate i which determines its value.

The notions mentioned above, and their interplay, come up in a wide variety of research areas, ranging from hardness of approximation [Hås99, Hås97, DS02, Kho02], to learning theory, property testing [FKR$^+$], theory of social choice [Kal01] and more [BKS99]. In many such applications, one is given a long-code word, and needs to verify that this word is close to a legal codeword. This is done by a test, or a construction, that rules out words which do not satisfy some simple criterion. It is therefore of great importance to find the simplest and weakest possible conditions over boolean functions, which ensure that they are decodable.

The decoding of a long-code word f is the coordinate i on which f depends. However in many cases it suffices to ensure that f 'depends significantly' on some small number of coordinates ([Hås99, Hås97]). In other cases ([DS02]), one needs to ensure that f is 'almost-completely determined' by a constant number J of its coordinates. In this case, the elements in the set $\mathcal{J}$ of coordinates that determine f can be seen as a list of possible decodings for f, since only encoding of such elements may have significant correlation with f.

(ϵ, J)-juntas. The sense in which f should be 'almost-completely determined' by the coordinates in $\mathcal{J}$ needs some attention. A more precise statement would be that there exists a function f' which only depends on coordinates in $\mathcal{J}$, such that for an input x, chosen according to a certain distribution μ, $\mathsf{f}(x)$ equals $\mathsf{f}'(x)$ with probability at least $1 - \epsilon$. In that case we say that f is an $(\epsilon, |\mathcal{J}|)$-junta according to the distribution μ.

For a fixed distribution μ, we are therefore interested in finding a weak-as-possible condition that is satisfied by any long-code word, and such that any other boolean function f satisfying it must be an (ϵ, J)-junta, for some small ϵ and a constant J. The distribution μ may vary in different applications – for instance, in showing that it is hard to approximate the minimal vertex-cover in a given graph, [DS02] use a distribution where each coordinate $i \in [n]$ is independently chosen to belong to x with some probability p. It is crucial for their proof that p is well separated from $1/2$. This so called p-biased distribution (namely the product distribution of p-biased coins), which is the focus of this part of the thesis , seems to have many more applications.

Noise sensitivity. The property studied herein is that of noise-sensitivity. The noise-sensitivity of a boolean function f is the probability that when some noise is applied to a random input x, obtaining another input y, f yields different values for x and y, namely $\mathsf{f}(x) \neq \mathsf{f}(y)$. We show that, for appropriate choice of parameters, (ϵ, J)-juntas are the only boolean functions whose noise-sensitivity is smaller than some threshold. In term of social choice, this means that juntas are the only noise-resistant election schemes.

Related Work

Average-sensitivity

Let $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1, 1\}$ be a boolean function. The influence of a coordinate $i \in [n]$ on f with respect to the distribution μ_p , is defined by

$$\mathsf{influence}_i(\mathsf{f}) \doteq \Pr_{x \sim \mu_p^{[n]}} [\mathsf{f}(x \setminus \{i\}) \neq \mathsf{f}(x \cup \{i\})]$$

The influence of i on f is therefore the probability that flipping it in a random input x changes the value of f. The sum of influences of all coordinates $i \in [n]$, is called the average-sensitivity of f. Equivalently, it is the expected number of coordinates, for a random input x, flipping of which changes the value of f. If f is a long-code word, its average-sensitivity equals 1, and if f is a J-junta then its average-sensitivity is obviously bounded by J.

In [Fri98] it was shown that having small average sensitivity is a sufficient condition for a function f to be a junta. More precisely, it was shown there that if the average-sensitivity of f equals m, then f is an $(\epsilon, c^{m/\epsilon})$-junta with respect to μ_p^n, where $c = c_p$ is some global constant. This result is the one utilized in [DS02] to show hardness-of-approximation for the Vertex-Cover problem, utilizing a distribution bias p in the vicinity of 1/3.

The condition of having small average-sensitivity, though sufficient, does not give a good characterization of functions which are close to being juntas. This may be observed by taking f' to be any J-junta, and then randomly flipping an ϵ-fraction of the values of f', obtaining f. It it almost surely the case that f will be a $(2\epsilon, J)$-junta, however its average sensitivity will be linear in n.

Noise-sensitivity

To obtain a better characterization of juntas, we observe how the value of f changes when the input is changed not on one, but on some small fraction of the coordinates. The λ-noise sensitivity of a boolean function f with respect to the distribution $\mu_P^{[n]}$, is defined to be the probability the value of f changes when noise is applied to a random input x. The noise is applied by first selecting a random set I of coordinates, taking each coordinate into I with probability λ, and then randomly resetting each coordinate in I. That is

$$\mathrm{NS}_{\lambda,p}(\mathsf{f}) \doteq \Pr_{x\sim\mu_p^{[n]},\ I\sim\mu_\lambda^{[n]},\ z\sim\mu_p^I}\left[\mathsf{f}(x) \neq \mathsf{f}\Big((x \setminus I) \cup z\Big)\right]$$

The condition of having small noise sensitivity is therefore very efficiently testable by a perturbation test, which randomly selects x and y as above and tests whether $\mathsf{f}(x) = \mathsf{f}(y)$. Such a perturbation test was embedded into the long-code test, used in [Hås97] for showing hardness of approximating the number of satisfiable linear-equations, with at most 3 variables in each equation, over $\mathbb{Z}_2$. The embedding of the perturbation test helped the test of [Hås97] rule out functions which do not have significant correlation with any long-code word. Such considerations also appear in other hardness results.

It is natural to ask (and was indeed asked by Hastad) if having small noise-sensitivity is in itself a sufficient condition for a function to be close to a junta. It follows from the recent result in [Bou01] that if the λ-noise-sensitivity of f with respect to $\mu_{1/2}$ is sufficiently small, than f is indeed close to a junta.

Fourier representation and noise. One can consider a boolean function f as an element in the space of real-valued functions. As such it has a Fourier representation, as a linear combination of Walsh-products. It is easily verified that in order for f to have small λ-noise sensitivity with respect to $\mu_{1/2}$, its weight on Walsh-products of high frequency must be small. The result actually stated in [Bou01] is that any function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ whose weight on high-frequencies is small enough must be close to a J-junta, for some constant J, namely

Theorem [Bou01]. *Let $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ be a boolean function, and let k be a positive integer and $\epsilon > 0$ any fixed constant. Then for every $\eta > 0$ there is a constant c_η, such that if*

$$\sum_{|S|>k} \left|\widehat{\mathsf{f}}(S)\right|^2 < c_\eta k^{-\frac{1}{2}-\eta}$$

then f is an $(\epsilon, k10^k)$-junta with respect to $\mu_{1/2}$.

Our Results

In this part of the thesis we extend the result of [Bou01] to the case of the p-biased measure μ_p. For a bias p, we deal with the expansion of the boolean function f as a linear combination of p-biased Walsh-products (the biased-measure analogue of the Fourier-expansion of f, as defined in [Tal94]). It states that if f has very small weight on 'large' Walsh-products, then it must be close to a junta. Using $\widehat{\mathsf{f}}(S)$ to denote the coefficient of the biased Walsh-product corresponding to $S \subset [n]$, we have

Theorem 12.1. *Fix a positive integer ℓ. Then there exists a constant $\delta > 0$, such that for every $\epsilon > 0$ and every boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ satisfying*

$$\sum_{|S|>k} \left|\widehat{\mathsf{f}}(S)\right|^2 \le \left(\frac{\epsilon}{k}\right)^{(\ell+1)/\ell}$$

is an $(O(\epsilon), J)$-junta[*], *where*

$$J = O\left(\delta^{-4k}\epsilon^{(\ell+1)/\ell}k^{(2\ell+1)/\ell}\right)$$

The parameters obtained by the above theorem are different than those in [Bou01]. The main difference, is the dependence on k of the threshold on $\sum_{|S|>k}\left|\widehat{\mathsf{f}}(S)\right|^2$, beyond which f is ensured to be close to a junta of constant size. Here, it is required that this weight be bounded by $O(k^{-(\ell+1)/\ell})$, while the result in [Bou01] requires only a bound of order $K^{-1/2-\eta}$

[*] The O notation here hides constants which are independent of ϵ, k, and n. However they may depend on the bias, p, and on ℓ.

for every constant $\eta > 0$. Our proof of Theorem 14.1 is, however, different than that of Bourgain, and its technique may of of independent importance.

Note that we cannot obtain a result for the biased case by applying the same technique used in [Bou01], since it uses an inequality concerning the norm of the Rademacher projection, which we cannot reproduce for the biased case. Once this fact is extended to the biased case, it seems that the result in [Bou01] would easily extend to the biased case as well.

As a direct corollary of Theorem 12.1, we have that juntas are the only noise-resistant boolean functions, that is, a boolean function f whose λ-noise-sensitivity with respect to μ_p is small, must be ϵ-close to a junta of size independent of n.

Corollary 12.2. *There exists a constant $\delta > 0$ for which the following holds. For any integer ℓ and any parameter $\lambda > 0$, fix $k = \log_{(1-\lambda)}(1/2)$. Then every boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1, 1\}$ whose λ-noise-sensitivity with respect to μ^n is at most $(\epsilon/k)^{(\ell+1)/\ell}$, is an $[O(\epsilon), J]$-junta with*

$$J = O\big(\delta^{-4k}\epsilon^{(\ell+1)/\ell}k^{(2\ell+1)/\ell}\big)$$

Asymptotic behavior of the distance from a junta

In light of the above discussion, it seems natural to ask what is the best bound on the distance of a boolean function f from a junta, given that its weight on large biased Walsh-products is small. We give such a bound, which is optimal up to a constant factor, in the case where the weight of f on Walsh-products larger than k is asymptotically small: the following theorem takes effect when this weight is smaller than some negative exponent in k.

Theorem 12.3. *Let $k > 0$ be a fixed integer. Let $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1, 1\}$ be a boolean function and let $\epsilon \doteq \sum_{|S|>k} \big|\widehat{\mathsf{f}}(S)\big|^2$. Then f is an $((1 + o(1))\epsilon, J)$-junta, where J is a constant which depends only on k.*

Theorem 12.3 was proven in [FKN01] for the case $k = 1$, and with respect to the uniform measure $\mu_{1/2}$. We prove Theorem 12.3 by first giving an alternative proof for the case $k = 1$, which is valid with respect to every μ_p, and then extending it to the case $k > 1$.

Structure of this part

In Chapter 13 we formally define the p-biased measure, and present the biased Walsh-products which replace the usual Fourier-basis. In this chapter we also define basic notions such as restrictions, variations, and the noise-sensitivity of a function f, and show their connections with the expansion of f as a combination of biased Walsh-products. In addition we state in Chapter 13 a biased version for the Beckner-Bonami hyper-contractive estimate. The proof of this estimate is deferred to Chapter 18.

In Chapter 14 we give a simple conceptual proof of Theorem 14.1, which contains the main argument of the proof of Theorem 12.1, but achieves somewhat worse parameters. In Chapter 15 we show an alternative proof for the theorem of [FKN01], namely the case $k = 1$ in Theorem 12.3. This proof not only extends to the case of biased-measure, but is also extendable to the case $k > 1$, thus obtaining Theorem 12.3. This is shown in Chapter 16. Finally, in Chapter 17 we prove Theorem 12.1 simply by plugging the result of Theorem 12.3 into the proof of Theorem 14.1. Corollary 12.2 is easily obtained from Theorem 12.1 in the same chapter.

Chapter 13

Preliminaries

We begin by formally defining the product-measure with which we work. For a parameter $q,\ 0 < q < 1$ and a finite set I, define a probability measure μ_q^I on $\mathcal{P}(I)$ by

$$\mu_q^I(A) \doteq (1-q)^{|I|}\left(\frac{q}{1-q}\right)^{|A|}$$

Now fix a bias $p,\ 0 < p < 1$, which is to remain fixed throughout this part. When referring to the p-biased measure, we shall denote μ^I instead of μ_p^I for shortness. We also abbreviate μ_q^n for $\mu_q^{[n]}$.

Let us now formally define a J-junta, and when a boolean function is said to be ϵ-close to one.

Definition 13.1 (junta). *A boolean function* $\mathsf{f}\colon \mathcal{P}(n) \to \{-1,1\}$ *is a* J-junta *if there exists a set* $\mathcal{J} \subset [n]$ *of size* $|\mathcal{J}| \leq J$ *such that*

$$\forall x \quad \mathsf{f}(x) = \mathsf{f}(x \cap \mathcal{J})$$

Definition 13.2 ($[\epsilon, J]$-junta). *A boolean function* f *is an* $[\epsilon, J]$-junta *if there exists a* J*-junta* f' *that disagrees with* f *on at most* ϵ*-fraction of the points, namely*

$$\Pr_{x\sim\mu_p^n}\left[\mathsf{f}(x) \neq \mathsf{f}'(x)\right] \leq \epsilon.$$

Discrete Fourier Expansion

It has shown to be useful to treat boolean functions as elements of the space of real-valued functions $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$. This allows the introduction of powerful tools into the study of boolean functions, such as the discrete Fourier transform. We next define the basic notions we need concerning the space of real-valued functions over $\mathcal{P}([n])$.

Inner-product. The biased inner-product of two real-valued functions f, g over $\mathcal{P}([n])$ is defined by

$$\langle \mathsf{f}, \mathsf{g} \rangle \doteq \mathop{\mathbb{E}}_{x \sim \mu^n} [\mathsf{f}(x)\mathsf{g}(x)]$$

Norm. For every $1 \leq q < \infty$, we define the q-norm of a function $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$ by

$$\|\mathsf{f}\|_q \doteq \left(\mathop{\mathbb{E}}_{x \sim \mu^n} [|\mathsf{f}(x)|^q] \right)^{1/q}$$

Fourier basis. The usual Fourier basis for the space of functions $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1, 1\}$ is not orthonormal (or even orthogonal) with respect to the biased inner-product. Following [Tal94], we thus define the following basis which is orthonormal with respect to the biased inner-product. When $p = 1/2$, this is the usual Walsh/Fourier basis.

Definition 13.3 (biased Walsh-Products). *For every $i \in [n]$, we define the i'th biased Rademacher function $\chi_{\{i\}}\colon \mathcal{P}([n]) \to \mathbb{R}$ by*

$$\chi_{\{i\}}(x) \doteq \begin{cases} \sqrt{p/(1-p)} & i \notin x \\ \\ -\sqrt{(1-p)/p} & i \in x \end{cases}$$

For every set $S \subseteq [n]$, the biased Walsh-product that corresponds to it is then defined by $\chi_S \doteq \prod_{i \in S} \chi_{\{i\}}$. The cardinality of S is called the size or the frequency of χ_S.

Projections

An important aspect of the Fourier representation is that it enables the definition of simple orthonormal projections of f. The projection of f onto a set Ω of Walsh-products is obtained by restricting the Fourier expansion of f to Walsh-products from Ω, namely

$$\mathsf{f}|_\Omega = \sum_{\chi_S \in \Omega} \widehat{\mathsf{f}}(S)\chi_S$$

The weight of f on Ω is defined to be

$$\| \, \mathsf{f}|_\Omega \, \|_2^2 = \sum_{\chi_S \in \Omega} \widehat{\mathsf{f}}^2(S)$$

Next let us define two such projections which are crucial for our analysis, dividing the Walsh-products into low-frequencies and high-frequencies.

Definition 13.4 (frequency separation). *For a given* $1 \leq k \leq n$ *and a function* f *denote*

$$\mathsf{f}^{\leq k} = \sum_{|S|\leq k} \widehat{\mathsf{f}}(S)\chi_S \ , \quad \textit{and} \qquad \mathsf{f}^{>k} = \sum_{|S|>k} \widehat{\mathsf{f}}(S)\chi_S$$

The averaging projection. Let $I \subseteq [n]$ be a set of indices. For a function f over $\mathcal{P}([n])$, consider the function obtained from it by averaging, for each element $x \in \mathcal{P}([n])$, over all distinct input settings of the form $(x \setminus I) \cup z$ where $z \subseteq I$ – this is a non boolean function that depends only on $\bar{I} = [n] \setminus I$. This function, denoted by $\mathsf{Avg}_I[\mathsf{f}] : \mathcal{P}(\bar{I}) \to \mathbb{R}$, is formally defined by

$$\mathsf{Avg}_I[\mathsf{f}](x) \doteq \mathop{\mathbb{E}}_{z\sim\mu^I}[\mathsf{f}((x \setminus I) \cup z)]$$

One observes that Avg_I can also be written as the projection onto the set of Walsh-products whose support is disjoint from I, namely

$$\mathsf{Avg}_I[\mathsf{f}] = \sum_{S\cap I=\emptyset} \widehat{\mathsf{f}}(S)\chi_S$$

The following proposition connects the 1-norm, the 2-norm, and the average function.

Proposition 13.5. *Let* $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ *be a boolean function, and let* $I \subseteq [n]$*. Then*

$$\|\mathsf{f} - \mathsf{Avg}_I[\mathsf{f}]\|_2^2 = \|\mathsf{f} - \mathsf{Avg}_I[\mathsf{f}]\|_1$$

Proof. For every $y \in \mathcal{P}(\bar{I})$ (where $\bar{I} \doteq [n] \setminus I$), define

$$\alpha(y) \doteq \mathop{\Pr}_{z\sim\mu^I}[\mathsf{f}(y \cup z) = 1]$$

Then for every $x \in \mathcal{P}([n])$,

$$\mathsf{Avg}_I[\mathsf{f}](x) = 2\alpha(x \setminus I) - 1$$

Hence

$$\begin{aligned}
\|\mathsf{f} - \mathsf{Avg}_I[\mathsf{f}]\|_2 &= \mathop{\mathbb{E}}_{x\sim\mu^n}\left[(\mathsf{f}(x) - \mathsf{Avg}_I[\mathsf{f}](x))^2\right] = \\
&= \mathop{\mathbb{E}}_{y\sim\mu^{\bar{I}}}\mathop{\mathbb{E}}_{z\sim\mu^I}\left[\Big(\mathsf{f}(y \cup z) - \mathsf{Avg}_I[\mathsf{f}](y \cup z)\Big)^2\right] = \\
&= \mathop{\mathbb{E}}_{y\sim\mu^{\bar{I}}}\left[\alpha(y)(2 - 2\alpha(y))^2 + (1 - \alpha(y)) \cdot (2\alpha(y))^2\right] = \\
&= \mathop{\mathbb{E}}_{y\sim\mu^{\bar{I}}}[4\alpha(y)(1 - \alpha(y))] = \\
&= \mathop{\mathbb{E}}_{y\sim\mu^{\bar{I}}}\mathop{\mathbb{E}}_{z\sim\mu^I}[|\mathsf{f}(y \cup z) - \mathsf{Avg}_I[\mathsf{f}](y \cup z)|] = \|\mathsf{f} - \mathsf{Avg}_I[\mathsf{f}]\|_1
\end{aligned}$$

■

Variations

We now define the variation of a boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ on a subset of the coordinates $I \subseteq [n]$. The variation of f on a singleton $\{i\}$ coincides with the classical definition ([BL89, KKL88]) of the influence of the i'th coordinate on f (up to a fixed constant factor, if $p \neq 1/2$). The variation of f on I is twice the probability that f yields different values, given two random inputs that agree on all the coordinates outside I, that is

$$\mathsf{Vr}_{\mathsf{f}}(I) = 2 \Pr_{\substack{y \sim \mu^{[n] \setminus I} \\ z_1, z_2 \sim \mu^I}} [\mathsf{f}(y \cup z_1) \neq \mathsf{f}(y \cup z_2)]$$

Following is a definition of the variation of f on I which coincides with the one above for boolean functions f, but also extends to the case of real-valued functions (the fact that the two characterizations of the variation coincide for boolean functions is proven in Proposition 8.11 on page 101).

Definition 13.6 (variation). *The* variation *of a function* $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$ *on a set* $I \subseteq [n]$ *of coordinates is defined by*

$$\mathsf{Vr}_{\mathsf{f}}(I) \doteq \|\mathsf{f} - \mathsf{Avg}_I[\mathsf{f}]\|_2^2 = \sum_{S \cup I \neq \emptyset} \widehat{\mathsf{f}}^2(S)$$

A useful property of variations, which is obvious from the definition, is their sub-additivity.

Claim 13.7 (sub-additivity). *Let* $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$*, and let* $I_1, I_2 \subseteq n$*. Then*

$$\mathsf{Vr}_{\mathsf{f}}(I_1 \cup I_2) \leq \mathsf{Vr}_{\mathsf{f}}(I_1) + \mathsf{Vr}_{\mathsf{f}}(I_2)$$

Another very important property of variations, is that if a boolean function f has small variation on a set of coordinates, then it is almost independent of these coordinates. Putting it differently, if the variation of f on the complement of a set I is small, than f is close to some boolean function g which depends only on the coordinates in I.

Proposition 13.8. *Let* $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ *be a boolean function, and let* $I \subseteq [n]$ *be a set of coordinates satisfying* $\mathsf{Vr}_{\mathsf{f}}(\bar{I}) < 2\epsilon$*, where* $\bar{I} \doteq [n] \setminus I$*. Then there exists a noolean function* $\mathsf{g}\colon \mathcal{P}([n]) \to \{-1,1\}$ *which depends only on coordinates from* I*, and satisfies*

$$\Pr_{x \sim \mu^n} [\mathsf{f}(x) \neq \mathsf{g}(x)] < \epsilon$$

Proof. Let $\mathsf{g} \doteq \operatorname{sign}(\mathsf{Avg}_{\bar{I}}[\mathsf{f}])$ (we arbitrarily set $\operatorname{sign}(0) \doteq 1$). Then g depends only on coordinates from I. Let us show that $\mathsf{f}(x) = \mathsf{g}(x)$ for most x's.

For $y \in \mathcal{P}(I)$, denote

$$\alpha(y) \doteq \Pr_{z \sim \mu^{\bar{I}}} [\mathsf{f}(y \cup z) \neq \mathsf{g}(y \cup z)]$$

and note that $\alpha(y) \leq 1/2$ for all y. Therefore, we have

$$\Pr_{x \sim \mu^n}[\mathsf{f}(x) \neq \mathsf{g}(x)] = \mathop{\mathbb{E}}_{y \sim \mu^I}[\alpha(y)] \leq \mathop{\mathbb{E}}_{y \sim \mu^I}[2\alpha(y)(1-\alpha(y))] =$$
$$= \Pr_{\substack{y \sim \mu^I \\ z_1, z_2 \sim \mu^{\bar{I}}}}[\mathsf{f}(y \cup z_1) \neq \mathsf{f}(y \cup z_2)] = \frac{1}{2}\mathsf{Vr}_{\mathsf{f}}(\bar{I}) < \epsilon$$

■

Combining the above two ways to project a function – one according to frequency and the other according to non-emptiness of the intersection with some given subset of the indices – we can define:

Definition 13.9 (low-frequency variation). *The* k-low-frequency variation *on a subset* $I \subseteq [n]$ *of* f*, is*

$$\mathsf{Vr}_{\mathsf{f}}^{\leq k}(I) \doteq \mathsf{Vr}_{\mathsf{f}^{\leq k}}(I) = \sum_{\substack{S \cap I \neq \emptyset \\ |S| \leq k}} \widehat{\mathsf{f}}^2(S)$$

For shortness, denote $\mathsf{Vr}_{\mathsf{f}}^{\leq k}(i) \doteq \mathsf{Vr}_{\mathsf{f}}^{\leq k}(\{i\})$.

Restrictions

For a given set of coordinates $I \subseteq [n]$ and every $x \in \mathcal{P}(\bar{I})$, let us denote by $\mathsf{f}_I[x] \colon \mathcal{P}(I) \to \{-1, 1\}$ the boolean function defined by

$$\mathsf{f}_I[x](y) \doteq \mathsf{f}(x \cup y)$$

The Fourier expansion of $\mathsf{f}_I[x]$ can be deduced from the Fourier expansion of f. For every $S \subseteq I$, it is easily seen that

$$\widehat{\mathsf{f}_I[x]}(S) = \sum_{\substack{T \subseteq [n] \\ T \cap I = S}} \widehat{\mathsf{f}}(T) \chi_{T \setminus S}(x)$$

where $\chi_{T \setminus S}$ can in fact be replaced by χ_T.

Note that the variation on a subset I of a function f can be expressed in terms of restrictions. It is the expected variance of $\mathsf{f}_I[x]$, over all input settings $x \sim \mu^{\bar{I}}$ outside I. This leads immediately to the following claim.

Claim 13.10. *Let* f *be a boolean function, and let* $I \subseteq [n]$*. Then*

$$\mathsf{Vr}_{\mathsf{f}}(I) = \mathop{\mathbb{E}}_{x \sim \mu^{\bar{I}}}\left[\mathsf{Vr}_{\mathsf{f}_I[x]}(I)\right]$$

Noise-Sensitivity

As already mentioned in the introduction, the λ-noise-sensitivity of a boolean function $\mathsf{f}\colon [n] \to \{-1,1\}$ (with respect to μ_p) is defined by

$$\mathrm{NS}_{\lambda,p}(\mathsf{f}) \doteq \Pr_{x\sim\mu^n\,,\; I\sim\mu^n_\lambda\,,\; z\sim\mu^I}\left[\mathsf{f}(x) \neq \mathsf{f}\Big((x \setminus I) \cup z\Big)\right]$$

The following proposition shows how the λ-noise-sensitivity of f is connected with its Fourier-expansion.

Proposition 13.11. *Let $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ be a boolean function. Then for every parameter λ,*

$$\mathrm{NS}_{\lambda,p}(\mathsf{f}) = \frac{1}{2} - \frac{1}{2}\sum_S (1-\lambda)^{|S|}\widehat{\mathsf{f}}(S)^2$$

Proof. We omit the simple proof ■

Bonami-Beckner Inequality

We define for every $0 \le \delta \le 1$ an operator T_δ over real-valued functions $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$. At each point x, $T_\delta[\mathsf{f}](x)$ is the expected value of f when a $(1-\delta)$-noise is applied to x, that is

$$T_\delta[\mathsf{f}](x) = \mathop{\mathbb{E}}_{I\sim\mu^n_{(1-\delta)}\,,\; z\sim\mu^I}[\mathsf{f}((x \setminus I) \cup z)] = \sum_S \delta^{|S|}\widehat{\mathsf{f}}(x)\chi_S(x)$$

Bonami and Beckner proved that, in the uniform case, T_δ is hyper-contractive for appropriate values of δ:

Theorem 13.12. *Let $q \ge r \ge 1$, and let $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$. Then in the uniform case, namely when the norms are taken with respect to $\mu^n_{1/2}$,*

$$\|\, T_\delta[\mathsf{f}]\,\|_q \le \|\mathsf{f}\|_r \quad \textit{for any} \quad \delta \le \sqrt{(r-1)/(q-1)}\,.$$

In [Fri98], a special case of Theorem 13.12 was shown to hold for the biased case as well. We prove another special case of this theorem, which is needed for our purposes.

Theorem 13.13. *There exists a constant $\delta = \delta_p > 0$, such that for every $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$,*

$$\|\, T_\delta[\mathsf{f}]\,\|_4 \le \|\mathsf{f}\|_2$$

where the norms are taken with respect to μ^n_p.

This theorem is proven in Chapter 18. Note that in the sequel all the parameters denoted δ refer, unless noted otherwise, to the parameter δ for which Theorem 13.13 holds. The best δ for which the theorem holds was, in fact, found recently by K. Oleszkiewicz [Ole02] to be $\delta_p = (1 + p^{-1/2}(1-p)^{-1/2})^{-1/2}$, which is of order $p^{1/4}$ for small values of p.

The analysis herein, as well as many other Fourier analytic results concerning boolean functions, utilizes the following simple corollary of this hyper-contractivity estimate.

Corollary 13.14. *There exists a constant $\delta > 0$ such that any function $\mathsf{g}\colon \mathcal{P}(n) \to \mathbb{R}$ for which $\mathsf{g}^{>k} = 0$, satisfies*

$$\|\mathsf{g}\|_4 \leq \delta^{-k} \|\mathsf{g}\|_2$$

Proof. Take δ to be as in Theorem 13.13, and apply T_δ to $\mathsf{f} \doteq (T_\delta)^{-1}(\mathsf{g})$. The corollary now follows from Theorem 13.13 by applying Parseval's identity to f. ■

Chapter 14

The Main Argument

The main result of this chapter shows that a function f whose weight is concentrated on low-frequencies is close to a junta. The parameters it achieves can be improved, though. In Chapter 17 the parameters are indeed improved by essentially repeating the same proof, and plugging in the parameters obtained from Theorem 16.1, proven in Chapter 16.

Theorem 14.1. *There exists a constant $\delta > 0$, such that every boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ satisfying $\|\mathsf{f}^{>k}\|_2^2 \le (\epsilon/k)^2$, is an $[O(\epsilon), J]$-junta for $J = O(\epsilon^{-2}k^3\delta^{-4k})$.*

Note that the O-notation here, and throughout this part, hides constants which are independent of ϵ, k, and n, but may depend on the bias, p.

Theorem 14.1 easily implies Corollary 14.2 below, showing that a function whose λ-noise-sensitivity is small must be close to a junta.

Corollary 14.2. *There exists a constant $\delta > 0$ with the following property. For any parameter $\lambda > 0$, fix $k = \log_{(1-\lambda)}(1/2)$. Then every boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ whose λ-noise-sensitivity with respect to μ_p^n is bounded by $(\epsilon/k)^2$, is an $[O(\epsilon), J]$-junta, where*

$$J = O(\epsilon^{-2}k^3\delta^{-2k})$$

Proof. Let f be a boolean function as stated in Corollary 14.2. Then according to proposition 13.11,

$$\mathrm{NS}_{\lambda,p}(\mathsf{f}) = (\epsilon/k)^2 \ge \frac{1}{2} - \frac{1}{2}\sum_S (1-\lambda)^{|S|}\widehat{\mathsf{f}}(S)^2 \ge \frac{1}{2} - \frac{1}{2}\left(\sum_{|S|\le k}\widehat{\mathsf{f}}(S)^2 + \frac{1}{2}\sum_{|S|>k}\widehat{\mathsf{f}}(S)^2\right)$$

Since f is boolean, we have $\sum_S \widehat{\mathsf{f}}(S)^2 = \|\mathsf{f}\|_2^2 = 1$, hence we obtain from the above inequality that

$$\sum_{|S|>k}\widehat{\mathsf{f}}(S)^2 \le 4(\epsilon/k)^2$$

Corollary 14.2 now follows from Theorem 14.1. ■

Proof of Theorem 14.1

We now move to the proof of Theorem 14.1. In fact, we prove the following theorem, which yeilds Theorem 14.1 by setting $r \doteq k^2/\epsilon$ and $\tau \doteq \delta^{4k}\epsilon^2/k^2$.

Theorem 14.3. *There exist global positive constants C and δ, such that for any $\tau > 0$, and for any positive integers k and $r > 2k$, the following holds. For every boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$, f is an $[\gamma, J]$-junta with respect to μ^n, where $J = k/\tau$ and*

$$\gamma = Cr\Big(\delta^{-4k}\tau + k^2/r^2 + \|\mathsf{f}^{>k}\|_2^2\Big)$$

Proof. First, we specify a set of coordinates that will be shown to determine most of the values of f. These are the coordinates on which f has non-negligible k-variation.

Definition 14.4. *Let*

$$\mathcal{J} \doteq \Big\{ i \in [n] \mid \mathsf{Vr}_{\mathsf{f}}^{\leq k}(i) \geq \tau \Big\}$$

Also, denote $\bar{\mathcal{J}} \doteq [n] \setminus \mathcal{J}$.

The following is a simple observation.

Claim 14.5. *$|\mathcal{J}| \leq k/\tau$.* ■

To prove that f is indeed close to a junta dominated by $\mathcal{J}$, let us randomly partition the coordinates in $\bar{\mathcal{J}}$ into r subsets $I_1, \ldots, I_r$. Lemma 14.6 below, which is proven in the next section, states that for every fixed h, the expectation of $\mathsf{Vr}_{\mathsf{f}}(I_h)$ is very small (note that for every h, I_h is a random subset of $\bar{\mathcal{J}}$, distributed according to $\mu_{1/r}^{\bar{\mathcal{J}}}$). In light of Proposition 13.8, this means that when a partition of $\bar{\mathcal{J}}$ is chosen randomly, it is expected that f be almost independent of each subset in the partition.

Lemma 14.6. *There exists a global constant C and a global positive constant $\delta > 0$, such that for any $r > 2k$,*

$$\mathop{\mathbb{E}}_{I \sim \mu_{1/r}^{\bar{\mathcal{J}}}} [\mathsf{Vr}_{\mathsf{f}}(I)] \leq C\Big(\delta^{-4k}\tau + k^2/r^2 + \|\mathsf{f}^{>k}\|_2^2\Big)$$

The linearity of expectation and the sub-additivity of the variation, now imply that in fact f is almost independent of all the coordinates in $\bar{\mathcal{J}}$, that is

$$\mathsf{Vr}_{\mathsf{f}}(\bar{\mathcal{J}}) \leq \mathbb{E}\left[\sum_{h=1}^{r} \mathsf{Vr}_{\mathsf{f}}(I_h)\right] \leq Cr\Big(\delta^{-4k}\tau + k^2/r^2 + \|\mathsf{f}^{>k}\|_2^2\Big)$$

From Proposition 13.8 we thus obtain that f is a $\Big[\frac{1}{2}Cr\Big(\delta^{-4k}\tau + k^2/r^2 + \|\mathsf{f}^{>k}\|_2^2\Big), |\mathcal{J}|\Big]$-junta. This completes the proof, since $|\mathcal{J}| \leq k/\tau$. ■

14.1 Random Subsets of $\bar{\mathcal{J}}$ Have Small Variation

The proof of Lemma 14.6 proceeds by showing that on a random I chosen according to $\mu_{1/r}^{\bar{\mathcal{J}}}$, the following holds with high probability: for almost all input settings x outside I the weight of $\mathsf{f}_I[x]$ is concentrated on Walsh-products of size at most 1. Once this is shown, it is possible to apply Theorem 14.7 (stated below) which states that in this case $\mathsf{f}_I[x]$ must be close to either a dictatorship or a constant. We then show that $\mathsf{f}_I[x]$ cannot be close to a non-constant dictatorship too often, hence it is usually almost constant, hence the lemma follows.

When $k = 1$

We start by a theorem that considers the case where a boolean function f has low weight on Walsh-products of size more than one. It shows that in this case f must be close to a dictatorship. This theorem was proven in [FKN01] for the uniform case, and we extend the proof for the biased case as well. Note that the proof of this theorem does not generalize directly to the case of frequencies higher than 1, since it uses the fact that the values of Walsh-products of size one on a random assignment are completely independent. This property does not hold for Walsh-products of larger size.

Theorem 14.7. *There exists a global constant M so that for any boolean function f, there exists a boolean dictatorship g (g may be constant) such that*

$$\|\mathsf{f} - \mathsf{g}\|_2^2 \leq M \cdot \|\mathsf{f}^{>1}\|_2^2$$

Proof. This follows directly from Corollary 15.2 in Chapter 15. ■

We use the following simplified form of Theorem 14.7, which follows immediately from it by computing the Fourier-coefficients of a non-constant dictatorship:

Corollary 14.8. *There exists a global constant M, so that given any boolean function $\mathsf{f}\colon \mathcal{P}(m) \to \{-1, 1\}$, either there exists a coordinate i such that $|\widehat{\mathsf{f}}(\{i\})| > \sqrt{p}$, or $\mathsf{Vr}_{\mathsf{f}}([m]) \leq M\|\mathsf{f}^{>1}\|_2^2$.*

Few Dictatorships

We now return to the proof of Lemma 14.6 which states, in essence, that for most choices of $I \subseteq \bar{\mathcal{J}}$, $\mathsf{f}_I[x]$ is almost constant for most of the x's in $\mathcal{P}(\bar{I})$. We begin by giving an upper-estimate on the (weighted) number of restrictions $\mathsf{f}_I[x]$ that can be close to non-constant dictatorships. Then, in the next subsection, we show that indeed most of these restriction are almost constant.

For a given $I \subseteq [n]$ denote the 'dictatorship set' by

$$\mathcal{D}_I \doteq \left\{ x \in \mathcal{P}(\bar{I}) \;\middle|\; \exists\, i \in I \text{ for which } \left|\widehat{\mathsf{f}_I[x]}(\{i\})\right| > \sqrt{p} \right\}$$

To bound the measure of $\mathcal{D}_I$, we use the fact that the coefficient of χ_i in $\mathsf{f}_I[x]$ is a function of x that is concentrated on low-frequencies, and has small norm (since every $i \in I$ has small variation). We thus use the following lemma, which utilizes Theorem 13.13 to show that such a function cannot often attain large values. We conclude that the coefficient of χ_i is almost never high enough for it to become a dictator.

Lemma 14.9. *Let δ be as in Theorem 13.13, and let $0 < \alpha < \beta$ be any parameters. Then for any function $\mathsf{g}\colon \mathcal{P}(m) \to \mathbb{R}$*

$$\Pr_{x\sim\mu^m}\left[|\mathsf{g}(x)| > \beta\right] \le \alpha^{-4}\delta^{-4k}\|\mathsf{g}^{\le k}\|_2^4 + (\beta-\alpha)^{-2}\|\mathsf{g}^{>k}\|_2^2$$

Proof. We start off with a simple claim concerning the case where only the low frequencies portion of the function is considered.

Claim 14.10. *Let $\mathsf{g}\colon \mathcal{P}(m) \to \mathbb{R}$ be a real-valued function such that $\mathsf{g}^{>k} = 0$, and let $\alpha < 1$ be a positive parameter; then*

$$\Pr_{x\sim\mu^m}\left[|\mathsf{g}(x)| > \alpha\right] \le \alpha^{-4}\delta^{-4k}\|\mathsf{g}\|_2^4$$

Proof. By applying Markov's inequality for $|\mathsf{g}|^4$ and then applying Theorem 13.13, we have

$$\alpha^4 \cdot \Pr_{x\sim\mu^m}\left[|\mathsf{g}(x)| > \alpha\right] \le \|\mathsf{g}\|_4^4 \le \delta^{-4k}\|\mathsf{g}\|_2^4$$

■

Now to prove Lemma 14.9, we break g into its low-frequency and its high-frequency parts. The probability that $|\mathsf{g}(x)|$ evaluates above β by bounding the probability that the low-frequency part evaluates above some $\alpha < \beta$, and the probability of the high-frequency part to evaluate above $\beta - \alpha$ (if those two parts evaluate to a smaller value than prescribed, they can never reach β):

$$\begin{aligned}
\Pr_{x\sim\mu^m}\left[|\mathsf{g}(x)| > \beta\right] &\le \Pr_{x\sim\mu^m}\left[\left|\mathsf{g}^{\le k}(x)\right| > \alpha\right] + \Pr_{x\sim\mu^m}\left[\left|\mathsf{g}^{>k}(x)\right| > \beta-\alpha\right] \le \\
&\le \alpha^{-4}\delta^{-4k}\|\mathsf{g}^{\le k}\|_2^4 + \Pr_{x\sim\mu^m}\left[(\mathsf{g}^{>k}(x))^2 > (\beta-\alpha)^2\right] \le \\
&\le \alpha^{-4}\delta^{-4k}\|\mathsf{g}^{\le k}\|_2^4 + (\beta-\alpha)^{-2}\|\mathsf{g}^{>k}\|_2^2
\end{aligned}$$

■

Now fix $i \in I$ and consider the function $g_i \colon \mathcal{P}(\widehat{I}) \to \mathbb{R}$, which assigns to every x the coefficient of χ_i in f_I. That is,

$$g_i(x) = \widehat{\mathsf{f}_I[x]}(\{i\})$$

For $\mathsf{f}_I[x]$ to be a dictatorship, one of the g_i's must evaluate to at least $\sqrt{p}$ in absolute value. Applying lemma 14.9, with parameters $\alpha = \sqrt{p}/2$ and $\beta = \sqrt{p}$, we get a bound on the probability, for a random x, that $\mathsf{f}_I[x]$ is a dictatorship.

$$\begin{aligned}
\Pr_{x \sim \mu^{\bar{I}}}[x \in \mathcal{D}_I] &\leq \sum_{i \in I} \Pr_{x \sim \mu^{\bar{I}}}[|g_i(x)| > \sqrt{p}] \leq \\
&= 16p^{-2}\delta^{-4k} \sum_{i \in I} \|\mathsf{g}_i^{\leq k}\|_2^4 + \frac{4}{p} \sum_{i \in I} \|\mathsf{g}^{>k}\|_2^2 = \\
&= 16p^{-2}\delta^{-4k} \sum_{i \in I} \Big\| \sum_{\substack{|S| \leq k \\ S \cap I = \{i\}}} \widehat{\mathsf{f}}(S)\chi_S \Big\|_2^4 + \frac{4}{p} \sum_{i \in I} \Big\| \sum_{\substack{|S| > k \\ S \cap I = \{i\}}} \widehat{\mathsf{f}}(S)\chi_S \Big\|_2^2 \leq \\
&\leq 16p^{-2}\delta^{-4k} \sum_{i \in I} \Bigg(\sum_{\substack{|S| \leq k \\ S \cap I = \{i\}}} \widehat{\mathsf{f}}^2(S) \Bigg)^2 + \frac{4}{p} \|\mathsf{f}^{>k}\|_2^2
\end{aligned}$$

Since $\sum_{|S \cap I| = 1} \widehat{\mathsf{f}}^2(S) \leq 1$, it follows that

$$\sum_{i \in I} \Bigg(\sum_{\substack{|S| \leq k \\ S \cap I = \{i\}}} \widehat{\mathsf{f}}^2(S) \Bigg)^2 \leq \max_{i \in I} \sum_{\substack{|S| \leq k \\ S \cap I = \{i\}}} \widehat{\mathsf{f}}^2(S) = \max_{i \in I} \mathsf{Vr}_{\mathsf{f}}^{\leq k}(i) < \tau$$

Altogether this implies that for some constant M_1,

$$\Pr_{x \sim \mu^{\bar{I}}}[x \in \mathcal{D}_I] \leq M_1 \delta^{-4k} \tau + M_1 \|\mathsf{f}^{>k}\|_2^2$$

Restrictions are Mostly Constant

We are now ready to prove that the restrictions $\mathsf{f}_I[x]$ are mostly constant. In fact, we show that the variation of f on I is, with high probability, quite small. First, note that for an x such that $x \notin \mathcal{D}_I$, Corollary 14.8 asserts that

$$\mathsf{Vr}_{\mathsf{f}_I[x]}(I) \leq M \sum_{|R| > 1} \widehat{\mathsf{f}_I[x]}^2(R)$$

and by Claim 13.10 we have that

$$\begin{aligned}
\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}[\mathsf{Vr}_{\mathsf{f}}(I)] = \mathop{\mathbb{E}}_{\substack{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}\\ x\sim\mu^{\bar{I}}}}\left[\mathsf{Vr}_{\mathsf{f}_{I[x]}}(I)\right] \quad \leq \\
\leq \mathop{\Pr}_{\substack{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}\\ x\sim\mu^{\bar{I}}}}[x\in\mathcal{D}_I] + \mathop{\mathbb{E}}_{\substack{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}\\ x\sim\mu^{\bar{I}}}}\left[M\sum_{|R|>1}\widehat{\mathsf{f}_I[x]}^2(R)\right] \quad \leq \\
\leq M_1\delta^{-4k}\tau + M_1\|\mathsf{f}^{>k}\|_2^2 + M\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}\left[\sum_{|S\cap I|>1}\widehat{\mathsf{f}}^2(S)\right] \leq \\
\leq M_1\delta^{-4k}\tau + M_1\|\mathsf{f}^{>k}\|_2^2 + M\|\mathsf{f}^{>k}\|_2^2 + M\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}\left[\sum_{\substack{|S|\leq k\\ |S\cap I|>1}}\widehat{\mathsf{f}}^2(S)\right]
\end{aligned}$$

Now note that
$$\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}\Big[\sum_{\substack{|S|\leq k\\ |S\cap I|>1}}\widehat{\mathsf{f}}^2(S)\Big] \leq \frac{k^2/r^2}{1-k/r} \quad .$$

This follows since the total weight of all Walsh-products is bounded by 1, and since

$$\Pr_I[|S\cap I|>1] \leq \sum_{i=2}^{k}\binom{k}{i}r^{-i}(1-1/r)^{k-i} \leq \sum_{i=2}^{k}k^i r^{-i} \leq \frac{k^2/r^2}{1-k/r}$$

Therefore, we get that overall, the expectation of the variation is bounded by

$$\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}[\mathsf{Vr}_{\mathsf{f}}(I)] \leq M_1\delta^{-4k}\tau + M_1\|\mathsf{f}^{>k}\|_2^2 + M\left(\frac{k^2/r^2}{1-k/r} + \|\mathsf{f}^{>k}\|_2^2\right)$$

This completes the proof of Lemma 14.6.

Chapter 15

Biased FKN

The following theorem shows that in order for a real-valued linear function to be close to boolean, its weight must be concentrated on the constant character and perhaps on one more character of size one. It is therefore close to a 'real-valued dictatorship', a real function which depends on only one of the coordinates. The main result of this chapter, Corollary 15.2, follows it, showing that a boolean function whose weight is concentrated on Walsh-products of size at most 1 is in fact close to a boolean-valued dictatorship.

Theorem 15.1. *Let* $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$ *be a linear real valued function, namely* $\mathsf{f}^{>1} = 0$*. Let* $\epsilon \doteq \| \, |\mathsf{f}| - 1 \, \|_2^2$ *measure the squared distance of* f *from the nearest boolean function. Then, denoting by* i_o *the index such that* $\left|\widehat{\mathsf{f}}(\{i_o\})\right|$ *is maximal, we have*

$$\|\mathsf{f} - \left(\widehat{\mathsf{f}}(\emptyset) + \widehat{\mathsf{f}}(\{i_o\})\chi_{\{i_o\}}\right)\|_2^2 < (1 + o(1))\epsilon$$

Before we prove Theorem 15.1, we state and prove the following corollary, which immediately implies Theorem 14.7.

Corollary 15.2. *Let* $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1, 1\}$ *be a boolean function, and let* $\epsilon \doteq \|\mathsf{f}^{>1}\|_2^2$*. Then* f *is a* $\big((1 + o(1))\epsilon\,,\, 1\big)$*-junta, namely it is* $(1 + o(1))\epsilon$*-close to some boolean dictatorship.*

Proof. Note that $\| \left|\mathsf{f}^{\leq 1}\right| - 1 \, \|_2^2 \leq \|\mathsf{f}^{>1}\|_2^2 = \epsilon$. Hence according to Theorem 15.1, there is some coordinate $i_o \in [n]$ such that

$$\begin{aligned}\mathsf{Vr}_{\mathsf{f}}([n] \setminus \{i_o\}) = \|\mathsf{f} - \mathsf{Avg}_{[n]\setminus\{i_o\}}[\mathsf{f}]\|_2^2 = \|\mathsf{f} - \left(\widehat{\mathsf{f}}(\emptyset) + \widehat{\mathsf{f}}(\{i_o\})\chi_{\{i_o\}}\right)\|_2^2 \leq \\ \leq \epsilon + \|\mathsf{f}^{\leq 1} - \left(\widehat{\mathsf{f}}(\emptyset) + \widehat{\mathsf{f}}(\{i_o\})\chi_{\{i_o\}}\right)\|_2^2 < (2 + o(1))\epsilon\end{aligned}$$

It follows from Proposition 13.8 that there exists a boolean function g that depends only on the coordinate i_o (and is thus a dictatorship), such that

$$\Pr_{x \sim \mu^n}[\mathsf{f}(x) \neq \mathsf{g}(x)] < (1 + o(1))\epsilon$$

Therefore f is a $\big((1 - o(1))\epsilon\,,\, 1\big)$-junta. ■

Proof of Theorem 15.1: For simplicity, we write $\mathsf{f} = a_0 + \sum_{i=1}^n a_i \chi_{\{i\}}$, and assume without loss of generality that $|a_1| \geq |a_2| \geq \ldots \geq |a_n|$. We should thus prove that $\sum_{i=2}^n |a_i|^2 < (1 + o(1))\epsilon$. First, we prove that there cannot be a large coefficient within $a_2, \ldots, a_n$.

Claim 15.3. *For some global constant c_1 (depending only on p) and for all i, $2 \leq i \leq n$, $|a_i| \leq c_1\sqrt{\epsilon}$.*

Proof. The proof is very elementary, and we give only its outline. Suppose that $|a_2|$ (and therefore also $|a_1|$) is larger than specified. Then for every setting $z \in \mathcal{P}(\{3, \ldots, n\})$ of the coordinates $3, \ldots, n$, it is easy to find a setting $y \in \mathcal{P}(\{1, 2\})$ for the first two coordinates such that for $x \doteq y \cup z$

$$\Big(|\,\mathsf{f}(x)\,| - 1\Big)^2 > \frac{\epsilon}{(\min\{p, 1-p\})^2} \tag{15.1}$$

Hence for a random input x, Equation (15.1) holds for x with probability at least $(\min\{p, 1-p\})^2$, and therefore $\|\,|\mathsf{f}| - 1\,\|_2^2 > \epsilon$, a contradiction. ■

According to Claim 15.3, for every $2 \leq i \leq n$, $|a_i|^2 \leq c_1^2\epsilon$. We thus choose $m \in \{2, \ldots, n\}$ to be the minimal index satisfying

$$\sum_{i=m}^n |a_i|^2 \leq (c_1^2 + 2)\epsilon \tag{15.2}$$

Denote $I \doteq \{m, \ldots, n\}$. Then

$$\epsilon \geq \|\,|\mathsf{f}| - 1\,\|_2^2 = \mathop{\mathbb{E}}_{x\sim\mu^n}\left[(|\mathsf{f}(x)| - 1)^2\right] = \mathop{\mathbb{E}}_{y\sim\mu^{\bar{I}}}\left[\mathop{\mathbb{E}}_{z\sim\mu^I}\left[(|\mathsf{f}(y \cup z)| - 1)^2\right]\right] =$$
$$= \mathop{\mathbb{E}}_{y\sim\mu^{\bar{I}}}\left[\|\,|\mathsf{f}_I[y]| - 1\,\|_2^2\right]$$

hence for some $y \in \mathcal{P}(\bar{I})$, $\|\,|\mathsf{f}_I[y]| - 1\,\|_2^2 \leq \epsilon$. Now $\mathsf{f}_I[y]$ has the form

$$\mathsf{f}_I[y] = b + \sum_{i=m}^n a_i\chi_i$$

for some b, and therefore it satisfies the conditions of Theorem 15.1, with the additional property that $\|\mathsf{f}_I^{>0}[y]\|_2^2 \leq (c_1^2 + 2)\epsilon$. We use the following lemma, which deals with such a situation.

Lemma 15.4. *Fix any (large) constant $c > 0$ and let $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$ be a function satisfying $\mathsf{f}^{>1} \equiv 0$. Let $\epsilon \doteq \||\mathsf{f}| - 1\|_2^2$, and suppose further that $\|\mathsf{f}^{>0}\|_2^2 < c\epsilon$. Then it also holds that*

$$\|\mathsf{f}^{>0}\|_2^2 < (1 + o(1))\epsilon$$

Before proving Lemma 15.4, let us show how it concludes the proof of Theorem 15.1. Lemma 15.4 implies that

$$\sum_{i=m}^{n} |a_i|^2 = \|\mathsf{f}_I^{>0}[y]\|_2^2 < (1+o(1))\epsilon$$

If $m = 2$, this is what we wanted to show. If $m > 2$, Claim 15.3 implies that m is not the minimal index satisfying (15.2), a contradiction. ■

Proof of Lemma 15.4

We now return to the proof of Lemma 15.4. For convenience, we write $\mathsf{f} = b + \sum_{i=1}^{n} a_i \chi_{\{i\}}$. Then the variance (not the variation!) of f is given by $\sum_i |a_i|^2$. Now, since the variance of $|\mathsf{f}|$ is bounded by $\| \, |\mathsf{f}| - 1 \, \|_2^2$ (this expression is minimized by replacing 1 with the expectation of $|\mathsf{f}|$), we have $\mathcal{V}(|\mathsf{f}|) < \epsilon$. Lemma 15.4 will therefore follow if we show that $\mathcal{V}(\mathsf{f})$ is essentially bounded by $\mathcal{V}(|\mathsf{f}|)$.

To prove this, we first show that the expectation of f, b, is well separated from zero. This holds since $|\mathsf{f}|$ is ϵ-close to 1 on the one hand, and $c\epsilon$-close to $|b|$ on the other hand. After proving this we assume, for instance, that b is positive. It follows that for almost all inputs x, $\mathsf{f}(x) = |\mathsf{f}(x)|$, since the weight of the non-constant part of f is rather small. This implies that $\mathbb{E}\mathsf{f} \approx \mathbb{E}|\mathsf{f}|$ and hence that $\mathcal{V}(\mathsf{f}) \approx \mathcal{V}(|\mathsf{f}|)$.

Fixing $c_2 \doteq (1+\sqrt{c})$, we have

$$\| \, |b| - 1 \, \|_2 \le \| \, |\mathsf{f}| - |b| \, \|_2 + \| \, |\mathsf{f}| - 1 \, \|_2 \le \|\mathsf{f} - b\|_2 + \sqrt{\epsilon} \le c_2\sqrt{\epsilon}$$

and hence $|b| \ge 1 - c_2\sqrt{\epsilon}$, therefore b is well-separated from zero. We assume without loss of generality that b is positive.

Writing $|\mathsf{f}| - \mathsf{f} = 2|\mathsf{f}|\mathbf{1}_{\{\mathsf{f}<0\}}$, we have

$$\mathbb{E}|\mathsf{f}| - \mathbb{E}\mathsf{f} \le 2\mathbb{E}|\mathsf{f}|\mathbf{1}_{\{\mathsf{f}<0\}} \tag{15.3}$$

To show that the expectations on the left-hand side are approximately equal, we bound the term on the right-hand side using the following special case of Azuma's inequality (see [Sch99] for a proof).

Theorem 15.5 (Azuma's inequality). *Let $X = \sum_{i=1}^{n} X_i$ be a sum of independent random variables with zero expectation, such that the absolute value of each x_i is bounded by d_i. Then*

$$\Pr\left[|X| > t\right] \le 2\exp\left(\frac{-t^2}{\sum_{i=1}^{n} d_i^2}\right)$$

The absolute value of a Rademacher function $\chi_{\{i\}}$ is bounded by some constant c_3, which only depends on p. Denoting $\lambda \doteq \sum_i |a_i|^2$, we have, by applying Azuma's inequality to $\sum_{i=1}^n a_i \chi_i$, that

$$\begin{aligned}\mathbb{E}|\mathsf{f}|\mathbf{1}_{\{\mathsf{f}<0\}} &= \int_{t=0}^{\infty} \Pr\left[\mathsf{f} < -t\right] dt = \int_{t=0}^{\infty} \Pr\left[b + \sum_i a_i \chi_i < -t\right] dt = \\ &= \int_{t=b}^{\infty} \Pr\left[\sum_i a_i \chi_i < -t\right] dt \leq 2 \int_{t=b}^{\infty} \exp\left(\frac{-t^2}{c_3^2 \lambda}\right) dt \leq \\ &\leq \frac{c_3^2 \lambda}{b} \int_{t=b}^{\infty} \frac{2t}{c_3^2 \lambda} \exp\left(\frac{-t^2}{c_3^2 \lambda}\right) dt \leq \frac{c_3^2 \lambda}{b} \exp\left(\frac{-b^2}{c_3^2 \lambda}\right)\end{aligned}$$

Now since $\lambda < c\epsilon$ and $b > 1 - c_2\sqrt{\epsilon}$, we have $\mathbb{E}|\mathsf{f}|\mathbf{1}_{\{\mathsf{f}<0\}} = o(\epsilon)$. Therefore, it follows from Equation (15.3) that

$$\begin{aligned}\epsilon > \||\mathsf{f}| - 1\|_2^2 &\geq \mathcal{V}(|\mathsf{f}|) = \|\mathsf{f}\|_2^2 - \mathbb{E}|\mathsf{f}|^2 = \mathcal{V}(\mathsf{f}) + \mathbb{E}\mathsf{f}^2 - \mathbb{E}|\mathsf{f}|^2 = \\ &= \mathcal{V}(\mathsf{f}) + (\mathbb{E}\mathsf{f} + \mathbb{E}|\mathsf{f}|)\big(\mathbb{E}\mathsf{f} - \mathbb{E}|\mathsf{f}|^2\big) \geq \\ &\geq \mathcal{V}(\mathsf{f}) + o(\epsilon) = \sum_i |a_i|^2 + o(\epsilon)\end{aligned}$$

which completes the proof.

Chapter 16

Extending FKN to Higher Frequencies

Following, is an extension of Theorem 15.1 to the case where f is concentrated on Walsh-products of size at most k rather than 1. This extension is applicable only when the weight of f on higher frequencies is smaller than a tiny constant (exponentially small in k), and in this sense it is weaker than Theorem 14.1. However, it gives a much better estimate of the asymptotic behavior of the distance of f from a junta, as a function of its weight on higher frequencies: The squared 2-norm distance from a (real-valued) junta is shown to be at most $1 + o(1)$ times the weight on high frequencies. We do not know whether the small range for which we prove this estimate is a weakness of our proof, or whether this really is the range where the squared 2-norm distance from a junta behaves according to this estimate.

In Chapter 17 it is shown that the following theorem may be used to considerably improve the parameters in Theorem 14.1.

Theorem 16.1 (high-frequency FKN). *Let $\mathsf{f}\colon \mathcal{P}([n]) \to \mathbb{R}$ be a real valued function of degree k, namely $\mathsf{f}^{>k} \equiv 0$. Let $\epsilon \doteq \| |\mathsf{f}| - 1 \|_2^2$ measure the squared distance of f from the nearest boolean function. Then there exists a subset $\mathcal{J} \subseteq \mathcal{P}([n])$ of coordinates whose size is bounded by a global constant (depending only on k), such that*

$$\mathsf{Vr}_{\mathsf{f}}(\bar{\mathcal{J}}) \leq (1 + o(1))\epsilon \ ,$$

where $\bar{\mathcal{J}} \doteq [n] \setminus \mathcal{J}$.

Note that theorem 12.3 follows from Theorem 16.1, using the same proof as in Corollary 15.2.

16.1 Proof of Theorem 16.1

We begin by taking τ to be a small enough constant ($\tau \approx \delta^{16k}$), that is specified exactly later. The set $\mathcal{J}$ can already be declared by

$$\mathcal{J} \doteq \{i \in [n] \mid \mathsf{Vr}_{\mathsf{f}}(\{i\}) > \tau\}$$

Note that since f is of degree k, $|\mathcal{J}| \leq k\tau^{-1}$, so $|\mathcal{J}|$ is bounded by a constant and therefore it suffices to prove that $\mathsf{Vr}_{\mathsf{f}}(\bar{\mathcal{J}}) \leq (1 + o(1))\epsilon$,.

Suppose that $\bar{\mathcal{J}}$ is not empty (otherwise there is nothing to prove), and let us assume, without loss of generality, that $\epsilon < \tau$. We consider sets $I \subseteq \bar{\mathcal{J}}$ that satisfy $\mathsf{Vr}_{\mathsf{f}}(I) \leq 3\tau$, and take $I \subseteq \bar{\mathcal{J}}$ to be a maximal set with this property.

Program of Proof. In the proof of Theorem 15.1 we used the fact that the variation on a set I of coordinates is also the variation on I of any restriction $\mathsf{f}_I[x]$. We could thus fix x and focus only on $\mathsf{f}_I[x]$, as in Lemma 15.4. Here this is not the case, however according to Claim 13.10 the variation on I of f, which is bounded by τ, is the average of the variations on I of restrictions of the form $\mathsf{f}_I[x]$. The proof thus begins by bounding the deviation of the variations on I of restrictions $\mathsf{f}_I[x]$, showing that the contribution of restriction with high variation to this average is very small. For restrictions $\mathsf{f}_I[x]$ where the variation on I is not very high, it is shown that the squared 2-norm distance of $\mathsf{f}_I[x]$ from the nearest boolean function is essentially bounded below by $\mathsf{Vr}_{\mathsf{f}_I[x]}(I)$.

By averaging over all restrictions, this implies that the distance of f from the nearest boolean function is essentially bounded below by $\mathsf{Vr}_{\mathsf{f}}(I)$, and therefore $\mathsf{Vr}_{\mathsf{f}}(I) < (1+o(1))\epsilon$. This completes the proof, since if $I = \bar{\mathcal{J}}$ we are obviously done, but on the other hand, if $I \neq \bar{\mathcal{J}}$, one can add a coordinate to I, keeping its variation below 3τ, in contradiction to the maximality of I.

Bounding High Variations of Restrictions

To show that there cannot be too many restrictions $\mathsf{f}_I[x]$ with large variation, we need the following lemma, proven in the next section.

Lemma 16.2. *Let $\mathsf{g}_1, \ldots, \mathsf{g}_m \colon \mathcal{P}([l]) \to \mathbb{R}$ be real-valued functions such that $\mathsf{g}_i^{>k} \equiv 0$ for every i. Then for every $\alpha \geq 0$,*

$$\Pr_{x \sim \mu^m}\left[\sum |\mathsf{g}_i(x)|^2 > \alpha\right] \leq 256\alpha^{-2}\delta^{-4k}\left(\sum_{i=1}^{m} \|\mathsf{g}_i\|_2^2\right)^2$$

where $\delta = \min\left\{\delta_p, \delta_{1/2}\right\}$ is the minimum between the δ parameters obtained from Theorem 13.13 for p and for $1/2$.

For shortness, denote $\eta \doteq \mathsf{Vr}_{\mathsf{f}}(I)$ (then $\eta < 3\tau$), and let

$$\mathcal{D} \doteq \big\{ x \in \mathcal{P}(\bar{I}) \mid \mathsf{Vr}_{\mathsf{f}_I[x]}(I) > \eta^{3/4} \big\}$$

be the set of restrictions whose variation is much higher than expected.

Proposition 16.3.

$$\mathop{\mathbb{E}}_{x \sim \mu^{\bar{I}}} \left[\mathsf{Vr}_{\mathsf{f}_I[x]}(I) \mathbf{1}_{\{x \in \mathcal{D}\}} \right] < 512 \delta^{-4k} \eta^{5/4}$$

Proof. For a non-empty set $S \subseteq I$, define for every $x \in \bar{I}$,

$$g_S(x) \doteq \widehat{\mathsf{f}_I[x]}(S)$$

Then each g_S is a function of degree at most $k-1$, and for every x,

$$\mathsf{Vr}_{\mathsf{f}_I[x]}(I) = \sum_{\substack{S \subseteq I \\ S \neq \emptyset}} g_S^2(x)$$

It follows that

$$\sum_{\substack{S \subseteq I \\ S \neq \emptyset}} \|g_S\|_2^2 = \mathop{\mathbb{E}}_x \Big[\sum_{\substack{S \subseteq I \\ S \neq \emptyset}} g_S^2(x) \Big] = \mathop{\mathbb{E}}_x \left[\mathsf{Vr}_{\mathsf{f}_I[x]}(I) \right] = \mathsf{Vr}_{\mathsf{f}}(I) = \eta$$

Hence

$$\begin{aligned} \mathbb{E} \mathsf{Vr}_{\mathsf{f}_I[x]}(I) \mathbf{1}_{\{x \in \mathcal{D}\}} = & \\ &= \int_{t=0}^{\infty} \Pr\left[\sum g_S(x)^2 \geq \max(t, \eta^{3/4}) \right] dt = \\ &= \int_{t=0}^{\eta^{3/4}} \Pr\left[\sum g_S(x)^2 \geq \eta^{3/4} \right] dt + \\ &\quad + \int_{t=\eta^{3/4}}^{\infty} \Pr\left[\sum g_S(x)^2 \geq t \right] dt \leq \end{aligned}$$

(using Lemma 16.2)

$$\begin{aligned} &\leq \eta^{3/4} \cdot 256 \delta^{-4k} \eta^{-3/2} \eta^2 + 256 \delta^{-4k} \eta^2 \int_{t=\eta^{3/4}}^{\infty} t^{-2} dt = \\ &= 512 \delta^{-4k} \eta^{5/4} \end{aligned}$$

■

Bounding $\mathsf{Vr}_{\mathsf{f}_I[x]}(I)$ for $x \notin \mathcal{D}$

Proposition 16.4. *For every $x \notin \mathcal{D}$,*

$$\| \, |\mathsf{f}_I[x]| - 1 \, \|_2^2 \geq \mathsf{Vr}_{\mathsf{f}_I[x]}(I) - 20\delta^{-4k}\eta^{3/2}$$

Proof. Define

$$\mathcal{C} \doteq \left\{ x \in \bar{I} \;\middle|\; \left| |\widehat{\mathsf{f}_I[x]}(\emptyset)| - 1 \right| > \frac{1}{2} \right\}$$

We proof the statement separately for $x \in \mathcal{C} \setminus \mathcal{D}$ and for $x \notin \mathcal{C} \cup \mathcal{D}$.

The case $x \in \mathcal{C} \setminus \mathcal{D}$. It suffices to show that in this case for most $y \in \mathcal{P}(I)$, $\left| |\mathsf{f}_I[x](y)| - 1 \right| \geq 1/4$. Note that $\mathsf{f}_I[x] - \widehat{\mathsf{f}_I[x]}(\emptyset)$ is a function of degree at most k, and that since $x \notin \mathcal{D}$,

$$\| \mathsf{f}_I[x] - \widehat{\mathsf{f}_I[x]}(\emptyset) \|_2^2 = \mathsf{Vr}_{\mathsf{f}_I[x]}(I) \leq \eta^{3/4}$$

Hence by Claim 14.10

$$\Pr_{y \sim \mu^I} \left[\left| \mathsf{f}_I[x](y) - \widehat{\mathsf{f}_I[x]}(\emptyset) \right| > 1/4 \right] < 4^4 \delta^{-4k} \eta^{3/2}$$

It follows that with probability at least $1 - 4^4\delta^{-4k}\eta^{3/2}$, $\left| |\mathsf{f}_I[x](y)| - 1 \right| > 1/4$. Therefore in this case

$$\| \, |\mathsf{f}_I[x]| - 1 \, \|_2^2 \geq \frac{1}{16}(1 - 4^4\delta^{-4k}\eta^{3/2}) \gg \tau^{3/4} > \eta^{3/4} \geq \mathsf{Vr}_{\mathsf{f}_I[x]}(I)$$

The case $x \notin \mathcal{C} \cup \mathcal{D}$. Recall that $\mathsf{Vr}_{\mathsf{f}_I[x]}(I) = \mathcal{V}(\mathsf{f}_I[x])$ and note that $\| \, |\mathsf{f}_I[x]| - 1 \, \|_2^2$ is bounded from below by $\mathcal{V}(|\mathsf{f}_I[x]|)$. We thus show that $\mathcal{V}(|\mathsf{f}_I[x]|) \gtrsim \mathcal{V}(\mathsf{f}_I[x])$. For this purpose, we assume without loss of generality that $\widehat{\mathsf{f}_I[x]}(\emptyset)$ is positive (it is therefore at least 1/2 and at most 3/2). One sees that

$$\begin{aligned}
\mathcal{V}(\mathsf{f}_I[x]) - \mathcal{V}(|\mathsf{f}_I[x]|) &= \| \mathsf{f}_I[x]\|_1^2 - \left|\widehat{\mathsf{f}_I[x]}(\emptyset)\right|^2 = \\
&= \Big(\| \mathsf{f}_I[x]\|_1 + \widehat{\mathsf{f}_I[x]}(\emptyset)\Big)\Big(\| \mathsf{f}_I[x]\|_1 - \widehat{\mathsf{f}_I[x]}(\emptyset)\Big) \leq \\
&\leq \Big(\| \mathsf{f}_I[x]\|_2 + \widehat{\mathsf{f}_I[x]}(\emptyset)\Big)\Big(\| \mathsf{f}_I[x]\|_1 - \widehat{\mathsf{f}_I[x]}(\emptyset)\Big) \leq \\
&\leq \bigg(\Big(\mathcal{V}(\mathsf{f}_I[x]) + \widehat{\mathsf{f}_I[x]}(\emptyset)^2\Big)^{1/2} + \widehat{\mathsf{f}_I[x]}(\emptyset)\bigg)\Big(\| \mathsf{f}_I[x]\|_1 - \widehat{\mathsf{f}_I[x]}(\emptyset)\Big) \leq \\
&\leq 6\Big(\| \mathsf{f}_I[x]\|_1 - \widehat{\mathsf{f}_I[x]}(\emptyset)\Big) = \\
&= 6 \mathop{\mathbb{E}}_{y\sim\mu^I} \big[|\, \mathsf{f}_I[x](y)\,| - \mathsf{f}_I[x](y)\big] = \\
&= 6 \mathop{\mathbb{E}}_{y\sim\mu^I} \big[|\, \mathsf{f}_I[x](y)\,| \cdot \mathbf{1}_{\{\mathsf{f}_I[x](y)<0\}}\big] \leq 6 \int_{t=0}^{\infty} \Pr\left[\mathsf{f}_I[x] < -t\right] dt = \\
&= 6 \int_{t=0}^{\infty} \Pr\Big[\mathsf{f}_I[x] - \widehat{\mathsf{f}_I[x]}(\emptyset) + \widehat{\mathsf{f}_I[x]}(\emptyset) > -t\Big] dt \leq
\end{aligned}$$

(using claim 14.10)

$$\leq 6\delta^{-4k}\eta^{3/2} \int_{t=\widehat{\mathsf{f}_I[x]}(\emptyset)}^{\infty} t^{-4} dt \leq 20\delta^{-4k}\eta^{3/2}$$

Hence we are done. ■

Completion of the Argument

From Proposition 16.4 and Proposition 16.3, we have

$$\begin{aligned}
\eta = \mathsf{Vr}_{\mathsf{f}}(I) &= \mathop{\mathbb{E}}_{x\sim\mu^{\bar{I}}} \big[\mathsf{Vr}_{\mathsf{f}_I[x]}(I)\big] = \\
&= \mathop{\mathbb{E}}_{x\sim\mu^{\bar{I}}} \big[\mathsf{Vr}_{\mathsf{f}_I[x]}(I) \cdot \mathbf{1}_{\{x\in\mathcal{D}\}}\big] + \mathop{\mathbb{E}}_{x\sim\mu^{\bar{I}}} \big[\mathsf{Vr}_{\mathsf{f}_I[x]}(I) \cdot \mathbf{1}_{\{x\notin\mathcal{D}\}}\big] < \\
&< 512\delta^{-4k}\eta^{5/4} + \mathop{\mathbb{E}}_{x\sim\mu^{\bar{I}}} \big[\|\, |\mathsf{f}_I[x]| - 1\,\|_2^2 \cdot \mathbf{1}_{\{x\notin\mathcal{D}\}}\big] + 20\delta^{-4k}\eta^{3/2} \leq \\
&\leq 532\delta^{-4k}\eta^{5/4} + \mathop{\mathbb{E}}_{x\sim\mu^{\bar{I}}} \big[\|\, |\mathsf{f}_I[x]| - 1\,\|_2^2\big] = \\
&= 532\delta^{-4k}(\mathsf{Vr}_{\mathsf{f}}(I))^{5/4} + \|\, |\mathsf{f}| - 1\,\|_2^2 = \\
&= 532\delta^{-4k}\eta^{5/4} + \epsilon
\end{aligned}$$

From which it follows that

$$\eta\big(1 - 532\delta^{-4k}\eta^{1/4}\big) < \epsilon \tag{16.1}$$

We now select τ to be

$$\frac{\delta^{16k}}{3(1064)^4}$$

Since $\eta < 3\tau$, we have $532\delta^{-4k}\eta^{1/4} < 1/2$, and thus Equation (16.1) yields $\eta < 2\epsilon$. Putting this into Equation (16.1) again, we get

$$\mathsf{Vr}_{\mathsf{f}}(I) = \eta < \frac{\epsilon}{1 - 532\delta^{-4k}\eta^{1/4}} < \epsilon\big(1 + 1064\delta^{-4k}\eta^{1/4}\big) < \\ < \epsilon\big(1 + 1064\delta^{-4k}(2\epsilon)^{1/4}\big) = \epsilon\big(1 + o(1)\big)$$

thus completing the proof.

16.2 Proof of Lemma 16.2

Before we prove Lemma 16.2, we need the following technical observation.

Lemma 16.5. *Let $\lambda_1, \ldots, \lambda_m$ be (not all zero) real numbers, and let $y_1, \ldots, y_n$ be independent random variables, distributed uniformly on $\{-1, 1\}$. Then*

$$\Pr\left[\Big(\sum_{i=1}^{n} \lambda_i y_i\Big)^2 > \frac{1}{4}\sum_{i=1}^{n} \lambda_i^2\right] > \frac{1}{16}$$

Proof. Set $\lambda^2 \doteq \sum_i \lambda_i^2$, and let

$$p(t) \doteq \Pr\big[\Big(\sum_{i=1}^{n} \lambda_i y_i\Big)^2 > t\big]$$

Then

$$\lambda^2 = \mathbb{E}\Big(\sum_{i=1}^{n} \lambda_i y_i\Big)^2 = \int_{t=0}^{\infty} p(t)dt = \int_{t=0}^{\lambda^2/4} p(t)dt + \int_{t=\lambda^2/4}^{8\lambda^2} p(t)dt + \int_{t=8\lambda^2}^{\infty} p(t)dt \leq \\ \leq \lambda^2/4 + 8\lambda^2 p(\lambda^2/4) + \int_{t=8\lambda^2}^{\infty} p(t)dt \tag{16.2}$$

Let us bound the last term on the right-hand side of (16.2). We use Azuma's inequality (Theorem 15.5).

$$\int_{t=8\lambda^2}^{\infty} \Pr\big[\Big(\sum_{i=1}^{n} \lambda_i y_i\Big)^2 > t\big]dt < 2\int_{t=8\lambda^2}^{\infty} \exp\left(-\frac{t}{2\lambda^2}\right)dt = 4\lambda^2 \exp\left(-\frac{8\lambda^2}{2\lambda^2}\right) < \lambda^2/4$$

Putting this back into (16.2) we have

$$p(\lambda^2/4) > \frac{\lambda^2/2}{8\lambda^2} = 1/16$$

which is what we wanted. ■

Proof of Lemma 16.2: For $x \in \mathcal{P}([l])$ and $y \in \mathcal{P}([l+m] \setminus [l])$, let

$$\mathsf{g}(x \cup y) \dot{=} \sum_{i \notin y} \mathsf{g}_i(x) - \sum_{i \in y} \mathsf{g}_i(x) = \sum_{i=1}^{m} v_i(y)\mathsf{g}_i(x)$$

where v_i is the i'th Rademacher function for bias 1/2. Then g contains mixed walsh-products (with some biased Rademacher functions and some uniform Rademachers) of size at most $k+1$, and

$$\|\mathsf{g}\|^2_{2,\mu^l \times \mu^m_{1/2}} = \sum_{i=1}^{m} \|\mathsf{g}_i\|_2^2$$

According to Lemma 16.5,

$$\Pr_{x \sim \mu^m}\left[\sum |\mathsf{g}_i(x)|^2 > \alpha\right] \leq 16 \Pr_{\substack{x \sim \mu^l \\ y \sim \mu^m_{1/2}}}\left[\mathsf{g}(x \cup y)^2 > \alpha/4\right] \tag{16.3}$$

To bound the right-hand side of (16.3), we use Claim 14.10 with respect to the measure* $\mu^l \times \mu^m_{1/2}$. We obtain, for some global constant δ (here δ is the minimum between the δ that is valid in Theorem 13.13 for the uniform measure and for the biased measure)

$$\Pr_{x \sim \mu^m}\left[\sum |\mathsf{g}_i(x)|^2 > \alpha\right] \leq 16 \cdot 16\alpha^{-2}\delta^{-4k}\|\mathsf{g}\|_2^4 \leq 256\alpha^{-2}\delta^{-4k}\left(\sum_{i=1}^{m} \|\mathsf{g}_i\|_2^2\right)^2$$

■

* Claim 14.10 requires Theorem 13.13. As is shown in [Bec75], this theorem can be applied to a product of two-point spaces, even if each is equipped with a different measure. In our case the coordinates of x lie in two-point spaces with a biased measure, and the coordinates of y are uniformly distributed

Chapter 17

Improving the Junta Threshold

Building on the strengthening of Theorem 14.7 by Theorem 16.1, we turn to improve the tradeoff between $\|\mathsf{f}^{>k}\|_2^2$ and the distance of f from a junta. As a consequence, we improve the threshold on $\|\mathsf{f}^{>k}\|_2^2$ which suffices to ensure that f is close to a junta. The main argument in this chapter is contained in the following lemma.

Lemma 17.1. *Fix a positive integer ℓ, and let $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ be a boolean function. Then there exists a global constant C and a global positive constant $\delta > 0$, such that for any $\tau > 0$ and any positive integers k and $r > 2k$,*

$$\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}\left[\mathsf{Vr}_{\mathsf{f}}(I)\right] \le C\Big(\delta^{-4k}\tau + (k/r)^{\ell+1} + \|\mathsf{f}^{>k}\|_2^2\Big)$$

where $\mathcal{J} = \Big\{i \in [n] \mid \mathsf{Vr}_{\mathsf{f}}^{\le k}(i) \ge \tau\Big\}$.

Let us show how Lemma 17.1 implies Thereom 12.1. For convenience, we first cite it again here.

Theorem 12.1. *Fix a positive integer ℓ. Then there exists a constant $\delta > 0$, such that for every $\epsilon > 0$ and every boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$ satisfying*

$$\|\mathsf{f}^{>k}\|_2^2 \le \left(\frac{\epsilon}{k}\right)^{(\ell+1)/\ell}$$

is an $(O(\epsilon), J)$-junta, where

$$J = O\big(\delta^{-4k}\epsilon^{(\ell+1)/\ell}k^{(2\ell+1)/\ell}\big)$$

Proof. Following the proof of Theorem 14.3, but using the parameters of Lemma 17.1, we obtain the following Lemma.

Lemma 17.2. *Fix a positive integer ℓ. There exist global positive constants C and δ, such that for any $\tau > 0$, and for any positive integers k and $r > 2k$, the following holds. For every boolean function $\mathsf{f}\colon \mathcal{P}([n]) \to \{-1,1\}$, f is an $[\epsilon, J]$-junta with respect to μ^n, where $J = k/\tau$ and*

$$\epsilon = Cr\Big(\delta^{-4k}\tau + (k/r)^{\ell+1} + \|\mathsf{f}^{>k}\|_2^2\Big)$$

Taking

$$r \doteq \big(k^{\ell+1}/\epsilon\big)^{1/\ell} \quad \text{and} \quad \tau \doteq \delta^{4k}(\epsilon/k)^{(\ell+1)/\ell}$$

we obtain Theorem 12.1. ■

Corollary 12.2 follows from Theorem 12.1, with the same proof as that of corollary 14.2.

17.1 Proof of Lemma 17.1

The proof of Lemma 17.1 is similar to that of Lemma 14.6. First, it is shown that on a random I chosen according to $\mu_{1/r}^{\bar{\mathcal{J}}}$, it is very likely that for almost all input settings x outside I the weight of $\mathsf{f}_I[x]$ is concentrated on Walsh-products of size at most ℓ. Theorem 16.1 is then applied, showing that in this case $\mathsf{f}_I[x]$ must either have a large Walsh-coefficient, or be close to constant. It is then shown that in fact the first alternative almost never occurs, hence the lemma follows.

When $k = \ell$

We start by a corollary of Theorem 16.1, showing that a boolean function that is concentrated on Walsh-products of size at most ℓ, either has a large Walsh-coefficient, or is very close to constant.

Corollary 17.3. *Fix a positive integer ℓ. Then there exist constants d,M, with properties as follows. For every boolean function $\mathsf{g}\colon \mathcal{P}(I) \to \{-1,1\}$, either there exists a non-empty subset $T \subseteq I$ of size at most ℓ with $|\widehat{\mathsf{g}}(T)| > d$; or $\mathsf{Vr}_{\mathsf{g}}(I) < M\|\mathsf{g}^{>\ell}\|_2^2$.*

Proof. We can assume that $\|\mathsf{g}^{>\ell}\|_2^2$ is smaller than any tiny constant. In that case, Theorem 16.1 implies that there exists a constant-sized subset $\mathcal{J} \subseteq I$ such that

$$\|\mathsf{g} - \mathsf{Avg}_{\bar{\mathcal{J}}}[\mathsf{g}]\|_2^2 \le (1 + o(1))\|\mathsf{g}^{>\ell}\|_2^2$$

Denoting $\mathsf{g}' \doteq \operatorname{sign}(\mathsf{Avg}_{\bar{\mathcal{J}}}[\mathsf{g}])$, one easily verifies that

$$\|\mathsf{g} - \mathsf{g}'\|_2^2 \le (4 + o(1))\|\mathsf{g}^{>\ell}\|_2^2$$

If g' is constant, we are done. If it is not constant, it must be that $\mathsf{Avg}_{\bar{\mathcal{J}}}[\mathsf{g}] - \widehat{\mathsf{g}}(\emptyset)$ obtains values larger than 1. It follows that the weight of g on characters χ_T, where $T \subseteq \mathcal{J}$ and $T \neq \emptyset$, is bounded from below by some very small constant. Since there are only a constant number of such characters, g satisfies the first condition in Corollary 17.3. ■

Few Large Coefficients

We now return to the proof of Lemma 17.1 which states, in essence, that for most I's and x's $\mathsf{f}_I[x]$ is almost constant. We begin by giving an upper-estimate on the (weighted) number of restrictions $\mathsf{f}_I[x]$ that can be far from constant. The next subsection will show that indeed most restriction are almost constant.

For a given $I \subseteq [n]$ denote

$$\mathcal{D}_I \doteq \left\{ x \in \mathcal{P}(\bar{I}) \mid \exists T \in I \quad |T| \leq \ell \ , \left|\widehat{\mathsf{f}_I[x]}(T)\right| > d \right\}$$

where d is as in Corollary 17.3. To bound the measure of $\mathcal{D}_I$, we note that the coefficient of χ_T in $\mathsf{f}_I[x]$ is a function of x that is concentrated on low-frequencies, and has small norm (since every $i \in I$ has small variation). Hence according to Theorem 13.13, it cannot often attain large values, and therefore the coefficient of χ_T almost never reaches d.

Fix $T \subseteq I$ to be a non-empty set of size at most ℓ, and consider the function $g_T : \mathcal{P}(\widehat{I}) \to \mathbb{R}$, which assigns to every x the coefficient of χ_T in f_I. That is,

$$g_T(x) = \widehat{\mathsf{f}_I[x]}(T)$$

For x to be in $\mathcal{D}$, one of the g_T's must evaluate to at least d in absolute value. Applying lemma 14.9, with parameters $\alpha = d/2$ and $\beta = d$, we get a bound on the probability, for a random x, that $\mathsf{f}_I[x]$ is a dictatorship.

$$\begin{aligned}
\Pr_{x \sim \mu^{\bar{I}}}[x \in \mathcal{D}_I] &\leq \sum_{\substack{T \subseteq I \\ T \neq \emptyset}} \Pr_{x \sim \mu^{\bar{I}}}[|g_T(x)| > d] \leq \\
&= 16 d^{-4} \delta^{-4k} \sum_{\substack{T \subseteq I \\ T \neq \emptyset}} \|\mathsf{g}_T^{\leq k}\|_2^4 + \frac{4}{d^2} \sum_{T \subseteq I} \|\mathsf{g}^{>k}\|_2^2 = \\
&= 16 d^{-4} \delta^{-4k} \sum_{\substack{T \subseteq I \\ T \neq \emptyset}} \Bigg\| \sum_{\substack{S \subseteq [n], |S| \leq k \\ S \cap I = T}} \widehat{\mathsf{f}}(S) \chi_S \Bigg\|_2^4 + \frac{4}{d^2} \sum_{\substack{T \subseteq I \\ T \neq \emptyset}} \Bigg\| \sum_{\substack{S \subseteq [n], |S| > k \\ S \cap I = T}} \widehat{\mathsf{f}}(S) \chi_S \Bigg\|_2^2 \leq \\
&\leq 16 d^{-4} \delta^{-4k} \sum_{\substack{T \subseteq I \\ T \neq \emptyset}} \Bigg(\sum_{\substack{S \subseteq [n], |S| \leq k \\ S \cap I = T}} \widehat{\mathsf{f}}^2(S) \Bigg)^2 + \frac{4}{d^2} \|\mathsf{f}^{>k}\|_2^2
\end{aligned}$$

Since the sum of $\widehat{\mathsf{f}}^2(S)$ over *all* S's equals 1, we have

$$\sum_{\substack{T \subseteq I \\ T \neq \emptyset}} \Bigg(\sum_{\substack{S \subseteq [n], |S| \leq k \\ S \cap I = T}} \widehat{\mathsf{f}}^2(S) \Bigg)^2 \leq \max_{\substack{T \subseteq I \\ T \neq \emptyset}} \sum_{\substack{S \subseteq [n], |S| \leq k \\ S \cap I = T}} \widehat{\mathsf{f}}^2(S) \leq \max_{i \in I} \mathsf{Vr}_{\mathsf{f}}^{\leq k}(i) < \tau$$

Altogether this implies that for some constant M_1,

$$\Pr_{x\sim\mu^{\bar{I}}}[x\in\mathcal{D}_I]\le M_1\delta^{-4k}\tau+M_1\|\mathsf{f}^{>k}\|_2^2$$

Restrictions are Mostly Constant

We are now ready to prove that the restrictions $\mathsf{f}_I[x]$ are mostly constant. First, note that for an x such that $x\notin\mathcal{D}_I$, Corollary 17.3 asserts that

$$\mathsf{Vr}_{\mathsf{f}_I[x]}(I)\le M\sum_{|R|>\ell}\widehat{\mathsf{f}_I[x]}^2(R)$$

and by Claim 13.10 we have that

$$\begin{aligned}\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}[\mathsf{Vr}_{\mathsf{f}}(I)]=\mathop{\mathbb{E}}_{\substack{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}\\ x\sim\mu^{\bar{I}}}}\left[\mathsf{Vr}_{\mathsf{f}_I[x]}(I)\right]\quad&\le\\ \le\Pr_{\substack{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}\\ x\sim\mu^{\bar{I}}}}[x\in\mathcal{D}_I]+\mathop{\mathbb{E}}_{\substack{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}\\ x\sim\mu^{\bar{I}}}}\left[M\sum_{|R|>\ell}\widehat{\mathsf{f}_I[x]}^2(R)\right]\quad&\le\\ \le M_1\delta^{-4k}\tau+M_1\|\mathsf{f}^{>k}\|_2^2+M\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}\left[\sum_{|S\cap I|>\ell}\widehat{\mathsf{f}}^2(S)\right]&\end{aligned}$$

Now note that
$$\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}\Big[\sum_{\substack{|S|\le k\\ |S\cap I|>\ell}}\widehat{\mathsf{f}}^2(S)\Big]\le\frac{(k/r)^{\ell+1}}{1-k/r}\ :$$

This holds since for $S\subseteq[n]$ with $|S|\le k$,

$$\Pr_I[|S\cap I|>\ell]\le\sum_{i=\ell+1}^{k}\binom{k}{i}r^{-i}(1-1/r)^{k-i}\le\sum_{i=\ell+1}^{k}k^ir^{-i}\le\frac{(k/r)^{\ell+1}}{1-k/r}$$

and since the total weight of all Walsh-products is bounded by 1.

Therefore, we get that the overall probability of disagreement with the majority is bounded by

$$\mathop{\mathbb{E}}_{I\sim\mu_{1/r}^{\bar{\mathcal{J}}}}[\mathsf{Vr}_{\mathsf{f}}(I)]\le M_1\delta^{-4k}\tau+M_1\|\mathsf{f}^{>k}\|_2^2+M\left(\frac{(k/r)^{\ell+1}}{1-k/r}+\|\mathsf{f}^{>k}\|_2^2\right)$$

Chapter 18

The Biased Bonami-Beckner Inequality

In this chapter we prove Theorem 13.13. As is shown in [Bec75], it is enough to prove the statement for the two-point space, namely for the case where $\mathsf{f}\colon \{\emptyset, \{1\}\} \to \mathbb{R}$. By homogeneity, we may assume that $\mathsf{f}(\emptyset) = 1$, and denote $t \doteq \mathsf{f}(\{1\})$. Also, for convenience we denote $q \doteq 1 - p$. It is easy to verify that

$$T_\delta[\mathsf{f}](\emptyset) = q + \delta q + (1-\delta)pt \ , \quad \text{and} \quad T_\delta[\mathsf{f}](\{1\}) = (1-\delta)q + (p + \delta q)t$$

Hence we should show that

Lemma 18.1. *for every $p \in (0,1)$ there exists a $\delta = \delta_p > 0$, such that for every t,*

$$\left[q(q + \delta p + (1-\delta)pt)^4 + p((1-\delta)q + (p+\delta q)t)^4\right]^{1/4} \le \sqrt{q + pt^2}$$

Proof. The following inequality is equivalent to the inequality stated in lemma 18.1.

$$q(q + \delta p + (1-\delta)pt)^4 + p((1-\delta)q + (p+\delta q)t)^4 - (q + pt^2)^2 \le 0 \qquad (\circ)$$

To prove it for every t (for an appropriate δ_p), it is obviously enough to consider only non-negative t's in $(\circ)$. Moreover, notice that replacing t by $1/t$ in $(\circ)$ and multiplying by t^4 yields the same inequality, only where the roles of p and q are reversed. We can therefore find a parameter δ that is suitable only for t's in the segment $[0,1]$, and then take δ_p to be the minimum between the δ parameters obtained for p and that which is obtained for q (it is known that $\|T_\delta(\mathsf{f})\|_4$ is a decreasing function in δ).

We are thus left with the task of proving that for every $p \in (0,1)$, there exists a positive δ, such that $(\circ)$ holds for every $t \in [0,1]$. Denoting

$$A_\delta(t) \doteq q(q + \delta p + (1-\delta)pt)^4 + p((1-\delta)q + (p+\delta q)t)^4$$

and

$$B(t) \doteq (q + pt^2)^2$$

this is equivalent to finding a δ such that

$$\forall\ t \in [0,1] \qquad \ln A_\delta(t) - \ln B(t) \leq 0 \tag{$*$}$$

For every δ, $A_\delta(1) = B(1) = 1$, and hence $(*)$ holds there as an equality. We would like to show that the derivative, with respect to t, of the left-hand side of $(*)$ is non-negative in the segment $[0,1]$ for an appropriate δ. This would imply that for this δ, $(*)$ holds throughout the segment.

Since it can also be verified that $A'_\delta(1) = B'(1) = 1$ (where F' denotes the derivative of F with respect to t), we have that the derivative of the left-hand side of $(*)$ zeros at 1 as well. Hence by similar arguments as above, if we prove that the second derivative of the left-hand side of $(*)$ is non-positive for $t \in [0,1]$, for a suitable positive δ, we are done.

What we show, in fact, is that

$$\max_{t \in [0,1]} (\ln A_0 - \ln B)''(t) < 0$$

That is, we show that for $\delta = 0$, the maximum of the second derivative in $(*)$ is strictly negative. Since the maximum is a continuous function of δ, we have that there also exists a positive δ for which the maximum is still negative, which completes the proof.

Taking two derivatives of the left-hand side of $(*)$, we obtain

$$(\ln A_0 - \ln B)''(t) = -4p\left(\frac{p}{(q+pt)^2} + \frac{q - pt^2}{(q+pt^2)^2}\right)$$

Taking a common denominator, we get

$$(\ln A_0 - \ln B)''(t) = -4pq(q+pt)^{-2}(q+pt^2)^{-2}\big(q + 2pqt + (3p^2 - pq)t^2 - 2p^2t^3\big)$$

The third clause in the above expression equals

$$2p^2(t^2 - t^3) + pq(t - t^2) + p^2t^2 + pqt + q$$

which is at least q for every $t \in [0,1]$. It follows that the second derivative is strictly negative for every such t. The proof of Lemma 18.1, and hence the proof of Theorem 13.13, is thus complete. ■

Discussion and Open Problems

Following are some remarks and open-problems arising from, or related to, results in this thesis.

The Sliding-Scale Conjecture

In Part I of this thesis, it is proved that the sliding-scale conjecture of [BGLR93] holds for almost polynomial range-size, namely where the size of the range is up to $2^{\log^{\beta} n}$, and β is any constant below 1. It would be most interesting to know whether it holds for the full applicable range, namely for range-size up to and including some polynomial in the length of the proof. In addition to giving a full characterization of NP in terms of PCP, in the case of a constant number of accesses for each local-test, such a result is likely to have other applications.

For example, in [DKRS98] it is proven that the problem of finding the distance between a given vector and a given lattice in $\mathbb{R}^n$, namely the Closest Vector Problem, is NP-hard to approximate to within almost polynomial factors (in fact they obtain an in-approximability ratio of the form $2^{\log^{\beta} n}$, where β approaches 1 as n goes to infinity). Some of the methods used there are very similar to those used in the first part of this thesis. It seems that the methods required for proving that the sliding-scale conjecture holds for polynomially-sized range, are likely to also give a polynomial in-approximability ratio for the Closest Vector Problem.

The J-Junta Test

An efficient non-adaptive one-sided J-junta test. Instead of the two-sided version of the J-junta test which appears in Chapter 9, it is also possible to obtain quadratic dependency on J by somewhat relaxing the soundness requirement (as proposed to us by A. Wigderson). This is obtained if while requiring the test to accept every J-junta, we only require that it rejects inputs which are, say, not even $(\epsilon, 2J)$-juntas.

To achieve the quadratic dependency on J, note that we chose the number of elements in the partition to be quadratic in J, so that any $J+1$ influential coordinates would go into distinct subsets in the partition with high probability. If we allow juntas of size $2J$ to be accepted, it is enough to take a partition of size only linear in J. This reduces the number of queries by a factor of J. But since the subsets in the partition are now larger, we can take the 'junta threshold' to be linear in $1/J$, and reduce by a factor of J the number of independence tests applied to each subset, as explained there for the two-sided case.

Lower-bound conjecture. We believe that J^2/ϵ is a lower-bound for the query complexity of both the one-sided and the two-sided non-adaptive J-junta tests (up to logarithmic factors). In light of the two-sided test presented in Chapter 9, this seems to be a tight lower-bound, up to logarithmic factors, for the two-sided test. We believe that J^2/ϵ is a lower-bound also for the relaxed test proposed in the previous remark.

Testing permutations for non-juntas. With regards to testing that f is a permutation of a given function h (Chapter 11), we can pose the following question. If we know that a function h with n variables is far from all J-juntas, does it imply a lower bound (that depends on J) on the number of queries required for testing that f is a permutation of h? The lower bound proof presented in Chapter 10 already implies such a bound for *some* functions h, namely, those that are characters of size J.

Random Walks on Cayley Graphs

In Chapter 10 it is proven that a random walk on a weighted Cayley graph of $\mathbb{Z}_n^q$ has similar distributions in time t and in time $t+2$ for relatively small values of t, even though it may take much longer before the walk becomes stationary. Since the bound given on t is independent of the weights given to the generators in the graph, this is a property of the group $\mathbb{Z}_n^q$. The following question therefore arises naturally: for what other groups G can one prove a convergence result similar to Theorem 6.5?

Functions over Products of Probability-Spaces

The Rademacher Projection. Let $\{\chi_S\}_{S\subseteq[n]}$ denote the set of p-biased Walsh products. For a real-valued function $\mathsf{f} : \{0,1\}^n \to \mathbb{R}$, whose Fourier expansion is given by $\sum_{S\subseteq[n]} \widehat{\mathsf{f}}(S)\chi_S$, define $Rad(\mathsf{f}) \doteq \sum_{i\in[n]} \widehat{\mathsf{f}}(\{i\})\chi_{\{i\}}$. In the case where $p = 1/2$, and for $1 < q < 2$, the proof in [Bou01] uses the inequality

$$(q-1)^{1/2}\|Rad(\mathsf{f})\|_2 \leq \|\mathsf{f}\|_q \tag{18.1}$$

It seems that if the same inequality were proven for the case where $p \neq 1/2$, the rest of the proof in [Bou01] would adapt easily to the biased case as well.

Once this is achieved, the parameters of Theorem 12.1 (page 124), and hence of Corollary 12.2 (page 125), would be improved to parameters similar to those in [Bou01]. This would lead to a two-query test for the biased long-code, similar to the two-point test which appears in [Kho02].

Generalized Bonami-Beckner Inequality. In the definition on page 132 of the Bonami-Beckner operator T_δ, both a combinatorial formula and a Fourier analytic formula for T_δ are given. It is easy to extend the combinatorial formula so that T_δ is applicable to functions defined over a general product of probability spaces. Hence it is natural to ask whether an analogue of Theorem 13.12 holds for the case where T_δ is applied to functions defined over such a domain.

We show (Theorem 13.13 in page 13.13) a special case of such an analogue, for the biased measure on the discrete cube. A full analogue in the case of the biased measure on the discrete cube, with tight parameters, was recently shown in [Ole02]. Such an analogue in the general case seems to imply the generalization of the results in [KKL88], [Fri98], and also of Theorem 12.1, to the case of boolean functions that are defined over general products of probability-spaces.

Generalized Fourier Expansion.

Using the notation of Part III, consider a function $\mathsf{f} : \mathcal{P}([n]) \to \mathbb{R}$ that is defined over the discrete hypercube, equipped with the p-biased measure. For a fixed set $S \subseteq [n]$ of coordinates, let T_S denote the orthogonal projection defined by $T_S(\mathsf{f}) \doteq \widehat{\mathsf{f}}(S)\chi_S$. It is not hard to find a combinatorial definition for this projection, similar to that of the Bonami-Beckner operator, using appropriate linear combinations of the averaging projections defined in Chapter 13. Once this is done, the operator T_S can be defined for functions over general products of probability-spaces.

Let $\mathcal{P}([n])$ now denote the product of some general probability-spaces $\Omega_1, \ldots, \Omega_n$ (we are now using the notation of Part II). One can verify that the projections $\{T_S\}_{S\subseteq[n]}$ form a spectral decomposition of the identity operator on $L_2(\mathcal{P}([n]))$, hence every function $\mathsf{f} : \mathcal{P}([n]) \to \mathbb{R}$ has a decomposition

$$\mathsf{f} = \sum_{S\subseteq[n]} T_S(\mathsf{f})$$

This decomposition may serve as a generalized Fourier expansion as far as variations and noise-sensitivity are concerned. For example, the Fourier analytic formula for the Beckner-Bonami operator has an analogue in the generalized product-space case using the above decomposition. The noise-sensitivity and the variation of a boolean function f also have formulas, similar to those in Proposition 13.11 (page 132) and Proposition 7.4 (page 92): to obtain these formulas one should simply replace $\widehat{\mathsf{f}}(S)^2$ by $\|T_S(\mathsf{f})\|_2^2$.

Low-weight on high-frequencies, and juntas. Using the above decomposition, it makes sense to define the weight of a given function f on frequencies higher than k, by $\sum_{|S|>k} \|T_S(\mathsf{f})\|_2^2$. It seems that proving an analogue of the Bonami-Beckner inequality in the case of functions defined over products of probability spaces, would lead to the generalization of the result in [FKN01], as well as of Theorem 12.3 (page 125).

Rademacher projection revisited. Using the above decomposition, one can also apply the analogue of the Rademacher projection, to functions defined over a general product of probability-spaces. This leads to the question of whether Inequality 18.1 has an analogue in the generalized case. If this is so, and the generalized Bonami-Beckner inequality can also be proven, it seems to imply that an analogue of the result of [Bou01] also holds in the generalized setting.

Other Tests

An $\mathsf{f}(x) \neq \mathsf{f}(x')$ test. Our characterization of juntas using noise-sensitivity implies that a boolean function f that satisfies $\mathsf{f}(x) = \mathsf{f}(x')$ with high probability, when x and x' are chosen randomly according to an appropriate distribution, must be close to a junta. It seems interesting, however, to also find a characterization of juntas of the form $\mathsf{f}(x) \neq \mathsf{f}(x')$. Specifically, a test for juntas that applies two queries to f and accepts if these queries yield different values, may be applicable to obtain a better hardness of approximation result for the Max-Cut problem.

Characterizing testable properties. It is somewhat ambitious, but very interesting, to ask the following question. Can one formulate conditions relating to the Fourier expansion of families of boolean functions, such that all families that satisfy them are testable using a number of queries that is independent of the number of coordinates?

Appendix A

Proof of Proposition 8.7

For $0 \leq x \leq t$, $\quad e^{-x/t} \leq 1 - \frac{x}{et}$. This holds since $e^{-x/t}$ is convex as a function of x, and since the inequality holds at the ends of the segment $[0, t]$. It follows that for all i,

$$\mathbb{E}\left[e^{-X_i/t}\right] \leq \mathbb{E}\left[1 - \frac{X_i}{et}\right] = 1 - \frac{\mathbb{E}[X_i]}{et}$$

Since the expectation is multiplicative for independent variables, we have

$$\mathbb{E}\left[e^{-X/t}\right] = \prod_{i=1}^{l} \mathbb{E}\left[e^{-X_i/t}\right] \leq \prod_{i=1}^{l} \left(1 - \frac{\mathbb{E}[X_i]}{et}\right)$$

We use the convexity of the above expression, together with the fact that $\sum_i \mathbb{E}[X_i] = \alpha$, and obtain

$$\mathbb{E}\left[e^{-X/t}\right] \leq \left(1 - \frac{\alpha}{elt}\right)^l \leq e^{-\alpha/et}$$

The Markov inequality now yields

$$\Pr[X \leq \eta\alpha] = \Pr\left[e^{-X/t} \geq e^{-\eta\alpha/t}\right] \leq \frac{e^{-\alpha/et}}{e^{-\eta\alpha/t}} = e^{\frac{\alpha}{et}(\eta e - 1)}$$

Bibliography

[ABMP98] M. Alekhnovich, S. Buss, S. Moran, and T. Pitassi. Minimum propositional proof length is NP-hard to linearly approximate. Manuscript, 1998.

[AD86] D. Aldous and P. Diaconis. Shuffling cards and stopping times. *American Mathematical Monthly*, 93(5):333–348, 1986.

[ALM$^+$98] 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, May 1998.

[AS97] Sanjeev Arora and Madhu Sudan. Improved low degree testing and its applications. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 485–495, El Paso, Texas, 4–6 May 1997.

[AS98] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs: a new characterization of NP. *Journal of the ACM*, 45(1):70–122, January 1998.

[AS00] Noga Alon and Joel H. Spencer. *The probabilistic method.* Wiley-Interscience [John Wiley & Sons], New York, second edition, 2000. With an appendix on the life and work of Paul Erdős.

[Bec75] W. Beckner. Inequalities in Fourier analysis. *Annals of Mathematics*, 102:159–182, 1975.

[BFL91] L. Babai, L. Fortnow, and C. Lund. Non-deterministic exponential time has two-prover interactive protocols. *Computational Complexity*, 1:3–40, 1991.

[BGLR93] M. Bellare, S. Goldwasser, C. Lund, and A. Russell. Efficient multi-prover interactive proofs with applications to approximation problems. In *Proc. 25th ACM Symp. on Theory of Computing*, pages 113–131, 1993.

[BGS98] Mihir Bellare, Oded Goldreich, and Madhu Sudan. Free bits, PCPs, and nonapproximability—towards tight results. *SIAM Journal on Computing*, 27(3):804–915, June 1998.

[BHL95] Avrim Blum, Lisa Hellerstein, and Nick Littlestone. Learning in the presence of finitely or infinitely many irrelevant attributes. *Journal of Computer and System Sciences*, 50(1):32–40, February 1995.

[BKS99] Itai Benjamini, Gil Kalai, and Oded Schramm. Noise sensitivity of Boolean functions and applications to percolation. *Inst. Hautes Études Sci. Publ. Math.*, (90):5–43, 1999.

[BL89] M. Ben-Or and N. Linial. Collective coin flipping. *ADVCR: Advances in Computing Research*, 5, 1989.

[BL96] N. Bshouty and N. Littlestone. Attribute-efficient learning in query and mistake-bound models. In *Proceedings of the* 9^{st} *COLT*, pages 235–243, 1996.

[Bou01] J. Bourgain. On the distribution of the fourier spectrum of boolean functions. to appear in Israel J. of Math., 2001.

[CG02] H. Chockler and D. Gutfreund. Property testing: Worst case vs. average case. manuscript, 2002.

[DFK$^+$99] I. Dinur, E. Fischer, G. Kindler, R. Raz, and S. Safra. PCP characterizations of NP: Towards a polynomially-small error-probability. In *Proc. 31th ACM Symp. on Theory of Computing*, 1999.

[DGL$^+$99] Yevgeniy Dodis, Oded Goldreich, Eric Lehman, Sofya Raskhodnikova, Dana Ron, and Alex Samorodnitsky. Improved testing algorithms for monotonicity. *Electronic Colloquium on Computational Complexity (ECCC)*, 6(017), 1999.

[DKRS98] I. Dinur, G. Kindler, R. Raz, and S. Safra. Approximating-CVP to within almost-polynomial factors is NP-hard. To appear in Combinatorica, 1998.

[DS98] I. Dinur and S. Safra. Monotone-minimum-satisfying assignment is NP-hard for almost polynomial factors. Manuscript, 1998.

[DS02] I. Dinur and S. Safra. On the importance of being biased. In *Proc. 34th ACM Symp. on Theory of Computing*, 2002.

[DT99] David Guijarro, Jun Tarui and Tatsuie Tsukiji. Finding relevant variables in PAC model with membership queries. In *Algorithmic Learning Theory, 10th International Conference, ALT '99, Tokyo, Japan, December 1999, Proceedings*, volume 1720 of *Lecture Notes in Artificial Intelligence*, pages 313–322. Springer, 1999.

[FGL$^+$91] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Approximating clique is almost NP-complete. In *Proc. 32nd IEEE Symp. on Foundations of Computer Science*, pages 2–12, 1991.

[Fis01] E. Fischer. The art of uninformed decisions: A primer to property testing. *The Bulletin of the European Association for Theoretical Computer Science*, 75:97–126, 2001.

[FKN01] E. Friedgut, G. Kalai, and A. Naor. Boolean functions whose fourier transform is concentrated on the first two levels and neutral social choice. To appear in Advances in Applied Mathematics, 2001.

[FKR$^+$] E. Fischer, G. Kindler, D. Ron, S. Safra, and A. Samorodnitsky. Testing juntas. To appear in FOCS 2002.

[FLN$^+$02] E. Fischer, E. Lehman, I. Newman, S. Raskhodnikova, R. Rubinfeld, and A. Samorodnitsky. Monotonicity testing over general poset domains. In *Proc. 34th ACM Symp. on Theory of Computing*, pages 474–483, 2002.

[Fri98] Ehud Friedgut. Boolean functions with low average sensitivity depend on few coordinates. *Combinatorica*, 18(1):27–35, 1998.

[FRS88] L. Fortnow, J. Rompel, and M. Sipser. On the power of multi-prover interactive protocols. In *Proceedings of the 3rd Conference on Structure in Complexity Theory*, pages 156–161, 1988.

[GGL$^+$00] Oded Goldreich, Shafi Goldwasser, Eric Lehman, Dana Ron, and Alex Samorodnitsky. Testing monotonicity. *Combinatorica*, 20(3):301–337, 2000.

[GGR98] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connections to learning and approximation. *Journal of the ACM*, 45(4):653–750, 1998.

[Hås97] Johan Håstad. Some optimal inapproximability results. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 1–10, El Paso, Texas, 4–6 May 1997.

[Hås99] Johan Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Math.*, 182(1):105–142, 1999.

[HPS93] J. Hastad, R. Phillips, and S. Safra. A well-characterized approximation problem. *Information Processing Letters*, 47:301–305, 1993.

[Kal01] G. Kalai. Social choice without rationality. To appear in Advances in Applied Mathematics, 2001.

[Kho02] Subhash Khot. On the power of unique 2-prover 1-round games. In *Proc. 34th ACM Symp. on Theory of Computing*, 2002.

[KKL88] J. Kahn, G. Kalai, and N. Linial. The influence of variables on Boolean functions. In IEEE, editor, *29th annual Symposium on Foundations of Computer Science, October 24–26, 1988, White Plains, New York*, pages 68–80. IEEE Computer Society Press, 1988.

[KS02] G. Kindler and M. Safra. Noise-resistant boolean functions are juntas. Manuscript, 2002.

[Lit87] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2:285, 1987.

[LY94] Carsten Lund and Mihalis Yannakakis. On the hardness of approximating minimization problems. *Journal of the ACM*, 41(5):960–981, 1994.

[Mar06] A. A. Markov. Extension of the law of large numbers to dependent events. *Bull. Soc. Phys. Math.*, 15(2):135–156, 1906.

[MOS02] E. Mossel, R. O'Donnell, and R. A. Servedio. Learning functions of k hidden variables. Manuscript, 2002.

[Ole02] Krzysztof Oleszkiewicz. On the non-symmetric khinchine-kahane inequality. Private communication, 2002.

[PRS01] M. Parnas, D. Ron, and A. Samorodnitsky. Proclaiming dictators and juntas or testing boolean formulae. In Proceedings of the 5$^{\text{th}}$ RANDOM conference, 2001. An improved version of this extended abstract will appear in *SIAM J. of Discrete Mathematics.*

[Raz98] Ran Raz. A parallel repetition theorem. *SIAM Journal on Computing*, 27(3):763–803, June 1998.

[Ron00] D. Ron. Property testing, 2000.

[RS92] R. Rubinfeld and M. Sudan. Testing polynomial functions efficiently and over rational domains. In *Proc. 3rd Annual ACM-SIAM Symp. on Discrete Algorithms*, pages 23–32, 1992.

[RS97] R. Raz and S. Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *Proc. 29th ACM Symp. on Theory of Computing*, pages 475–484, 1997.

[Sch99] Gideon Schechtman. Concentration, results and applications. *Preprint submitted to Elsevier Preprint*, 1999.

[Tal94] M. Talagrand. On Russo's approximate 0-1 law. *The Annals of Probability*, 22(3):1576–1587, 1994.

[UTW97] R. Uehara, K. Tsuchida, and I. Wegener. Optimal attribute-efficient learning of disjunction, parity and threshold functions. In Shai Ben-David, editor, *Proceedings of the 3rd European Conference on Computational Learning Theory*, volume 1208 of *LNAI*, pages 171–184, Berlin, March 17–19 1997. Springer.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de