On Obtaining Pseudorandomness from Error-Correcting Codes

Thesis by

Shankar Kalyanaraman

In Partial Fulfillment of the Requirements

for the Degree of

Master of Science

California Institute of Technology

Pasadena, California

2005

(Submitted June 2, 2006)

To ma, pa, Sriram and the fantastic summer of 2005 during which this thesis was written amid and in spite of many entreating distractions.

Abstract

Constructing pseudorandom objects based on codes has been the focus of some recent research. These constructions were based on specific algebraic codes and were rather simple in their structure in that a random index into a codeword was picked and m subsequent symbols output. In this work, we explore the question of whether it is possible to extend the scope of application of this paradigm of constructions to larger families of codes.

We show in this work that there exist such pseudorandom objects based on cyclic, linear codes that fool linear tests. When restricted to just algebraic codes, our techniques yield constructions that fool low-degree tests. Specifically, our results show that Reed-Solomon codes can be used to obtain pseudorandom objects, albeit in a weakened form. To the best of our knowledge, this is the first instance of Reed-Solomon codes being used to this effect.

In the process, we also touch upon one of the holy grails of derandomization. It should come as no surprise that pseudorandom objects that fool low-degree tests are automatically correlated to derandomizing polynomial identity testing. We look at whether our constructions are general enough to answer this important question and while we come up short in our endeavor, we believe our approach adds a new perspective to this problem and hopefully a meaningful opening to solving it.

Contents

Chapter 1

Introduction

Over the past decade, the study of randomness as a measure of complexity has come to play an increasingly crucial role alongside time and space. Without venturing into the foreign realms of physics and referring readers interested in any further discussion to [Cal04], we point out that sources of "pure" randomness are hard to find in nature and even sources such as Zener diodes or radioactive emissions by unstable elements like uranium cannot be relied upon to provide a long sequence of "truly random" bits. Subsequently, it is not hard to see why randomness is seen as another measure of complexity of an algorithm alongside time and space.

1.1 Computational and statistical indistinguishability

Fortunately for us, most algorithms that call for randomness are not too demanding and only require that the bits appear to be random and we shall refer to these bits as "pseudorandom" bits. In slightly more formal terms, we only require that the output of pseudorandom bits is "indistinguishable" from a stream of perfectly random bits. But in satisfying one requirement, we have raised another question – what do we mean by indistinguishability? The answer to this question comes in different flavors.

On the one hand we could require that any efficiently computable function upon input from a distribution over pseudorandom sequences of bits behaves almost identically to the case when the input drew from a uniformly random distribution and this is referred to as *computational indistinguishability.* On the other, we could ask for a stronger form of indistinguishability – referred to as *statistical indistinguishability* wherein the statistical distance between a distribution over pseudorandom sequences and a uniformly random distribution is negligibly small.

The statistical notion of indistinguishability is stronger than the computational because in some sense it requires indistinguishability even in regards to computationally inefficient functions. Two types of pseudorandom objects can be constructed based on these two ideas of indistinguishability both of which have found ample application in complexity theory.

1.1.1 Pseudorandom generators

Pseudorandom generators or PRGs are objects based on computational indistinguishability. A PRG takes as input a short random seed and obtains as output a longer pseudorandom string that is computationally indistinguishable from its uniformly random counterpart. Constructions of PRGs have been motivated from their applications in cryptography and complexity theory. Briefly, the main theme behind these constructions captures the requirement of computational indistinguishability by leveraging the hardness of a specific problem instance to obtain a PRG.

1.1.2 Extractors

Extractors on the other hand are pseudorandom objects based on statistical indistinguishability. They take as input a distribution which is not completely random and obtain a distribution that is statistically indistinguishable from a uniformly random distribution. They are so-called because they aim to "extract" all the randomness from the input distribution. Since their introduction in a paper by Nisan and Zuckerman [NZ96] less than a decade ago, extractors have come to be widely studied. This is in part because of the wide range of applications extractors lend themselves to including expander graph constructions [WZ93], derandomization of probabilistic algorithms [Zuc97, NZ96, INW94, RR99], constructions of codes [TSZS01, Gur04] and so on. The curious reader is referred to the Shaltiel's excellent survey on extractors [Sha02]. Since most of this work focuses on extractors, it will be worth our while to understand them in more detail and also to discuss various approaches to constructing them.

1.1.3 Parameters for extractors

An extractor takes as inputs a weak random source and a small completely random "seed". The seed is needed because it can be shown easily that in its absence even a single bit of randomness cannot be extracted from the weak source. We also need to quantify what exactly we mean when we say a source has "weak" randomness. To this end, we introduce the concept of *min-entropy*. For a probability distribution D and a random variable X sampled according to D, the min-entropy of D is defined as $\log(1/p^*)$ where

$$p^* = \max \Pr_D[X]$$

Therefore, a uniform distribution on $\{0,1\}^n$ has min-entropy n and a distribution on $\{0,1\}^n$ with min-entropy k satisfies $\Pr_D[x] \leq 2^{-k}$ for all x.

Ideally we would like to construct an extractor with a small seed and output length close to the min-entropy k of the input source. Radhakrishnan and Ta-Shma [RTS00] showed using the probabilistic method that there exist (k, ϵ) extractors with seed-length $\log n + 2\log(1/\epsilon) + O(1)$ and output length $m = k - O(\log(1/\epsilon))$ where k is the min-entropy of the input source and ϵ is how close the output distribution is statistically to the uniform distribution.

1.2 Historical overview of extractors

Some of the earliest constructions of extractors were based on families of hash functions [ILL89, SZ99] and the basic idea was that for a universal family $\mathcal{H}$ of hash functions $h : \{0,1\}^n \rightarrow \{0,1\}^l$, $h_y(x)$ is an $(l + 2\log(1/\epsilon), \epsilon)$ extractor. That is, $h_y(x)$ is ϵ-close to the uniform distribution on $\{0,1\}^l$ for x picked from a distribution with min-entropy $l + 2\log(1/\epsilon)$. "ϵ-closeness" is a measure of statistical distance that we explain in more detail subsequently. While these constructions were simple and clean, they were far from being optimal in seed-length which turned out to be larger than the randomness extracted. Some effort went into reducing the seed-length but the improvements were minor [SZ99].

Another line of work on extractors involved composing two extractors each working on small "blocks" of the weak random source. The relevant contribution in this regard came from a long series of papers beginning with Nisan and Zuckerman's work [NZ96, TS96, Zuc97].

1.2.1 Trevisan's breakthrough

A major breakthrough came in a paper by Trevisan [Tre01] who discovered an uncanny connection between some existing PRG constructions and extractors. Trevisan proposed constructions based on PRGs obtained by Nisan and Wigderson [NW94] and Impagliazzo and Wigderson [IW97]. We spend more time later illustrating the basic proof idea underlying Trevisan's constructions but in essence Trevisan exploited the hardness-randomness tradeoffs that were used to obtain PRGs in [NW94, IW97] and replaced the hard instances of problems in those constructions with weakly random sources to obtain an extractor instead.

1.2.2 Reconstruction proof paradigm

As clever as this idea is, the real legacy of Trevisan's work was the reconstruction proof technique which has been the centerpiece of subsequent extractor constructions

[TSZS01, SU01] and also forms the foundation of our work. Again, we will devote more time to this proof technique but we give some flavor of the technique at this point. If B denotes the set of all "bad" inputs where an input is "bad" if there exists a function that distinguishes it with non-negligible probability from a random input, Trevisan showed using the proof technique of [NW94, IW97] that B cannot be too big. Furthermore, if the input is drawn from a source with some min-entropy then the probability conditioned on the events $x \in B$ and $x \notin B$ that the output of the extractor is statistically distinguishable from a random sequence turns out to be small.

As mentioned earlier, Trevisan's work opened up a new way of thinking about extractors and fostered a long line of further improvements [RRV02, TSZS01, SU01]. Crucial among these were the near-simultaneous works by Ta-Shma, Zuckerman and Safra [TSZS01] and Shaltiel and Umans [SU01] that used the reconstruction proof paradigm to establish a fundamental connection between error-correcting codes and extractors. We will cover more on the reconstruction proof technique in §2.2 in Chapter 2.

1.3 Coding theory and pseudorandomness

Coding theory pertains to the study of error-correcting codes. Although the subject of coding theory may seem alien to theoretical computer science, ever since its (implicit?) involvement in the development of interactive proof systems coding theory, and in particular algebraic coding theory has become part of mainstream theoretical computer science [ALM$^+$98, BF90, BLR90]. Even prior to this however, codes have been used in connection with obtaining pseudorandomness. Naor and Naor [NN93] and Alon *et al* [AGHP92] used them to obtain k-wise ϵ-biased sample spaces. Alon *et al* [ABI86] showed the equivalence between l-wise independent sets and dual codes of distance $l + 1$.

Codes can also be used to improve PRG constructions. Blum and Micali [BM84] showed that good constructions of PRGs can be obtained using a one-way permutation and a corresponding hardcore predicate. The challenge then is to build a good hardcore predicate. For instance, Goldreich and Levin [GL89] used Hadamard codes to show the existence of a hardcore predicate. In general however, it can be shown that any good "list-decodable" code is a good candidate for obtaining such predicates. List-decodable codes are codes for which there exists an efficient decoding algorithm that retrieves a small list of possible codewords that are close to the received word. Sudan [Sud97] achieved a crucial breakthrough by showing that Reed-Solomon codes can be list-decoded beyond half the minimum distance. The notions of list-decodability and hardcore predicate bit construction tie in very closely with hardness amplification [BFNW93, IW97, STV01, Tre03]. We will review definitions from coding theory as well as the relevant algebra in Chapter 2.

1.3.1 Extractors from Reed-Müller codes

The extractor constructions given by Ta-Shma *et al* [TSZS01] and Shaltiel and Umans [SU01] use Reed-Müller codes which are algebraic codes. A message in a Reed-Müller code is interpreted as a multi-variate polynomial of small degree and the codeword is the evaluation of the polynomial over a field. We focus on the Shaltiel-Umans construction as it generalizes the approach taken in [TSZS01] and will describe it in slightly broader detail in a subsequent chapter but yield to the temptation of giving here a small sketch of how it works. The construction works in two stages. The input from the weak random source is taken to describe the co-efficients of a d-variate polynomial of degree h over a field $\mathbb{F}_q$ of size $q > 2$. Applying the Reed-Müller encoding to the polynomial we obtain a codeword in $\mathbb{F}_q^{q^m}$. The final output of the SU extractor consists of m "consecutive" elements starting from a position indexed by the random seed. For a code $\mathcal{C}$, we will use $f_{\mathcal{C},m}$ to denote this by

Definition 1

$$f_{\mathcal{C},m}(x,y) = (\mathcal{C}(x)[y+1], \mathcal{C}(x)[y+2], \ldots, \mathcal{C}(x)[y+m]) \tag{1.1}$$

Therefore, in the description above $\mathcal{C}$ is a Reed-Müller code. In the next stage, a binary code $\mathcal{C}'$ is used to encode each q-ary symbol in $f_{\mathcal{C},m}$. Finally, an additional random seed is used to index a single position in the binary codeword and output the corresponding binary symbol for each of the m q-ary symbols. This is given by

Definition 2

$$f'_{\mathcal{C},m,\mathcal{C}'}(x,y,z) = (\mathcal{C}'(\mathcal{C}(x)[y+1])[z], \mathcal{C}'(\mathcal{C}(x)[y+2])[z], \ldots, \mathcal{C}'(\mathcal{C}(x)[y+m])[z]) \tag{1.2}$$

1.3.2 Proof sketch for the Shaltiel-Umans extractor

In order to show that the construction above is indeed statistically indistinguishable from a uniformly random distribution on $\{0,1\}^m$ Shaltiel and Umans employ the reconstruction proof technique. At its simplest refinement, their proof assumes the contrary and constructs a function that takes as "advice" $m-1$ bits of the extractor and outputs the m-th bit with reasonably good success probability. By using a series of predictor functions whose inputs overlap with each other and using the list-decoding properties of the Reed-Müller code, we obtain the correct reconstruction of the entire codeword $\mathcal{C}(x)$ and subsequently x. But since x was picked from a distribution with some fixed min-entropy, if the total amount of advice used by the predictor is less than the min-entropy we will have obtained a contradiction since we were able to describe x with lesser information. The Shaltiel-Umans extractor will be discussed in more detail in §2.2.1 in Chapter 2.

1.4 Extractors from other codes?

In proving that the construction above is an extractor Shaltiel and Umans draw from many algebraic properties special to Reed-Müller codes. But is it possible to construct extractors without relying too heavily on such algebra? More generally, can we construct extractors from a larger class of codes?

We care to pose this question because of the numerous deep implications an answer in the affirmative holds with regard to other questions in complexity theory. Suppose for example that extractor constructions can be easily obtained from any linear code, this would demonstrate an intrinsic connection between pseudorandomness and algebraic coding theory because it would mean that linear codes are good pseudorandom objects. Furthermore, since such codes are well-understood owing to many decades of research invested into their study we can exploit this rich body of work towards refining our own understanding of the nature of pseudorandomness. Specifically, since there exists a plethora of explicit constructions of a variety of linear codes we would automatically obtain explicit extractor constructions. The wide scope of applications that extractors can be used in puts a further premium on this question.

But more than this, we would be interested in the methodology used to obtain pseudorandomness from codes. Our hope would be to exploit the reconstruction proof technique as applied to a larger body of codes than just Reed-Müller codes and achieve better parameters of extractors which we believe is possible by simplifying the technique.

We will leave the braver endeavor of seeking pseudorandomness from linear codes to subsequent pursuit and focus our attention in this work on a slightly smaller class of codes – cyclic linear codes. This class contains some well-known codes such as Reed-Müller and Reed-Solomon codes and manages to retain some of the nice algebraic properties of the former that were used in [SU01] which gives us hope in finding a favorable resolution to our question.

1.4.1 Extractors fooling linear tests

Our attempts at answering this question form the core contents of this work. We first state a theorem that meets this question part of the way where we show that cyclic linear codes beget good constructions for extractors that fool linear tests. For a cyclic linear code $\mathcal{C}$ with relative minimum distance δ and for some positive integer $m > 0$, let $f_{\mathcal{C},m}$ be as defined in (1.1).

Theorem 3 *For any k and $\rho > 0$, $f_{\mathcal{C},m}$ is a (k, ρ) q-ary extractor for the family of all linear tests, provided that $\delta > 1 - \rho/2$, and $k > m + \log(2/\rho)$.*

To prove this theorem, we resort to the reconstruction proof methodology but we adapt it to the case of cyclic linear codes. The proof sketch that we saw earlier spent some effort in going from a reasonably correct next-element predictor to an errorless prediction of the entire codeword. This involved using a set of predictor functions that overlapped at common points and then using the error-correcting properties of the Reed-Müller code to obtain the correct evaluation of the symbol before moving on to predict the next. This cumbersome procedure requires more advice bits which directly affects the output length via the contradiction that we hope to obtain. Furthermore, since the construction is based on a multivariate polynomial defined on a field, describing a random codeword symbol requires more in terms of seed-length as well. This results in a trade-off in the final extractor parameters and hence a less than optimal construction.

In our approach we eschew this expensive conversion from an error-prone predictor to an errorless predictor. Namely we show that for a cyclic linear code, if there exists a linear predictor with good success probability then it is *automatically* an errorless linear predictor. This avoids the complicated transformation in [SU01] and gets us the desired improvement in parameters with the caveat being however that it works only for a restricted class of tests. We cover this material in detail in §4.1 in Chapter 4.

1.4.2 Extractors fooling low-degree tests

Suppose now that we wish to broaden our class of linear prediction tests to include all tests of fixed degree, namely low-degree tests. Are cyclic linear codes still good candidates for obtaining extractors that fool all low-degree tests? It turns out from the following theorem that a subclass of them are. Let $\mathcal{C}$ be a fixed Reed-Müller code that encodes all messages given by the co-efficients of an l-variate polynomial of total degree h.

Theorem 4 *For any k and $\rho > 0$, $f_{\mathcal{C},m}$ is a (k, ρ) q-ary extractor for the family of all degree d prediction tests, provided that $\rho > 2dh/q$, and $k > m + \log(2/\rho)$.*

The reader may suspect that since there already exist constructions of extractors from Reed-Müller codes that fool *all* tests, Theorem 4 is redundant and offers much less than what is known. However, what distinguishes the construction described in the aforementioned theorem from the Shaltiel-Umans extractor is that it uses once again our lighter-weight reconstruction proof technique and also does not make use of any algebraic properties exclusive to Reed-Müller codes. As an important consequence to this, Reed-Solomon codes which constitute a subfamily of Reed-Müller codes can also be used to obtain extractors that fool degree-d prediction tests. Extractors based on Reed-Müller codes fooling low-degree prediction tests will be discussed in §4.2 under Chapter 4.

1.4.3 Two-source extractors for linear and low-degree tests

Suppose instead of our regular model of a weak random source and a short uniform source as inputs for the extractor, we now consider two weak random sources. Such extractors are referred to as multiple-source extractors and have sparked a recent spate of developments [BIW04, BKS$^+$04, Raz05]. We look at the natural analog of the general case and wonder if we can extend our constructions to make them work for the two-source case. This turns out to be the case when $\mathcal{C}$ is a systematic Reed-Solomon code that encodes messages given by co-efficients of a univariate polynomial of degree h.

Theorem 5 *Fix k_1, k_2, n_1, n_2, ρ. For $h < (\rho/2) \cdot 2^{k_2}$ and $q = 2^{n_2}$ the function $f_{\mathcal{C},m}(x, y)$ is a (k_1, k_2, ρ) q-ary extractor for the class of linear prediction tests with $k_1 > m + \log(2/\rho)$.*

Theorem 6 *Fix k_1, k_2, n_1, n_2, ρ. For $h < (\rho/2d) \cdot 2^{k_2}$ and $q = 2^{n_2}$ the function $f_{\mathcal{C},m}(x, y)$ is a (k_1, k_2, ρ) q-ary extractor for the class of degree-d prediction tests with $k_1 > m + \log(2/\rho)$.*

We will discuss these constructions also in Chapter 4.

1.5 From extractors to pseudorandom sets

Let us recall the defining property of extractors. If E is a (k, ϵ) extractor taking as inputs random variables X and Y, then when X is sampled according to a distribution with min-entropy at least k and Y is sampled according to the uniform distribution the random variable $E(X, Y)$ describes a distribution that is ϵ-close to the uniform distribution. We can also capture this property in the following manner. For every test T that distinguishes the extractor output from a string chosen uniformly at random with success probability ϵ, the support of X for which this occurs is of size at most 2^k.

A pseudorandom set is a collection of strings with the property that a string chosen uniformly at random from the collection "ϵ-fools" all distinguishing tests; in other words there exists no distinguishing test with success probability ϵ for the uniform distribution on the pseudorandom set.

This distinction between extractors and pseudorandom sets is crucial to understand because the pseudorandom set enjoys a property that is stronger than what the extractor gives us. The extractor property only ensures that for every distinguishing test on the output distribution, the number of "bad" inputs – or inputs for which the test distinguishes with success probability at least ϵ – is small whereas the pseudorandom set property means that there exists a string which ϵ-fools *all* distinguishing tests. This also explains why unconditional pseudorandom sets against all efficient distinguishing tests are that much harder to construct. The best success we have had so far in this regard are conditional pseudorandom sets that fool distinguishing tests described by small circuits [NW94, SU01, Uma02].

Fortunately, in the case of linear and low-degree tests this is rendered quite simple. Since we showed that prediction tests with reasonably good success probability for the extractors in Theorems 3 and 4 are automatically errorless, we only need to fix the first input of the extractor suitably to ensure that there exists no such errorless predictor. Then, iterating over all possible choices of the seed we will have our required unconditional pseudorandom set for linear and low-degree tests.

1.5.1 Pseudorandom sets fooling linear tests

Our next theorem states that from a systematic cyclic linear code $\mathcal{C}$ with blocklength $\bar{n}$, relative minimum distance δ and containing the all-ones codeword, one can construct a pseudorandom set fooling all linear tests.

Theorem 7 *Let x be such that $\mathcal{C}(x)[1 \dots \bar{k}] = 0^{\bar{k}-1}1$. Then $\mathcal{S} = \{f_{\mathcal{C},\bar{k}-1}(x, y) : 1 \leq y \leq \bar{n}\}$ is a q-ary pseudorandom set that fools all linear predictors with success probability ρ, provided that $\rho \geq 1 - \delta$.*

This theorem acquires added relevance because using the standard q-ary to binary transformation that we described earlier, we can obtain their binary equivalents which are called ϵ-biased spaces. In particular, using good binary list-decodable codes we can construct ϵ-biased spaces of size $O(m\text{polylog}(m, 1/\epsilon)/\epsilon^3)$ over $\{0,1\}^m$. This compares favorably with earlier constructions obtained in [AGHP92] and [NN93] which have sizes $(m/\epsilon)^2$ and m/ϵ^c respectively where $4 < c < 5$ although these constructions also had the property of k-wise independence. We must note however that the ϵ-bias in our constructions follows naturally by dint of the choice of a cyclic linear code as we will see in Lemma 11 in Chapter 6. Our constructions are presented in better light in §5.1.2 under Chapter 5.

1.5.2 Pseudorandom sets fooling low-degree tests

Analogous to the previous theorem, we utilize Theorem 4 to obtain pseudorandom sets that fool low-degree tests. Suppose $\mathcal{C}$ is a systematic Reed-Müller code with blocklength $\bar{n}$ that encodes all messages given by the co-efficients of an l-variate polynomial of total degree h where $\binom{h+l}{l} = \bar{k}$.

Theorem 8 *Let x be such that $\mathcal{C}(x)[1 \ldots \bar{k}] = 0^{\bar{k}-1}1$. Then $\mathcal{S} = \{f_{\mathcal{C},\bar{k}-1}(x,y) : 1 \leq y \leq \bar{n}\}$ is a q-ary ρ-pseudorandom set for the class of all degree d predictors, provided that $\rho \geq dh/q$.*

Some comment is in order regarding what exactly is achieved by Theorem 8. It would appear that a pseudorandom set as obtained in the theorem fooling all low-degree prediction tests is tantamount to derandomization of polynomial identity-testing since no low-degree polynomial appears to distinguish between the pseudorandom set and a uniformly random set. This would indeed be true if instead of prediction tests, the pseudorandom set were to fool all degree-d "distinguishing" tests instead of prediction tests which is what the theorem makes possible. The distinction between the two kinds of tests is that a degree-d distinguishing test is of the form $p(x_1, \ldots, x_m)$ where p is a degree-d polynomial whereas a degree-d prediction test is of the form $p(x_1, \ldots, x_{m-1}) - x_m$.

Still, armed with a pseudorandom set that fools low-degree prediction tests it seems tantalizingly close to be able to construct a set that fools low-degree distinguishing tests. Yao's famous lemma [Yao82] gives a standard relation between a prediction test to a distinguishing test but unfortunately does not preserve the low-degree property we need. We shall therefore have to be content with claiming that Theorem 4 allows for derandomizing polynomial identity-testing for a restricted class of polynomials comprising of all polynomials that may be expressed easily as a prediction test. No doubt minor improvements are possible in the nature of how the prediction test can be expressed. For instance, it is trivial to extend the theorem to construct pseudorandom sets that fool low-degree prediction tests of the form $p(x_1, \ldots, x_{m-1}) - x_m^k$ where $k \leq d$ and indeed, we encapsulate this into a generalized framework of "partition" prediction tests. We defer a full treatment on this subject to Chapter 6.

Chapter 2

Preliminary Definitions

2.1 Definitions and notation

2.1.1 Definitions from probability theory

For a finite set S, we can talk of a random variable X that assumes values in S under some random experiment.

Definition 9 *For a random variable X, a* **(discrete) probability distribution** *D on a finite set S assigns to every element $x \in S$ a positive real in $[0, 1]$ denoted by $\Pr_D[X = x]$ and signifying the probability that X assumes the value x during the random experiment.*

For the purposes of this work, we will take a distribution automatically to mean a discrete distribution on a finite set.

Definition 10 *The* **min-entropy** *of a random variable X with distribution D is denoted $H_\infty(X)$ and given by $H_\infty(X) = \min_{x \in S} \log(1/\Pr_D[X = x])$.*

Definition 11 *Let P, Q be distributions over a set S. P is* **ϵ-close** *to Q if*

$$\sum_{x \in S} \left| \Pr_P[X = x] - \Pr_Q[X = x] \right| \leq 2\epsilon$$

or equivalently if

$$\max_{A \subseteq S} \left| \Pr_P[X \in A] - \Pr_Q[X \in A] \right| \leq \epsilon$$

Wherever it is clear, we will only implicitly refer to the underlying distribution associated with X and denote $\Pr[X = x]$ to mean the probability that X when sampled according to this distribution takes the value x.

2.1.2 Definitions from algebra

Definition 12 *A* **ring** $(R, +, *)$ *is an algebraic structure comprising of a (possibly infinite) set of elements* R *and two binary operators* $+, * : R \times R \rightarrow R$ *with the following properties*

- $(R, +)$ *forms an abelian group and hence satisfies associativity and commutativity with respect to* $+$*, contains an identity element* 0 *and has an inverse element* $-a$ *for every element* $a \in R$ *satisfying* $a + (-a) = 0$.
- $*$ *is associative:* $a * (b * c) = (a * b) * c,\ a, b, c \in R$
- $*$ *distributes over* $+$*:* $a * (b + c) = a * b + a * c,\ a, b, c \in R$.

In a *commutative* ring R, $*$ is commutative. The tuple $(\mathbb{Z}, +, *)$ is an example of a ring where $\mathbb{Z}$ is the set of all integers.

Definition 13 *A* **field** $(F, +, *)$ *is a commutative ring with the following additional properties*

- *There exists a multiplicative identity element denoted* $1 \in F, 1 \neq 0$ *satisfying* $a \cdot 1 = a, a \in F$.
- *Every non-zero element* $a \in F$ *has a multiplicative inverse denoted* $a^{-1} \in F$ *satisfying* $a \cdot a^{-1} = 1$.

An example of a field is the tuple $(\mathbb{C}, +, *)$ where $\mathbb{C}$ is the set of all complex numbers.

Definition 14 *The* **fundamental theorem of algebra** *states that every complex polynomial of degree* n *has exactly* n *roots. In other words, if* $p(x) = \sum_{k=0}^{n} a_k x^k$ *where* $a_0, \ldots, a_n$ *are complex numbers then there exist (not necessarily distinct) complex numbers* $\alpha_1, \ldots, \alpha_n$ *such that* $p(x) = (x - \alpha_1)(x - \alpha_2) \ldots (x - \alpha_n)$.

2.1.3 Definitions in pseudorandomness

Definition 15 *A* **distinguisher** with success rate ϵ *for a random variable* $X = (X_1, X_2, \ldots, X_m)$ *defined on* $\mathbb{F}_q^m$ *is a function* $f : \mathbb{F}_q^m \to \mathbb{F}_q$ *with the property that*

$$|\Pr[f(X) = 0] - \Pr[f(U_m) = 0]| \geq \epsilon$$

where U_m *is the uniform distribution on* $\mathbb{F}_q^m$.

Definition 16 *An* i^{th}**-element predictor** with success probability ρ *for a random variable* $X = (X_1, X_2, \ldots, X_m)$ *with distribution* D *defined on* $\mathbb{F}_q^m$ *is a function* $f : \mathbb{F}_q^{i-1} \to \mathbb{F}_q$ *such that:* $\Pr_D[f(X_1, \ldots, X_{i-1}) = X_i] \geq \rho$*. If* $\rho = 1$ *we say that* f *is* errorless.

We will be concerned with linear and low-degree distinguishers and predictors. Note that a linear function f satisfies the identities (i) $f(\sum_{j=1}^k x_j) = \sum_{j=1}^k f(x_j) - (k-1)f(0)$ and (ii) $f(\alpha x) = \alpha f(x) - (\alpha - 1)f(0)$ for any scalar α. A *homogeneous* linear function f has $f(0) = 0$.

Definition 17 *A* (k, ρ) q**-ary extractor for a family of predictors** $\mathcal{P}$ *is a function* $E : \{0,1\}^n \times \{0,1\}^t \to \mathbb{F}_q^m$ *such that for every random variable* X *with* $H_\infty(X) \geq k$*, there is no* i^{th}*-element predictor* $f \in \mathcal{P}$ *for* $E(X, U_t)$ *with success probability* ρ *for any* $i = 1, \ldots m$.

In our notation, the usual q-ary extractors (as defined in, e.g., [SU01]) are simply q-ary extractors for the family of all predictors. Rather than referring to PRGs directly we prefer to describe the set of strings they produce.

Definition 18 *A* q*-ary* ρ**-pseudorandom set for a family of predictors** $\mathcal{P}$ *is a multiset* $S \subseteq \mathbb{F}_q^m$ *with the property that there is no* i*-th element predictor* $f \in \mathcal{P}$ *with success probability* ρ *for* $i = 1, \ldots, m$ *for the random variable induced by the uniform distribution on* S.

2.1.4 Definitions from coding theory

Definition 19 *The* **Hamming distance** *between two strings $a, b \in \mathbb{F}_q^{\bar{n}}$ denoted $\Delta(a,b)$ is the number of places where the strings are distinct:*

$$\Delta(a,b) = |\{i|a[i] \neq b[i]\}|$$

The **weight** *of a string $s \in \mathbb{F}_q^{\bar{n}}$ is denoted $w(s)$ and given by $\Delta(s, 0^{\bar{n}})$.*

Definition 20 *An $[\bar{n}, \bar{k}, \bar{d}]$* **$q$-ary linear code** *is a subspace $\mathcal{C} \subseteq \mathbb{F}_q^{\bar{n}}$ of dimension $\bar{k}$ for which the Hamming distance between every pair $x, y \in \mathcal{C}$ is at least $\bar{d}$.*

Definition 21 *The* **rate** *of an $[\bar{n}, \bar{k}, \bar{d}]$ q-ary code $\mathcal{C}$ is given by $\frac{\bar{k}}{\bar{n}}$ while its* **relative distance** *is given by $\frac{\bar{d}}{\bar{n}}$.*

Since a linear code $\mathcal{C}$ describes a subspace, we can talk of an $(\bar{k} \times \bar{n})$-*generator matrix* G that satisfies $x \cdot G \in \mathcal{C}$ for all $x \in \mathbb{F}_q^{\bar{k}}$. We will call such an $x \in \mathbb{F}_q^{\bar{k}}$ a *message*. Given a message x, we will use $\mathcal{C}(x)$ to mean the x-th codeword in $\mathcal{C}$. All of the codes we consider in this work are equipped with efficient ways to compute G.

Definition 22 *A code $\mathcal{C}$ is* **systematic** *if for all $x \in \mathbb{F}_q^k$:*

$$\mathcal{C}(x)[1 \ldots k] = x$$

Definition 23 *A code $\mathcal{C}$ is* **cyclic** *if it satisfies the following condition:*

$$(c_1, c_2, \ldots, c_{\bar{n}-1}, c_{\bar{n}}) \in \mathcal{C} \Rightarrow (c_{\bar{n}}, c_1, c_2, \ldots, c_{\bar{n}-1}) \in \mathcal{C}.$$

We always treat the indices into a cyclic code modulo $\bar{n}$.

A specific family of q-ary codes we will use is that of Reed-Müller codes.

Definition 24 *The codewords of a* **Reed-Müller code with parameters** *ℓ, h are the evaluations of ℓ-variate polynomials of total degree at most h, on the points $\mathbb{F}_q^\ell \backslash \{0\}$.*

When $\ell = 1$ we get a *Reed-Solomon code with parameter h.* Note that a Reed-Müller code has distance $(1-h/q)q^l$. All of these codes are cyclic (for an appropriate ordering of $\mathbb{F}_q^\ell \setminus \{0\}$) and linear. The nullspace of a subspace $\mathcal{C} \subseteq \mathbb{F}_q^{\bar{n}}$ comprises all vectors $h \in \mathbb{F}_q^{\bar{n}}$ such that $h \cdot c = 0$ for all $c \in \mathcal{C}$.

Definition 25 *The* **parity-check matrix** *denoted H for an $[\bar{n}, \bar{k}, \bar{d}]$ q-ary linear code $\mathcal{C}$ is an $(\bar{n} - \bar{k} \times \bar{n})$ matrix whose rows contain a basis of the nullspace of $\mathcal{C}$. Therefore, for all $c \in \mathcal{C}$ $c \cdot H^T = 0^{\bar{n}-\bar{k}}$.*

2.2 Reconstruction proof technique via the Shaltiel-Umans extractor

In this section we give an overview of the Shaltiel-Umans extractor and the underlying application of Trevisan's reconstruction proof technique. Although understanding the Shaltiel-Umans extractor or the reconstruction proof technique is not necessary to absorb the contents of our work, it would help to put the ideas in context. Much of the following treatment is rehashed and adapted from [SU01].

2.2.1 The Shaltiel-Umans extractor

Let x be sampled from a distribution on $\{0,1\}^n$ with min-entropy k. We will first focus on the q-ary case, where q is a value to be determined shortly. In the first step, we encode x using a q-ary Reed-Müller code with parameters d, h on $\mathbb{F}_q^d$. The blocklength of the Reed-Müller code is therefore q^d. The random seed is used to pick an index $\vec{v} \in \mathbb{F}_q^d$ and the final output consists of m consecutive symbols of the codeword starting at the $\vec{v}$-th position. We have overloaded the meaning of consecutive here and we really mean picking successive multiples in exponents of $A\vec{v}$ where A is a generator for the multiplicative group of $\mathbb{F}_q^d$. Therefore, m consecutive symbols starting at the $\vec{v}$-th position means picking symbols indexed at $\{A^i\vec{v}\}_{0\leq i\leq m-1}$. Note that there is an isomorphism between $\mathbb{F}_{q^d}$ and $\mathbb{F}_q^d$ and hence A corresponds to some cyclic generator $\alpha \in \mathbb{F}_{q^d}$. Consequently, A has the property that for any $\vec{v} \in \mathbb{F}_q^d$,

$\{A^i\vec{v}|1 \leq i \leq q^d\} = \mathbb{F}_q^d \setminus \vec{0}$. Furthermore, such a generator matrix can be found in time polynomial in q^d.

Since we are interpreting all $x \in \{0,1\}^n$ as message strings to be encoded using the Reed-Müller code, $\{0,1\}^n$ should be contained in the message space comprising all d-variate polynomials of total degree h. So, we choose h in terms of n, d satisfying

$$n \leq \binom{h+d-1}{d} \tag{2.1}$$

The following theorem is adapted from Theorem 4.5 in [SU01]. Let $\mathcal{C}$ be a Reed-Müller code with parameters d, h satisfying (2.1) and A be the matrix defined above.

Theorem 26 *Let $E(x,\vec{v}) = (\mathcal{C}(x)[\vec{v}], \mathcal{C}(x)[A\vec{v}], \ldots, \mathcal{C}(x)[A^{m-1}\vec{v}])$. Then for all d, h satisfying (2.1), x sampled from any distribution over $\{0,1\}^n$ with min-entropy k and $\vec{v}$ sampled randomly from $\mathbb{F}_q^d$ E is a (k,ρ) q-ary extractor provided that $k > O(mhd\log q)$ and $q > O(dh/\rho^3)$.*

In our work, we will only be interested in giving a proof sketch and more importantly going over the reconstruction technique. We refer the reader to [SU01] for corollaries and full proof of the theorem.

2.2.2 Proof sketch for the Shaltiel-Umans extractor

At its simplest, the idea is essentially proof by contradiction. If E were not a (k,ρ) extractor, then there is a predictor with good success rate. Using this predictor, we can reconstruct the entire codeword and hence x with lesser information than its min-entropy which is a contradiction.

For the reconstruction procedure, using an averaging argument we can show that it suffices to work with a fixed value of x chosen from $\{0,1\}^n$.

Definition 27 *A string $x \in \{0,1\}^n$ is* **ρ-good for a function $P : \mathbb{F}_q^{m-1} \to \mathbb{F}_q$ and**

E if

$$\Pr_{\vec{v}\in\mathbb{F}_q^d}[P(E(x,\vec{v})_{1,\ldots,m-1}) = E(x,\vec{v})_m] \geq \rho/2$$

Here, P is a predictor function that takes as input the first $m-1$ symbols of $E(x,\vec{v})$ and returns an element of $\mathbb{F}_q$. The reconstruction procedure $R^P : \{0,1\}^l \rightarrow \{0,1\}^n$ is structured as follows: R has oracle access to the predictor function P and takes as input l bits of advice returning an n-bit output.

Definition 28 *R is a ρ-**good reconstruction on** E with advice size l if for any predictor function P and x that is ρ-good for P and E:*

$$\Pr[\exists z \in \{0,1\}^l, R^P(z) = x] \geq 1/2$$

where the probability is taken over the distribution of the random coin tosses of R.

There are two steps to proving Theorem 26 – show that there exists a ρ-good reconstruction on E and show that if this is so, then E is a (k,ρ) q-ary extractor for $k > l + \log(2/\rho) + 2$. We will get to the first step shortly but suppose for now that there is such a reconstruction procedure.

Claim 29 *Let X be a random variable with distribution $D, H_\infty(D) \geq k$. Then*

$$\Pr_D[X \textit{ is } \rho\textit{-good for } P \textit{ and } E] \leq \rho/2$$

Proof. Let $p = \Pr_D[X$ is ρ-good for P and $E]$. Since there is a ρ-good reconstruction R on E with advice size l, for any ρ-good x the following is true from Definition 28 for X sampled according to D:

$$\Pr_D[\exists z \in \{0,1\}^l; X = R^P(z)] \geq p/2 \tag{2.2}$$

Now, there is a fixing of the random coin tosses of R for which (2.2) holds with the probability taken over only D. Such an R has at most 2^l outputs each corresponding to the input $z \in \{0,1\}^l$ and each such output is assumed by X with probability at most 2^{-k} since D has min-entropy at least k. Hence,

$$\Pr_D[\exists z \in \{0,1\}^l; X = R^P(z)] \leq 2^{l-k} \tag{2.3}$$

Combining (2.3) with (2.2) for a fixed R, we get that $p/2 \leq 2^{l-k}$ and for $k > l + \log(2/\rho) + 2$ this gives us $p \leq \rho/2$. □

We will now describe how R is obtained in the following lemma adapted from Lemma 4.13 in [SU01].

Lemma 1 *For parameters as in Theorem 26, there exists a ρ-good reconstruction on E with advice size $O(mhd \log q)$.*

Proof. (Sketch) Let P be a function $P : \mathbb{F}_q^{m-1} \longrightarrow \mathbb{F}_q$ and let $x \in \{0,1\}^n$ be ρ-good for P and E. This means that over a choice of $\vec{u} \in \mathbb{F}_q^d$ P predicts the $\vec{u}$-th point evaluation of $\mathcal{C}(x)$ correctly using evaluations on $m-1$ previous points $A^{-m+1}\vec{u}, \ldots, A^{-1}\vec{u}$ with probability $\rho/2$. The crux of the proof now is to reuse iteratively P's predictions along with existing advice to obtain predictions for the evaluation on all points of the field given by $A^i\vec{u}$. If the predictor were errorless, we would be home free by just giving as advice the first $m-1$ evaluations to the predictor and using it to obtain the entire codeword $\mathcal{C}(x)$. However, it has success probability only $\rho/2$ and so we need to error-correct each successive prediction before feeding it back to obtain newer predictions. To this end, we define two sets of degree $2r-1$ curves as follows:

- Pick $2r$ random points $\vec{y_1}, \ldots, \vec{y_{2r}}$ and $2r$ random and distinct values $t_1, \ldots, t_{2r}$
- Define $p_1 : \mathbb{F}_q \longrightarrow \mathbb{F}_q^d$ by $p_1(t_i) = \vec{y_i}, i = 1, \ldots, 2r$
- Define $p_2 : \mathbb{F}_q \longrightarrow \mathbb{F}_q^d$ by $p_2(t_i) = A\vec{y_i}, i = 1, \ldots, r$ and $p_2(t_i) = \vec{y_i}, i = r+1, \ldots, 2r$
- Define $P_{2j+1} = A^j p_1, j = 1, \ldots, q^d$ and $P_{2j+2} = A^j p_2, j = 1, \ldots, q^d$

The relevance of setting up these sets of interleaving curves will be clear from the properties they possess which we will state without proof.

Property 1 *For all i, P_i is a degree-$2r-1$ polynomial.*

Property 2 *The set $\{P_i(w)|w \in \mathbb{F}_q\}$ is $2r$-wise independent.*

Property 3 *For any consecutive pair $\{P_{i-1}, P_i\}$ there are r random points of intersection.*

Property 4 *The codeword $\mathcal{C}(x)$ restricted to P_i denoted $\mathcal{C}(x)[P_i(.)]$ is a univariate polynomial of degree at most $(2r-1)(h-1)$.*

In all of the above, $r = c'd$ where c' is a constant whose value will be determined later.

To jumpstart the reconstruction, we will feed in advice containing all the coefficients of $\mathcal{C}(x)[P_i(.)]$ for $1 \leq i \leq 2m$ which will require at most $2hr \cdot 2m \cdot \log q = 4mhr \log q$ advice bits. Using this, we proceed to use the predictor P to predict on curves $P_i, i > 2m$. The following claim is adapted from [SU01] and stated without proof:

Claim 30 *With probability at least $1 - 1/O(q^d)$ over the coin tosses of R:*

$$\Pr_{\vec{u} \in P_i}[P(\mathcal{C}(x)[A^{-m+1}\vec{u}], \ldots, \mathcal{C}(x)[A^{-1}\vec{u}]) = \mathcal{C}(x)[\vec{u}]] \geq \rho^3/4$$

Since P_i has q points, Claim 30 tells us that with high probability over the coin tosses of R at least $\rho^3 q/4$ points in P_i are correctly evaluated by the predictor out of a possible q points. Our goal now is to be able to error-correct the predictor on P_i and for this, we make use of Sudan's list-decoding bound [Sud97] captured in the following lemma

Lemma 2 ([Sud97]) *Let f be a function evaluated on $\bar{n}$ points and S be the set of all polynomials of degree $\bar{d}$ that agree with f on $\bar{t} > (2\bar{n}\bar{d})^{\frac{1}{2}}$ points. Then $|S| \leq 2\bar{n}/\bar{t}$.*

By choosing a sufficiently large c', we can ensure that the condition $\bar{t} > \sqrt{2\bar{n}\bar{d}}$ is satisfied and hence we will have a list of $2\bar{n}/\bar{t} = 8/\rho^3$ such degree-$2rh$ polynomials of which one of them is the correct reconstruction of the polynomial on P_i. We could use additional advice that tells us which entry in the list corresponds to the correct evaluation but this is expensive. Since a list will be generated for each prediction step, we will need advice at every single instance and this leads to a blow-up in the total advice required which would run counter to our objective of obtaining a contradiction by describing x with a short advice string.

Fortunately, our choice of P_i helps us resolve this. Recall from Property 3 that for any consecutive P_{i-1} and P_i, there are r random points of intersection. The remainder of the proof is guided by consequent simple observations that we will not prove. The first one follows from Schwartz-Zippel:

Observation 2.3 *Two different degree-$2rh$ polynomials agree on at most a $2rh/q$ fraction of points*

Observation 2.4 *Since we have inductively assumed that a complete evaluation on P_{i-1} is already known before we proceed to obtain the evaluation on P_i, with probability at most $(2rh/q)^r = O(\rho^d)$, an "incorrect" polynomial would agree with the evaluation on P_i.*

Observation 2.5 *With probability at least $1 - O(\rho^d)$ over the random coin tosses of R a correct evaluation of P_i is obtained.*

Since there are $2q^d$ prediction steps corresponding to each curve P_i, applying the union bound on the total number of prediction steps gives us that with probability at least $1/2$, the entire reconstruction procedure is successful. □

As noted while proving Lemma 1 we would have loved to avoid going through the complicated machinery including using error-correcting codes, interleaving prediction steps and using $2r$-wise independent curves in order to make the transition from a predictor with moderate success probability to an errorless predictor. This is indeed what we achieve in the case of extractors for linear and low-degree prediction tests. Briefly and at the expense of belaboring the point, we are able to show that if there is a predictor with reasonably good success probability it is *automatically* an errorless predictor. This helps significantly in further improving the output length parameter.

Chapter 3

Brief description of results and statements of theorems and proof outlines

In this chapter, we provide a broad overview of our results and give a flavor of our proof techniques.

3.1 Extractors fooling linear tests

Recall the definition of $f_{\mathcal{C},m}$ from Chapter 1:

$$f_{\mathcal{C},m}(x,y) = (\mathcal{C}(x)[y+1], \mathcal{C}(x)[y+2], \ldots, \mathcal{C}(x)[y+m])$$

Namely upon inputs x, y $f_{\mathcal{C},m}$ outputs m consecutive symbols of the codeword $\mathcal{C}(x)$ starting at the $(y+1)$-th position. Our first result pertains to showing that if $\mathcal{C}$ is a systematic cyclic, linear code then $f_{\mathcal{C},m}$ fools linear tests when x is chosen from sufficiently high min-entropy distributions. We do so by employing the reconstruction proof paradigm, namely we show that for fixed x if the random variable $f_{\mathcal{C},m}(x, U_t)$ has a linear predictor p, then x has a short description. In this case p is a linear function for which:

$$p(\mathcal{C}(x)[y+1], \mathcal{C}(x)[y+2], \ldots, \mathcal{C}(x)[y+m-1]) = \mathcal{C}(x)[y+m] \tag{3.1}$$

with noticeable probability over the choice of y. If we succeed in showing that x has a description that is smaller in size than the min-entropy of the distribution it is chosen from, we will have arrived at a contradiction thereby proving that no such linear function p can exist. Our key observation in showing this is that:

Observation 3.2 *If $\mathcal{C}$ has sufficiently good distance, then p must be* errorless.

To prove this we first select a subset S of those y for which (3.1) holds. In Lemma 3 in Chapter 4 we will show that if $\mathcal{C}$ has sufficiently good distance, then the codeword symbol at an arbitrary position r may be expressed as a linear combination ℓ of the values of $\mathcal{C}(x)$ at the positions S:

$$\mathcal{C}(x)[r] = \ell(\mathcal{C}(x)[y])_{y \in S} = \sum_{y \in S} c_y \mathcal{C}(x)[y]$$

Since $\mathcal{C}$ is *cyclic*, this same equation holds for *every* cyclic shift; i.e. for all i:

$$\mathcal{C}(x)[r+i] = \ell(\mathcal{C}(x)[y+i])_{y \in S} = \sum_{y \in S} c_y \mathcal{C}(x)[y+i]$$

These equations together with (3.1), which holds for all $y \in S$, imply that (3.1) holds for r. Since r was arbitrary, we conclude that p is indeed errorless. From here, it is easy to see that x may be described by $\mathcal{C}(x)[1 \ldots m-1]$, since we can use p to obtain $\mathcal{C}(x)[m]$, and again to obtain $\mathcal{C}(x)[m+1]$, and so on, until we have $\mathcal{C}(x)$ in its entirety. Finally decoding $\mathcal{C}(x)$ recovers x.

3.2.1 q-ary extractors to binary extractors

The q-ary extractors described above can be used to obtain binary extractors fooling linear tests by a simple application of binary codes akin to the construction of concatenated codes. Once the specific q-ary extractor is obtained as above, we use a binary linear code $\mathcal{C}'$ with suitable parameters to encode each symbol of the m elements of the q-ary extractor output. Then, using a separate seed we index a random position in the binary codeword of each of the m symbols to obtain the final m-bit output given by:

$$h_{\mathcal{C},\mathcal{C}',m}(x, y \circ z) = (\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)_z, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_2)_z, \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_m)_z)$$

We argue that this fools all linear tests by once again using the reconstruction proof technique. Namely,

Observation 3.3 *If there is a binary linear test p satisfying*

$$p(h_{\mathcal{C},\mathcal{C}',m}(x, y \circ z)_1, \ldots, h_{\mathcal{C},\mathcal{C}',m}(x, y \circ z)_{m-1}) = h_{\mathcal{C},\mathcal{C}',m}(x, y \circ z)_m$$

with noticeable probability over the choice of z, then by virtue of our choice of $\mathcal{C}'$ p is in fact an errorless binary linear predictor.

In order to complete the proof we now need to show that there exists an errorless q-ary linear predictor for the original q-ary extractor. We accomplish this by choosing an appropriate representation of the q-ary symbol in binary and then showing that p's errorless prediction retrieves the original q-ary symbol with noticeable probability over the choice of y. Then, we proceed to apply the analysis of Observation 3.2 to assert that the predictor so obtained is in fact an errorless predictor.

The rest of the proof involves implementing a reconstruction procedure that iteratively invokes this predictor on $m - 1$ bits of advice that is given it. Since the predictor is errorless for a reasonably good fraction of x, the reconstruction procedure itself succeeds in recovering x with good success probability. In order to complete our proof by contradiction we argue that given the min-entropy of the distribution from which x is sampled and the relevant parameters of the extractor construction, such a reconstruction procedure cannot have existed and hence nor could the predictor.

3.4 Extractors fooling low-degree tests

We extend our treatment to low-degree tests and observe that a subclass of cyclic, linear codes namely *polynomial* cyclic, linear codes like Reed-Müller codes and Reed-

Solomon codes fools this subclass. Once again we are tempted to argue using the line of approach described in §3.1 that for a Reed-Müller code $\mathcal{C}$ with sufficiently good distance parameter, if there exists a low-degree test given by:

$$p(\mathcal{C}(x)[y+1], \mathcal{C}(x)[y+2], \ldots, \mathcal{C}(x)[y+m-1]) = \mathcal{C}(x)[y+m] \tag{3.2}$$

claiming to work with a reasonably good success rate then the test is indeed errorless.

This line generally works except for some modifications we need to make in order to show that the test is errorless for $\mathcal{C}$. We make use of the fact that $\mathcal{C}(x)$ is now itself a low-degree polynomial over a field $\mathbb{F}_q$ of size q. This means that there is a mapping between the index y and values for variables $(y_1, y_2, \ldots, y_l) \in \mathbb{F}_q^l$ for which $r_x(y_1, y_2, \ldots, y_l) \equiv \mathcal{C}(x)[y]$, where r_x is a low-degree polynomial depending on x. Furthermore, some basic properties of fields give us that for all i there is a low-degree polynomial $r_{x,i}$ for which $r_{x,i}(y_1, y_2, \ldots, y_l) \equiv \mathcal{C}(x)[y+i]$.

Now, we observe that the left-hand side of (3.2) is a low-degree polynomial in $y_1, y_2, \ldots, y_l$, as is the right hand side. Furthermore, if they agree with reasonably high probability the Fundamental Theorem of Algebra (Definition 14 in Chapter 2) tells us that they must be *equal.* This implies that p is an *errorless* predictor, since equation (3.2) holds for all y.

The ensuing steps that include building a reconstruction procedure are verisimilar to the corresponding arguments in favor of extractors fooling linear predictors.

3.5 Two-source extractors that fool linear and low-degree tests

Following the previous discussions, we can also extend our constructions to multiple-source extractors that fool restricted classes of prediction tests. When $\mathcal{C}$ is a systematic Reed-Solomon code, $f_{\mathcal{C},m}(x, y)$ is a (k, ρ) q-ary two-source extractor for low-degree tests. Our proof structure is nearly preserved from the discussions above. The only alteration that we need to make is that instead of requiring that $\mathcal{C}$ have good relative distance in order for a predictor to be errorless, we shall require that its dimension be small compared to the support of the second source.

3.6 Unconditional PRGs fooling linear and low-degree tests

We saw above that depending on our choice of $\mathcal{C}$, the function $f_{\mathcal{C},m}(\cdot,\cdot)$ fooled varying classes of prediction tests. Suppose that for a given code $\mathcal{C}$ one can identify a fixed "good" x for which $f_{\mathcal{C},m}(x,\cdot)$ fools *all* prediction tests of a certain class. We call the set $\{f_{\mathcal{C},m}(x,y)|y \in \{0,1\}^t\}$ a *pseudorandom set* against all prediction tests of the class.

One of the surprising side-effects of having transformations from a predictor to an errorless predictor like the ones we have is that it is easy to find this "good" x *unconditionally*. This is because we need only to find a codeword that cannot have an errorless predictor. Indeed any codeword beginning with 0^m1 will suffice. If such a codeword has an errorless predictor p, then that predictor must output 0 since

$$p(\mathcal{C}(x)[y+1], \mathcal{C}(x)[y+2], \ldots, \mathcal{C}(x)[y+m-1]) = \mathcal{C}(x)[y+m]$$

implies $p(0,0,0\ldots,0) = 0$ (when $y = 0$) and $p(0,0,0\ldots,0) = 1$ (when $y = 1$), a contradiction.

This gives a simple construction of pseudorandom sets fooling all linear tests from any cyclic code with good distance. We can even extend this to the binary case whence our constructions are better known as ϵ-biased spaces. These probability spaces find applications in a variety of other problems including derandomization, communication complexity and constructions of hash functions. The interested reader is referred to [NN93] for more information on these topics and other applications of ϵ-biased spaces. [NN93] also gave a good construction of k-wise independent ϵ-biased probability spaces. In particular, their construction of an ϵ-biased space had size $O(m/\epsilon^c)$ where $c \in (4,5)$ is a positive constant. Subsequently Alon *et al* [AGHP92] gave a construction of k-wise independent ϵ-biased spaces with size $(m/\epsilon)^2$.

We are also able to conclude that substrings of low-degree polynomials compose a pseudorandom set that fools low-degree prediction tests, giving a derandomization of polynomial identity testing for this restricted class of tests.

Chapter 4

Extractors fooling linear and low-degree tests

In this chapter, we discuss constructions of extractors that fool linear and low-degree tests. Our construction for binary extractors will involve a two-step process. In the first step we construct q-ary extractors that fool linear tests over a field $\mathbb{F}_q$ and in the second step we use the q-ary extractor to obtain a binary extractor using an auxiliary linear code. The rest of the chapter is correspondingly premised for low-degree tests.

4.1 q-ary Extractors that fool $\mathbb{F}_q$-linear tests

Our construction is similar in technique to [SU01]. We consider Definition 1.1 in Chapter 1 of $f_{\mathcal{C},m}$: a codeword picked from a cyclic linear code $\mathcal{C}$ with distribution as specified by the min-entropy of the source and output m successive symbols of the codeword starting from a random position in the codeword. For a code $\mathcal{C}$ we denote $\mathcal{C}(x)$ to be the codeword obtained by encoding the message string x and $\mathcal{C}(x)[i]$ the i^{th} symbol in $\mathcal{C}(x)$. The following results describe important properties of linear codes that will be useful to us in the development of our proof.

Lemma 3 *Let $\mathcal{C}$ be a q-ary $[n', k', \delta n']$-linear code over $\mathbb{F}_q$ and let $S = \{t_1, \ldots, t_s\} \subseteq [m]$ be such that $|S| \geq (1-\delta)n'+1$. Further, let $1 \leq r \leq m$ be some arbitrary position.*

Then there exists an s-tuple $(l_1, \ldots, l_s) \in \mathbb{F}_q^s$ *such that for every codeword* $\mathcal{C}(x)$

$$\mathcal{C}(x)[r] = \sum_{i=1}^{s} l_i \mathcal{C}(x)[t_i]$$

Proof. For the case $r \in S$, this is trivial. We only therefore need to prove the statement for $r \in \overline{S}$. Let $\mathcal{C}(X) = [\mathcal{C}(X_{\overline{S}}) \mid \mathcal{C}(X_S)]$ be the row vector denoting the n' symbols of $\mathcal{C}(x)$ permuted so that the first $|\overline{S}|$ symbols correspond to the indices $r \in \overline{S}$. Note that $|\overline{S}| \leq \delta n' - 1$.

We will think of these $|\bar{S}|$ symbols as unknowns that we want to express in terms of the $|S|$ known symbols. Consider the system of n' homogeneous linear equations in $|\overline{S}|$ unknowns obtained by writing down the parity checks according to the relation

$$\begin{aligned} H \cdot \mathcal{C}(X)^T = H \cdot [\mathcal{C}(X_{\overline{S}}) \mid \mathcal{C}(X)_S]^T &= 0 && (4.1)\\ \Rightarrow H \cdot [\mathcal{C}(X_{\overline{S}}) \mid 0]^T + H \cdot [0 \mid \mathcal{C}(X_S)]^T &= 0 && (4.2)\\ \Rightarrow H \cdot [\mathcal{C}(X_{\overline{S}}) \mid 0]^T &= -H \cdot [0 \mid \mathcal{C}(X_S)]^T && (4.3)\end{aligned}$$

The right hand side in (4.3) given by $-H \cdot [0 \mid \mathcal{C}(X_S)]^T$ is the column vector of terms that are linear combinations of symbols $\{\mathcal{C}(x)[s] \mid s \in S\}$ whereas $\mathcal{C}(X_{\overline{S}})$ is the row vector of the unknown symbols.We will need to make use of the following property of linear codes.

Proposition 4 *The parity-check matrix* H *of an* $[n', k', \delta n']$ *q-ary linear code* $\mathcal{C}$ *satisfies* $H \cdot x = 0$ *for* $x \in \mathcal{C}$ *and has the property that any* $\delta n' - 1$ *columns of* H *are linearly independent.*

Since $\mathcal{C}$ is a linear code it satisfies Proposition 4 and hence any $\delta n' - 1$ columns of H are linearly independent. In particular therefore, (4.3) represents a non-homogeneous system of equations in $|\bar{S}|$ unknowns with a unique non-trivial solution. This means that each unknown symbol can now be written as a non-trivial linear combination of the known symbols in S. Furthermore, since the coefficients of the linear combination in terms of the known symbols corresponding to S are independent of the codeword, the lemma holds for all codewords in $\mathcal{C}$. □

Proof. (of Proposition 4) Suppose that there exists some $l = \delta n' - 1$ columns of H that are linearly dependent:

$$\sum_{j=1}^{l} b_{i_j} \cdot \mathbf{c_{i_j}} = 0 \tag{4.4}$$

where $\mathbf{c_{i_1}}, \ldots, \mathbf{c_{i_l}}$ are the columns of H and $b_{i_j} \neq 0$ for $j = 1, \ldots, l$. We can now construct an $x \in \mathbb{F}_q^{n'}$ such that for any position $i_k = i_{l+1}, \ldots, i_{n'}, x[i_k] = 0$ and for $i_j = i_1, \ldots, i_l, x[i_j] = b_{i_j}$. x satisfies $x \cdot H = 0$ from (4.4) and hence $x \in \mathcal{C}$. But x has distance $\delta n' - 1$ which is less than the minimum distance of $\mathcal{C}$ which is a contradiction. □

Using Lemma 3 we are in good shape to prove an important result about linear predictors for the extractors mentioned above. The result roughly states that a "reasonably correct" linear predictor for our extractor is in fact exactly correct.

Lemma 5 *Let $\mathcal{C}$ be a q-ary $[n', k', \delta n']$ cyclic linear code with $1^{n'} \in \mathcal{C}$. Suppose for some x, P is a linear i^{th}-element predictor with success probability $\rho > (1 - \delta)$ for the distribution induced by $f_{\mathcal{C},m}(x, y) = (\mathcal{C}(x)[y + 1], \mathcal{C}(x)[y + 2], \ldots, \mathcal{C}(x)[y + m])$ over y chosen uniformly from $[n']$. Then, P is an errorless linear predictor.*

Proof. Let P predict $p \geq (1 - \delta)n' + 1$ positions accurately. More precisely, the set $S = \{k | P(\mathcal{C}(x)[k+1], \ldots, \mathcal{C}(x)[k+i-1]) = \mathcal{C}(x)[k+i]\}$ has at least $|S| \geq (1-\delta)n'+1$. Consider the evaluation of P at an arbitrary position $r + i$. Since $\mathcal{C}$ is a linear code, we may apply Lemma 3 and express $\mathcal{C}(x)[r + i]$ as a linear combination in $\{\mathcal{C}(x)[s+i] : s \in S\}$ say $\mathcal{C}(x)[r+i] = \sum_{s \in S} b_s \mathcal{C}[s+i]$. Moreover, $\mathcal{C}$ is cyclic and hence for all $k, \mathcal{C}(x)[r + i + k] = \sum_{s \in S} b_s \mathcal{C}(x)[s + i + k]$. Evaluating f for x_{l_r+i} and using the properties of a linear function:

$$
\begin{aligned}
& f(\mathcal{C}(x)[r+1],\ldots,\mathcal{C}(x)[r+i-1]) && (4.5)\\
= \; & f(\sum_{s\in S} b_s\mathcal{C}(x)[s+1],\ldots,\sum_{s\in S} b_s\mathcal{C}(x)[s+i-1]) && (4.6)\\
= \; & \sum_{s\in S} f(b_s\mathcal{C}(x)[s+1],\ldots,b_s\mathcal{C}(x)[s+i-1]) && \\
& -(|S|-1)f(0,\ldots,0) && (4.7)\\
= \; & \sum_{s\in S} b_s f(\mathcal{C}(x)[s+1],\ldots,\mathcal{C}(x)[s+i-1]) && \\
& -\sum_{s\in S}(b_s-1)f(0,\ldots,0)-(|S|-1)f(0,\ldots,0) && (4.8)\\
= \; & \sum_{s\in S} b_s f(\mathcal{C}(x)[s+1],\ldots,\mathcal{C}(x)[s+i-1])-\sum_{s\in S} b_s f(0,\ldots,0) && \\
& +|S|f(0,\ldots,0)-(|S|-1)f(0,\ldots,0) && (4.9)\\
= \; & \sum_{s\in S} b_s\mathcal{C}(x)[s+i]+(1-\sum_{s\in S} b_s)f(0,\ldots,0) && (4.10)\\
= \; & \mathcal{C}(x)[r+i]+0 && (4.11)\\
= \; & \mathcal{C}(x)[r+i] && (4.12)
\end{aligned}
$$

where (4.8) follows from the fact that f is a linear function. (4.11) follows from the fact that $1^{n'} \in \mathcal{C}$ and hence also satisfies the linear combination $\mathcal{C}(x)[r+i] = \sum_{s\in S} b_s\mathcal{C}(x)[s+i]$ from the final observation in Lemma 3. □

We now have a direct extractor construction following from Lemma 5. We describe our construction formally in the theorem below but first we will state and prove a very important lemma that will be used both in our theorem on extractors for linear predictors as well as low-degree predictors.

Lemma 6 *Let P_d be a next-element degree-d predictor with success probability ρ for the random variable $f_{\mathcal{C},m}(X,Y)$ induced by sampling X according to D and Y uniformly from $\{0,1\}^t$. Suppose there exists a function $R : \{0,1\}^m \to \mathbb{F}_q^{n'}$ with oracle* *access to P_d such that*

$$\Pr_D[\exists a \in \{0,1\}^m, R(a) = \mathcal{C}(X)] \geq \rho/2$$

Then the function $f_{\mathcal{C},m}(x, y)$ is a (k, ρ) q-ary extractor with seed length $t = \log n'$ and satisfying $k > m + \log(2/\rho)$ for all degree-d predictors.

Proof. There are at most 2^m inputs a to R and hence at most 2^m possible outputs $\mathcal{C}(x)$. Since X is sampled according to a distribution D with min-entropy k, the random variable $\mathcal{C}(X)$ assumes each such output $\mathcal{C}(x)$ with probability at most 2^{-k}. Applying the union bound,

$$\Pr_D[\exists a \in \{0,1\}^m, R(a) = \mathcal{C}(X)] \leq 2^{i-1}2^{-k} \leq 2^{m-k} \tag{4.13}$$

But from the lemma statement,

$$\Pr_D[\exists a \in \{0,1\}^m, R(a) = \mathcal{C}(X)] \geq \rho/2$$

Hence for k such that $2^{m-k} < \rho/2$ or equivalently for $k > m + \log(2/\rho)$, we arrive at a contradiction to our original assumption that $f_{\mathcal{C},m}$ is not a (k, ρ) q-ary extractor for degree-d tests. This completes the proof of the lemma. □

Theorem 31 *Let k, n, ρ be fixed. Let $\mathcal{C}$ be an $[n', k', \delta']$ q-ary systematic cyclic linear code $\mathcal{C}$ with $1^{n'} \in \mathcal{C}$ such that $k' \geq n$ and $\delta' > 1 - \rho/2$. Then the function $f_{\mathcal{C},m}(x, y)$ is a (k, ρ) q-ary extractor with seed length $t = \log n'$ for the class of linear predictors provided $k > m + \log\left(\frac{2}{\rho}\right)$.*

Proof. Suppose $f_{\mathcal{C},m}$ is not an extractor with the parameters as claimed. Let D be a distribution associated with the pair of independent random variables (X, Y) where X is sampled according to a distribution D' of min-entropy k on $\{0,1\}^n$ and Y is sampled according to the uniform random distribution on $\{0,1\}^t$. Then, there exists for some random variable X and some i, a linear i^{th}-element predictor P satisfying

$$\Pr_{D}[P(f_{\mathcal{C},m}(X,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(X,Y)_i] \geq \rho \tag{4.14}$$

We have the following claim:

Claim 32

$$\Pr_{X \leftarrow D'}[\Pr_{U_t}[P(f_{\mathcal{C},m}(X,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(X,Y)_i] \geq \rho/2] \geq \rho/2 \tag{4.15}$$

Proof. We use an averaging argument to prove the claim. We will call all values x that X assumes for which:

$$\Pr_{U_t}[P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i] \geq \rho/2$$

"good" and those x for which:

$$\Pr_{U_t}[P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i] < \rho/2$$

"bad". Then, by conditional probabilities and (4.14)

$$\begin{aligned} & \Pr_{D}[P(f_{\mathcal{C},m}(X,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(X,Y)_i] \geq \rho \\ = \; & \Pr_{U_t}[P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i | x \text{ is "bad"}] \cdot \Pr_{D'}[X = x] \\ + \; & \Pr_{U_t}[P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i | x \text{ is "good"}] \cdot \Pr_{D'}[X = x] \geq \rho \end{aligned} \tag{4.16}$$

But

$$\Pr_{U_t}[P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i | x \text{ is "bad"}] \cdot \Pr_{D'}[X = x] < \rho/2 \cdot 1 = \rho/2 \tag{4.17}$$

Combining (4.16) and (4.17), we get

$$\begin{aligned} & \Pr_{U_t}[P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i | x \text{ is ``good''}] \cdot \Pr_{D'}[X = x] \geq \rho/2 \\ \Rightarrow \ & \Pr_{D'}[X = x] \geq \rho/2 \end{aligned} \tag{4.18}$$

where in (4.18) x is "good". This completes the proof. □

Therefore for a $\rho/2$ fraction of X sampled according to D' we have a linear i^{th}-element predictor P with success probability $\rho/2$ over a uniformly random distribution of Y. Also from the statement of the theorem $\rho/2 \geq 1 - \delta'$ and hence this satisfies the condition in Lemma 5 to give us that P is a linear i^{th}-element predictor with success probability 1.

We now describe a reconstruction procedure based on P that will use m bits of advice to obtain the entire codeword $\mathcal{C}(x)$ with high probability. We will first fix a good x, namely a choice of x for which Lemma 5 holds and P is an errorless predictor. To jumpstart the predictor, we feed it with the first $i-1$ consecutive symbols of $\mathcal{C}(x)$ to give us $P(\mathcal{C}(x)[1], \ldots, \mathcal{C}(x)[i-1]) = \mathcal{C}(x)[i]$. We iterate over the next set of $i-1$ symbols: $P(\mathcal{C}(x)[2], \ldots, \mathcal{C}(x)[i]) = \mathcal{C}(x)[i+1]$,$P(\mathcal{C}(x)[3], \ldots, \mathcal{C}(x)[i+1]) = \mathcal{C}(x)[i+2]$ and so on and eventually at the end of all the prediction steps we will have obtained $\mathcal{C}(x)$ and hence x itself. To express this formally, we augment P and define a new function $R : \mathbb{F}_q^{i-1} \to \mathbb{F}_q^{n'}$ such that $R(a)$ first invokes P on the $i-1$ elements given by the advice string a and follows it up with successive iterations to obtain $\mathcal{C}(x)$ for "good" x. In other words, if a encoded $E(x, 1)$ then

$$\begin{aligned} R(a) \ &= \ (\mathcal{C}(x)[1, \ldots, i-1], P(\mathcal{C}(x)[1, \ldots, i-1]), \\ & \quad\ P(\mathcal{C}(x)[2, \ldots, i-1], P(\mathcal{C}(x)[1, \ldots, i-1])), \ldots) \end{aligned}$$

From (4.15), we get:

$$\Pr_D[\exists a \in \mathbb{F}_q^{i-1}, R(a) = \mathcal{C}(X)] \geq \rho/2 \tag{4.19}$$

In order to be able to apply Lemma 6 for linear or degree-1 predictors we need to modify R so that it takes as input m-bit binary strings. Here we are helped by the fact that $\mathcal{C}$ is a systematic cyclic linear code and hence, $\mathcal{C}(x)[1,\ldots,n] = x \in \{0,1\}^n$. Since we are free to choose our advice bits from any set of symbols of $\mathcal{C}(x)$ because P is errorless and hence works correctly for all such positions we choose the first $i-1 \leq n$ symbols of $\mathcal{C}(x)$ each of which belongs to $\{0,1\}$. Hence R is *automatically* the required reconstruction function that we can use to apply Lemma 6 and complete the proof. □

4.1.1 From q-ary extractors to binary extractors

Our reduction from the q-ary extractors to the binary case follows the main idea of combining the q-ary code $\mathcal{C}$ with a good binary code $\mathcal{C}'$ akin to [TSZS01, SU01]. While we require that $\mathcal{C}$ be cyclic and linear, we only need $\mathcal{C}'$ to be a systematic binary linear code. We assume that $\mathbb{F}_q$ is an extension field of $\mathbb{F}_2 = \{0,1\}$. Additionally $\mathbb{F}_q$ can also be thought of as a $\log q$-dimensional vector space $\mathbb{F}_2^{\log q}$ with co-ordinates in $\mathbb{F}_2$. We assume an arbitrary basis $\vec{e}_1,\ldots,\vec{e}_{\log q} \in \mathbb{F}_2^{\log q}$ for $\mathbb{F}_q$ and represent every $z \in \mathbb{F}_q$ as $z = (z_1,\ldots,z_{\log q}) = \sum_{i=1}^{\log q} z_i\vec{e}_i, z_i \in \{0,1\}$.

Our main goal would be to show that the existence of a linear predictor for the binary extractor would imply the existence of a linear predictor for the q-ary extractor. But first, we will introduce and prove a proposition that will be required in proving our subsequent theorem on binary extractors for linear prediction tests.

Proposition 7 *Let $P : \mathbb{F}_q^m \to \{0,1\}$ be a q-ary linear distinguisher for a distribution D over $\mathbb{F}_q^m$ with advantage ϵ. Then, there exists a q-ary linear m^{th}-element predictor for D, $P' : \mathbb{F}_q^{m-1} \to \mathbb{F}_q$ such that*

$$\Pr_D[P'(x_1,\ldots,x_{m-1}) = x_m] \geq \frac{1}{q} + \frac{\epsilon}{q-1}$$

and for the case $\frac{1}{q} \le \epsilon \le 1 - \frac{1}{q}$,

$$\Pr_D[P'(x_1, \ldots, x_{m-1}) = x_m] \ge \frac{1}{q} + \epsilon$$

Proof. (Of Proposition 7) Since P is a linear distinguisher, we may assume without loss of generality that it is of the form $P(x_1, \ldots, x_m) = -x_m + \sum_{i=1}^{m-1} C_i x_i + C_0$. By definition,

$$|\Pr_D[P(x_1, \ldots, x_m) = 0] - \Pr_{U_m}[P(x_1, \ldots, x_m) = 0]| \ge \epsilon$$

where U_m is the uniform probability distribution on $\mathbb{F}_q^m$. Note that $\Pr_{U_m}[P(x_1, \ldots, x_m) = 0] = 1/q$. Two cases arise, namely that $\Pr_D[P(x_1, \ldots, x_m) = 0] \ge 1/q + \epsilon$ or $\Pr_D[P(x_1, \ldots, x_m) = 0] \le 1/q - \epsilon$. In the first case $P'(x_1, \ldots, x_{m-1}) = C_0 + \sum_{i=1}^{m-1} C_i x_i$ is an m^{th}-element predictor with success probability $\frac{1}{q} + \epsilon$. In the second case from a simple pigeonhole argument, there exists some $v \in \mathbb{F}_q; v \ne 0$ for which

$$\Pr_D[P(x_1, \ldots, x_m) = v] \ge \frac{1}{q-1} \cdot (1 - \frac{1}{q} + \epsilon) = \frac{1}{q} + \frac{\epsilon}{q-1}$$

Choosing $P'(x_1, \ldots, x_{m-1}) = C_0 + \sum_{i=1}^{m-1} C_i x_i$ where $C_0' = C_0 - v$ gets us an m^{th}-element predictor with success probability $\frac{1}{q} + \frac{\epsilon}{q-1}$. For the special case when $\frac{1}{q} \le \epsilon \le 1 - \frac{1}{q}$, we note that $\Pr_{U_m}[P(x_1, \ldots, x_m) = 0] = \frac{1}{q} \le \epsilon$ and hence the distinguisher property implies that only the first case is possible. □

Theorem 33 *Let k, n, ϵ be fixed. Let $\mathcal{C}$ be an $[n', k', \delta']$ q-ary systematic cyclic linear code with $1^{n'} \in \mathcal{C}$. Let $f_{\mathcal{C},m}$ be a (k, ρ) q-ary extractor based on $\mathcal{C}$ with $t = \log n', k' \ge n, \delta' \ge 1 - \frac{\rho}{2}$ and $k > m + \log(2/\rho)$ guaranteed by Theorem 31 for linear predictors. Let $\mathcal{C}'$ be a systematic $[n'', \log q, \delta'' n'']$ binary linear code with $1^{n''} \in \mathcal{C}'$ and $\delta_1 > 1/2 - \epsilon/2$. The function $h_{\mathcal{C},\mathcal{C}',m} : \{0,1\}^n \times \{0,1\}^{t+t_1} \to \{0,1\}^m$ given by*

$$h_{\mathcal{C},\mathcal{C}',m}(x, y \circ z) = (\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)_z, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_2)_z, \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_m)_z)$$

is a (k, ϵ) binary extractor with $t_1 = \log n_1$ for the class of linear predictors provided $\epsilon > 2\rho$. Here $f_{\mathcal{C},m}(x,y)_i \in \mathbb{F}_2^{\log q}$ is written in terms of the basis $(\vec{e}_1, \ldots, \vec{e}_{\log q})$.

Proof. The proof is similar to that of Theorem 31. Let B' be a binary linear distinguisher with advantage ϵ. Let X be a random variable in $\{0,1\}^m$ sampled according to some distribution D with min-entropy k. We apply Proposition 7 to the case $q = 2$ and obtain a binary linear predictor B from B' satisfying

$$\Pr_{X \leftarrow D, Y \circ Z \leftarrow U_{t+t'}}[B(h(X, Y \circ Z)_1, \ldots, h(X, Y \circ Z)_{i-1}) = h(X, Y \circ Z)_i] \geq 1/2 + \epsilon$$

where $U_{t+t'}$ is the uniform probability distribution on $\{0,1\}^{t+t'}$. Again, by virtue of an averaging argument we get

$$\begin{aligned} \Pr_{X \leftarrow D, Y \leftarrow U_t}[\Pr_{Z \leftarrow U_{t'}}[B(h(X, Y \circ Z)_1, \ldots, h(X, Y \circ Z)_{i-1}) &= h(X, Y \circ Z)_i] \\ &\geq 1/2 + \epsilon/2] \\ &\geq \epsilon/2 \end{aligned}$$

This means that for at least an $\epsilon/2$ fraction of (x, y) that (X, Y) assumes for which

$$\Pr_{Z \leftarrow U_{t'}}[B(h(x, y \circ Z)_1, h(x, y \circ Z)_2, \ldots, h(x, y \circ Z)_{i-1}) = h(x, y \circ Z)_i] \geq 1/2 + \epsilon/2$$

B is a binary linear predictor with success probability $1/2 + \epsilon/2$ over Z. We will call such an (x, y) good.

Lemma 8 *For a good choice of (x, y), let B be a binary linear predictor with success probability at least $1/2 + \epsilon/2$ for the function*

$$h(x, y \circ z) = (\mathcal{C}'(f_{\mathcal{C},m}(x, y)_1)_z, \mathcal{C}'(f_{\mathcal{C},m}(x, y)_2)_z, \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x, y)_m)_z)$$

Then B is an errorless binary linear predictor for $h(x, y \circ z)$ over the distribution on $\{0,1\}^m$ induced by sampling z uniformly from U_{t_1}.

Lemma 9 *Let $P : \mathbb{F}_q^{i-1} \to \mathbb{F}_q$ be a function given by:*

$$\begin{aligned} P(f_{\mathcal{C},m}(x,y)_{1,\ldots,i-1}) &= (B(\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)_1, \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_{i-1})_1), \\ & \quad B(\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)_2, \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_{i-1})_2), \\ & \quad \ldots \\ & \quad B(\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)_{\log q}, \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_{i-1})_{\log q})) \end{aligned}$$

Then P is a q-ary linear predictor for $f_{\mathcal{C},m}$ with success probability $\epsilon/2$ over the distribution on $\mathbb{F}_q^m$ with $x \leftarrow X, y \leftarrow U_t$.

Lemma 9 tells us that there exists a q-ary linear predictor P with success probability at least $\epsilon/2 > 1/q + \epsilon$ over $x \leftarrow X, y \leftarrow U_t$ for $f_{\mathcal{C},m}$. But $f_{\mathcal{C},m}$ is a $(k, \frac{1}{q} + \epsilon)$ q-ary extractor for linear predictors and hence we apply Theorem 31 to get a contradiction. This completes the proof of Theorem 33. □

Proof. (**Of Lemma 9**) From Lemma 8, we know that B is an errorless binary linear predictor. Also $\mathcal{C}'$ is a systematic binary linear code and hence the message m corresponds to the first $\log q$ bits of the codeword $\mathcal{C}'(m)$. Since to start with, the message was $f_{\mathcal{C},m}(x,y)_i$ for some i, and B correctly predicts the entire codeword $\mathcal{C}'(f_{\mathcal{C},m}(x,y))_i$, P predicts $f_{\mathcal{C},m}(x,y)_i$ correctly for all "good" choices of (x,y). But the fraction of such good choices is at least $\epsilon/2$. Therefore, P is a q-ary predictor with success probability at least $\epsilon/2$. Moreover, P is linear since each of $f_{\mathcal{C},m}(x,y)_i$'s co-ordinates in $\{0,1\}$ is expressed as a linear function in terms of B. Formally, if $B(a_1, \ldots, a_{i-1}) = \sum_{k=1}^{i-1} b_k a_k$ then

$$P(f_{\mathcal{C},m}(x,y)_{1,\ldots,i-1})_j = \sum_k b_k \mathcal{C}'(f_{\mathcal{C},m}(x,y)_k)[j] + C, j = 1, \ldots, \log q$$

and hence $P(f_{\mathcal{C},m}(x,y)_{1,\ldots,i-1}) = \sum_k b_k f_{\mathcal{C},m}(x,y)_k + C \cdot \vec{1}$. □

Proof. (**Of Lemma 8**) We examine B's working at some good pair given by (x,y) and some arbitrary z. Note that since $\mathcal{C}'$ is also a linear code and since $\delta_1 > 1/2 - \epsilon/2$

or equivalently $1/2 + \epsilon/2 > 1 - \delta_1$ Lemma 3 applies. Let S be the set of all s for which B predicts the i^{th}-bit correctly. From the above discussion, $|S| > (1 - \delta_1)n_1$. Then $B(\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)[z], \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_{i-1})[z])$ can be written as

$$\begin{aligned}
& B(\sum_{s\in S} c_s\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)[s], \ldots, \sum_{s\in S} c_s\mathcal{C}'(f_{\mathcal{C},m}(x,y)_{i-1})[s]) \\
&= \sum_{s\in S} B(c_s\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)[s], \ldots, c_s\mathcal{C}'(f_{\mathcal{C},m}(x,y)_{i-1})[s]) - (|S|-1)B(0,\ldots,0) \quad (4.20)\\
&= \sum_{s\in S} c_s B(\mathcal{C}'(f_{\mathcal{C},m}(x,y)_1)[s], \ldots, \mathcal{C}'(f_{\mathcal{C},m}(x,y)_{i-1})[s]) \\
&\quad - \sum_{s\in S}(c_s - 1)B(0,\ldots,0) - (|S|-1)B(0,\ldots,0) \quad (4.21)\\
&= \sum_{s\in S} c_s\mathcal{C}'(f_{\mathcal{C},m}(x,y)_i)[s] + (1 - \sum_{s\in S} c_s)B(0,\ldots,0) \quad (4.22)\\
&= \mathcal{C}'(f_{\mathcal{C},m}(x,y)_i)[z] + 0 \quad (4.23)\\
&= \mathcal{C}'(f_{\mathcal{C},m}(x,y)_i)[z] \quad (4.24)
\end{aligned}$$

(4.20,4.21) follow from the linearity of B and (4.22) follows from the fact that for $s \in S$ B predicts correctly. (4.23) follows from the fact that $1^{n_1} \in \mathcal{C}'$. We have shown that for a fixed "good" choice of (x,y) B is an errorless linear predictor over the distribution of z thus completing the proof of the claim. □

4.1.2 Specific constructions of extractors for linear tests

Using a Reed-Müller codes in Theorem 31 and a Hadamard code in Theorem 33 we obtain the following corollary:

Corollary 34 *Fix $n, k, \rho > 1/k^{O(1)}$. Let $\mathcal{C}$ be a Reed-Müller code with parameters $l = \log n/\log k, h = k$. Let $\mathbb{F}_q$ be a field of order $q = 2k/\rho$. Let $\mathcal{C}'$ be a $[q, \log q, 1/2]$ binary Hadamard code. Then $h : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ as defined in Theorem 33 is a (k,ϵ) binary extractor for the class of linear prediction tests with seedlength $t = O(\log n), m \geq k - O(\log k)$ and $\epsilon > 2\rho$.*

Proof. With parameters $h = k, l = \log n / \log k$ for a Reed-Müller code, applying Theorem 31 gives us a (k, ρ) q-ary extractor for linear prediction tests where $m \geq k - O(\log k)$ and $t = O(\log n)$. Applying Theorem 33 to this gives us a binary (k, ϵ) extractor for linear prediction tests where $\epsilon > 2\rho$. □

We can improve the parameters further by using the concatenated codes constructed in [GS00] in place of Hadamard codes and a Reed-Solomon code in place of the Reed-Müller code.

Corollary 35 *Fix $n, k, \rho > 1/k^{O(1)}$. Let $\mathcal{C}$ be a Reed-Müller code with parameters $l = 1, h = n/\log q$ where $\mathbb{F}_q$ is a field of order $q = 2n/\rho$. Let $\mathcal{C}'$ be a $[(\log q/\rho)^{O(1)}, \log q, 1/2]$ binary linear code. Then $h : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ as defined in Theorem 33 is a (k, ϵ) binary extractor for the class of linear prediction tests with seedlength $t = \log n + O(\log \log n) + O(\log(1/\epsilon)) + O(1), m \geq k - O(\log n)$ and $\epsilon > 2\rho$.*

Proof. Using the Reed-Müller code with the parameters mentioned we obtain a q-ary (k, ρ) extractor for linear prediction tests with seedlength $t_1 = \log n + \log(2/\rho)$ and $m > k - O(\log n)$ from Theorem 31. To this extractor, we apply the construction from Theorem 33 to obtain a binary (k, ϵ) extractor for linear prediction tests with additional seedlength $t_2 = O(\log \log n) + O(\log(1/\epsilon))$. Hence, the combined seedlength is $t = t_1 + t_2 = \log n + O(\log \log n) + O(\log(1/\epsilon))$. □

4.2 q-ary Extractors fooling low-degree tests

In this section, we discuss constructions for extractors that fool low-degree tests. Before we set to explain in detail the proofs and constructions, it will be worthwhile to illustrate similarities and differences between these constructions and those seen in the previous section. In the latter case, constructions were based on the family of cyclic linear codes whereas in the present case we will describe constructions that are based on the more restricted subfamily of Reed-Müller codes. Additionally our constructions are restricted to larger alphabets ($q > 2$). However, the proof structure is generally intact in that we will again show that a low-degree predictor with "reasonably good" success rate is actually errorless.

4.2.1 Errorless predictors for low-degree tests

We first state and prove a lemma for Reed-Müller codes analogous to that of Lemma 5.

Lemma 10 *Let $\mathcal{C}$ be a q-ary $[n', k', \delta n']$ Reed-Müller code with parameters l, h. Suppose for some x, p is a degree-d i-th element predictor for $f_{\mathcal{C},m}(x, y)$ with success probability $\rho > dh/q$ over the choice of y then p is an errorless predictor.*

Proof. From our definition of a Reed-Müller code, we observe that $\mathcal{C}(x)$ is a polynomial in l variables of total degree h, $r_x : \mathbb{F}_q^l \to \mathbb{F}_q$ and its symbols correspond to the evaluation of r_x on all $y = (y_1, \ldots, y_l) \in \mathbb{F}_q^l$. Therefore, $\mathcal{C}(x)[y] = r_x(y_1, \ldots, y_l)$. Using the fact that there exists an isomorphism $\mathbb{F}_q^l \leftrightarrow \mathbb{F}_{q^l}$ and the existence of a cyclical ordering of elements in $\mathbb{F}_{q^l}$, we may write for each i $y + i = (y'_1, \ldots, y'_l) \in \mathbb{F}_q^l$ in terms of a linear transformation function $t_i : \mathbb{F}^l \to \mathbb{F}^l$ where $t_i(y_1, \ldots, y_l) = (y'_1, \ldots, y'_l)$ so that $\mathcal{C}(x)[y+i] = r_x(y'_1, \ldots, y'_l) = r_x(t_i(y_1, \ldots, y_l)) = r_{x,i}(y_1, \ldots, y_l)$ for some l-variate degree-h polynomial $r_{x,i}$.

From the statement of the lemma, we know that p satisfies

$$\Pr_{Y \in \mathbb{F}_q^l}[p(\mathcal{C}(x)[Y+1], \ldots, \mathcal{C}(x)[Y+i-1]) = \mathcal{C}(x)[Y+i]] = \rho > dh/q \qquad (4.25)$$

Consider the i-variate polynomial $Q(z_1, \ldots, z_i) = z_i - p(z_1, \ldots, z_{i-1})$. Q also is of degree d since it has degree 1 in z_i and p has degree $d > 0$. The polynomial $Q'(y_1, \ldots, y_l) = Q(r_{x,1}, \ldots, r_{x,i})$ has total degree at most dh since each of $r_{x,1}, \ldots, r_{x,i}$ is a degree-h polynomial. Also, by construction Q' vanishes in exactly the points $y \in \mathbb{F}_q^l$ where p predicts accurately and so Q' vanishes on greater than dh/q fraction of points in $\mathbb{F}_q^l$. But by the Schwartz-Zippel lemma [Sch80, Zip79] we know that a non-zero polynomial of total degree dh can vanish in at most a dh/q fraction of points in $\mathbb{F}_q^l$ and hence Q' must be identically zero and vanish on all points in $\mathbb{F}_q^l$. Since Q'

vanishes in exactly those points where p predicts correctly, p must therefore be an errorless predictor. □

4.2.2 Main result

Theorem 36 *Let $\mathcal{C}$ be an $[\bar{n}, \bar{k}, \delta\bar{n}]$ q-ary systematic Reed-Müller code with parameters ℓ, h. For any k and $\rho > 0$, $f_{\mathcal{C},m}$ is a (k, ρ) q-ary extractor for the family of all degree d prediction tests, provided that $\rho > 2dh/q$, and $k > m + \log(2/\rho)$.*

Proof. By way of contradiction assume that $f_{\mathcal{C},m}$ is not a q-ary (k, ρ) extractor for all degree-d prediction tests. Then, there exists some random variable X with distribution D and min-entropy at least k such that there exists an i^{th}-element predictor P of degree at most d with success probability at least ρ over D and a random choice of $y \in \mathbb{F}_q^l$. In other words,

$$\Pr_{X \leftarrow D, Y \in \mathbb{F}_q^l} [P(f_{\mathcal{C},m}(X,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(X,Y)_i] \geq \rho$$

By an averaging argument identical to that in Claim 32, we have

$$\Pr_{X \leftarrow D} \left[\Pr_{Y \in \mathbb{F}_q^l} [P(f_{\mathcal{C},m}(X,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(X,Y)_i] \geq \rho/2 \right] \geq \rho/2 \tag{4.26}$$

As before, for at least a $\rho/2$ fraction of x P predicts the i^{th}-element of $f_{\mathcal{C},m}(x,Y)$ with probability at least $\rho/2 > dh/q$ over the values Y assumes. We will call all such values x that X assumes as good. Note that $P : \mathbb{F}_q^{i-1} \rightarrow \mathbb{F}_q$ is an $(i-1)$-variate polynomial of total degree at most d and hence from Lemma 10 P is an errorless predictor for the random variable Y:

$$\Pr_{Y \in \mathbb{F}_q^l} [P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i] = 1 \tag{4.27}$$

The reconstruction procedure is now identical to that described earlier in the proof of Theorem 31. We jumpstart the predictor P by feeding as advice the first

m consecutive symbols of $\mathcal{C}(x)$. We reuse subsequently predicted values as input to repeated invocations of P. Since for good x, P is an errorless predictor we will thus be able to retrieve the entire codeword. To express this formally, we define $R : \mathbb{F}_q^{i-1} \to \mathbb{F}_q^q$ given by

$$\begin{aligned} R(a) = (&\mathcal{C}(x)[1, \ldots, m-1], P(\mathcal{C}(x)[1, \ldots, m-1]),\\ &P(\mathcal{C}(x)[2, \ldots, m-1], P(\mathcal{C}(x)[1, \ldots, m-1])), \ldots) \end{aligned}$$

Using (4.26), we can analyze R's success rate:

$$\Pr_D[\exists a \in \mathbb{F}_q^{i-1}, R(a) = \mathcal{C}(X)] \geq \rho/2 \tag{4.28}$$

As before we started off with a systematic cyclic linear code, and so $\mathcal{C}(x)[1, \ldots, n] = x \in \{0,1\}^n$. Since we are free to choose our advice bits from any set of symbols of $\mathcal{C}(x)$ because P is errorless for our choice of x and hence works correctly for all such positions we choose the first $i-1 \leq n$ symbols of $\mathcal{C}(x)$ each of which belongs to $\{0,1\}$. Hence R is a function that has oracle access to a degree-d predictor P and takes as input an m-bit binary string a. Furthermore, it satisfies the conditions of Lemma 6 and hence we may apply Lemma 6 to complete the proof. □

4.2.3 Specific constructions of extractors for low-degree tests

We look at specific constructions of extractors fooling low-degree prediction tests. To this end we make use of Reed-Solomon codes. Recall that these are Reed-Müller codes with $l = 1$.

Corollary 37 *Fix n, k, d and $\rho > 1/k^{O(1)}$. Let $\mathcal{C}$ be a q-ary systematic Reed-Solomon code with parameters $q = 2dn/\rho$ and $h = n$. Then $f_{\mathcal{C},m}$ is a (k, ρ) q-ary extractor for the family of all degree d prediction tests, with seed length* $\log n + \log(2d/\rho)$ *and output length $m = k - \log(2/\rho)$.*

Proof. This follows from a straightforward application of Theorem 36 by substituting $h = n$. The seedlength is given by $t = \log n + \log(2d/\rho)$. □

4.3 Two-source extractors for linear and low-degree tests

So far we have seen constructions for seeded extractors that combined a weak random source with a short random seed to obtain an output that fools restricted classes of tests. We can further relax the requirement of the seed being random and talk of extractors from two weak random sources.

Definition 38 *A* (k_1, k_2, ρ) **q-ary extractor for a family of predictors** $\mathcal{P}$ *is a function* $E : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \mathbb{F}_q^m$ *such that for random variables* X_1, X_2 *with* $H_\infty(X_1) \geq k_1, H_\infty(X_2) \geq k_2$*, there is no* i^{th}*-element predictor* $f \in \mathcal{P}$ *for* $E(X_1, X_2)$ *with success probability* ρ *for any* $i = 1, \ldots m$.

As before, the usual (k_1, k_2, ρ) q-ary extractors can be defined as (k_1, k_2, ρ) q-ary extractors for the family of all predictors. Multiple source extractors have attracted a lot of interest in the last few years and led to a number of recent results [BKS$^+$04, BIW04, Raz05]. In this section, we will see that our previous constructions for seeded extractors for restricted classes of tests can be easily extended to the case of two-source extractors.

In proving our results for seeded extractors in the previous section we made use of Lemma 6. But in proving the lemma, we made no use of the fact that Y was sampled uniformly from $\{0,1\}^t$ and hence we have the following corollary which follows instantly from that lemma for an $[n', k', d']$ q-ary cyclic, linear code $\mathcal{C}$.

Corollary 39 *Let* P_d *be a next-element degree-d predictor with success probability* ρ *for the random variable* $f_{\mathcal{C},m}(X_1, X_2)$ *induced by sampling* X_1 *according to* D_1 *and* X_2 *according to* D_2*. Suppose there exists a function* $R : \{0,1\}^m \to \mathbb{F}_q^{n'}$ *with oracle access* *to P_d such that*

$$\Pr_{D_1}[\exists a \in \{0,1\}^m, R(a) = \mathcal{C}(X_1)] \geq \rho/2$$

Then the function $f_{\mathcal{C},m}(x,y)$ is a (k_1,k_2,ρ) q-ary extractor with seed length $t = \log n'$ and satisfying $k_1 > m + \log(2/\rho)$ for all degree-d predictors.

4.3.1 Two-source extractors for linear tests

We first show that $f_{\mathcal{C},m}$ as described previously is a two-source extractor for linear prediction tests provided $\mathcal{C}$ satisfies some constraints.

Theorem 40 *Let k_1, k_2, n_1, n_2, ρ be fixed. Let $\mathcal{C}$ be a systematic Reed-Solomon code with parameter $h < (\rho/2)\cdot 2^{k_2}$ and $q = 2^{n_2}$. Then the function $f_{\mathcal{C},m}(x,y)$ is a (k_1,k_2,ρ) q-ary extractor for the class of linear predictors with $k_1 > m + \log(2/\rho)$.*

Proof. Assume for the sake of contradiction that with the given parameters $f_{\mathcal{C},m}$ is not a (k_1,k_2,ρ) q-ary extractor. Then there exists for some choice of the pair of independent random variables (X_1,X_2) sampled according to a distribution $D = (D_1,D_2)$ over $(\{0,1\}^{n_1},\{0,1\}^{n_2})$ and some $i, 1 \leq i \leq m$ a linear i^{th}-element predictor P satisfying

$$\Pr_D\left[P(f_{\mathcal{C},m}(X_1,X_2)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(X_1,X_2)_i\right] \geq \rho$$

Using the averaging argument as before, we see that

$$\Pr_{X_1 \leftarrow D_1}\left[\Pr_{X_2 \leftarrow D_2}[P(f_{\mathcal{C},m}(X_1,X_2)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(X_1,X_2)_i] \geq \rho/2\right] \geq \rho/2 \tag{4.29}$$

We will call all x that X_1 assumes for which the inner probability is greater than $\rho/2$ good. From (4.29), we know that this fraction is greater than $\rho/2$. Since D_2 has min-entropy k_2, its support is at least 2^{k_2}. Therefore, for good $x \in \{0,1\}^{n_1}$ there are at least $(\rho/2)\cdot 2^{k_2}$ choices of y that X_2 assumes out of a total of 2^{n_2} for which $P(f_{\mathcal{C},m}(x,y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,y)_i$. Hence,

$$\Pr_{Y \leftarrow U_{n_2}}\left[P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i\right] \geq (\rho/2)\cdot 2^{k_2-n_2}$$

Furthermore, from the theorem statement $\mathcal{C}$ has parameter $h < (\rho/2) \cdot 2^{k_2}$ but for a Reed-Solomon code $h = (1-\delta)q$ where $\delta \cdot q = \delta \cdot 2^{n_2}$ is the minimum distance of the code. Therefore,

$$\Pr_{Y \leftarrow U_{n_2}} [P(f_{\mathcal{C},m}(x,Y)_{1,\ldots,i-1}) = f_{\mathcal{C},m}(x,Y)_i] \geq (\rho/2) \cdot 2^{k_2 - n_2} > (1-\delta) \tag{4.30}$$

Lemma 5 applies and therefore P is errorless.

Proceeding as usual, we define a reconstruction procedure $R : \mathbb{F}_q^{i-1} \rightarrow \mathbb{F}_q^{2^{n_2}}$ as

$$\begin{aligned} R(a) = (&\mathcal{C}(x)[1,\ldots,m-1], P(\mathcal{C}(x)[1,\ldots,m-1]), \\ &P(\mathcal{C}(x)[2,\ldots,m-1], P(\mathcal{C}(x)[1,\ldots,m-1])),\ldots) \end{aligned}$$

Combining this with our previous observations about P being errorless for good x, we get

$$\Pr_{x \in X_1} [\exists a \in \mathbb{F}_q^{i-1}, R(a) = \mathcal{C}(x)] \geq \rho/2 \tag{4.31}$$

$\mathcal{C}$ is a systematic Reed-Solomon code and so our previous arguments in favor of picking advice bits from the first n symbols of the codeword still hold and hence R satisfies the conditions set forth in Corollary 39 and this completes the proof. □

4.3.2 Two-source extractors for low-degree tests

Theorem 40 can be easily extended to obtain two-source extractors that fool low-degree tests.

Theorem 41 *Let $k_1, k_2, n_1, n_2, d, \rho$ be fixed. Let $\mathcal{C}$ be a systematic Reed-Solomon code with parameter $h < (\rho/2d) \cdot 2^{k_2}$ and $q = 2^{n_2}$. Then the function $f_{\mathcal{C},m}(x,y)$ is a (k_1,k_2,ρ) q-ary extractor for the class of degree-d prediction tests with $k_1 > m + \log(2/\rho)$.*

Proof. (Sketch) The proof structure is almost exactly identical to that of Theorem 40 and so we shall only provide a sketch to the proof. In fact, the only thing that changes is that instead of appealing to Lemma 5, we shall use Lemma 10 instead to show that a degree-d prediction test P with success probability ρ is errorless if

$$(\rho/2) \cdot 2^{k_2} > (dh/q) \cdot q = dh$$

Since this is true from the theorem statement, P is errorless. The rest of the proof follows exactly as before: we construct a reconstruction procedure R that uses $(i-1)$ bits of advice and reconstructs a "good" x with probability $\rho/2$. So, for $\rho/2 > 2^{m-k_1} \Rightarrow k_1 > m + \log(2/\rho)$ we have a contradiction to our assumption of existence of a degree-d prediction test. □

Chapter 5

Pseudorandom sets for linear and low-degree tests

In the previous chapter we looked at extractors for two classes of prediction tests. As we have seen these extractors use a weak random source in addition to an auxiliary short random seed as inputs. In this chapter we will pose and answer the following question: is it possible to fix the input coming from the weak random source to obtain a pseudorandom object determined only by a random seed that describes a distribution that fools the same classes of prediction tests? In particular, we will focus on unconditional pseudorandom sets for these classes of prediction tests. We recall the definition of a pseudorandom set from Chapter 2.

Definition 42 *A q-ary ρ-***pseudorandom set for a family of predictors** $\mathcal{P}$ *is a multiset $S \subseteq \mathbb{F}_q^m$ such that there is no i-th element predictor $f \in \mathcal{P}$ with success probability ρ for any i for the random variable induced by picking an element uniformly at random from S.*

5.1 Pseudorandom sets for linear tests

Applying Definition 42 to the case of linear tests, we can talk of a q-ary ρ-pseudorandom set for linear prediction tests.

5.1.1 q-ary pseudorandom sets

Our first result is for constructing unconditional pseudorandom sets that fool linear prediction tests. We state the theorem below and proceed to prove it. The idea captured in the theorem is to choose an input x in such a way that is foolproof against any errorless linear predictor.

Theorem 43 *Let $\mathcal{C}$ be a systematic $[n', k', \delta n']$ q-ary cyclic linear code with $1^{n'} \in \mathcal{C}$. Let x be such that $\mathcal{C}(x)[1 \ldots k'] = 0^{k'-1}1$. Then $\mathcal{S} = \{f_{\mathcal{C},m}(x, y) : 1 \leq y \leq n'\}$ is a q-ary ρ-pseudorandom set for the class of all linear prediction tests for $\rho > 1 - \delta$ and $m = k' - 1$.*

Proof. Since $\mathcal{C}$ is systematic and of dimension k', its first k' symbols are the message symbols and correctly describe a codeword in $\mathcal{C}$. Let us assume that there exists an i-th element linear predictor P with success probability ρ satisfying

$$\Pr_y[P(f_{\mathcal{C},k'-1}(x, y)_{1,\ldots,i-1}) = f_{\mathcal{C},k'-1}(x, y)_i] \geq \rho$$

Since $\rho > 1 - \delta$, by Lemma 5 P is an errorless predictor and for every choice of y, upon input $f_{\mathcal{C},k'-1}(x, y)_{1,\ldots,i-1}$ P predicts $f_{\mathcal{C},k'-1}(x, y)_i$ correctly. In particular

$$P(\mathcal{C}(x)[k' - i], \ldots, \mathcal{C}(x)[k' - 2]) = \mathcal{C}(x)[k' - 1] \tag{5.1}$$

$$P(\mathcal{C}(x)[k' - i + 1], \ldots, \mathcal{C}(x)[k' - 1]) = \mathcal{C}(x)[k'] \tag{5.2}$$

But from our choice of $\mathcal{C}(x)$, the inputs to P in (5.1) and (5.2) are

$$(\mathcal{C}(x)[k' - i], \ldots, \mathcal{C}(x)[k' - 2]) = (\mathcal{C}(x)[k' - i + 1], \ldots, \mathcal{C}(x)[k' - 1]) = (0, \ldots, 0)$$

whereas $\mathcal{C}(x)[k' - 1] = 0$ and $\mathcal{C}(x)[k'] = 1$ which is a contradiction to our assumption.

□

By way of example, the following corollary gives a construction for such a pseudorandom set using Reed-Solomon codes.

Corollary 44 *Fix m, ρ. Let $\mathcal{C}$ be a systematic Reed-Solomon code with parameters h, q satisfying $q > h/\rho$ and $h = m - 1$. Let x be such that $\mathcal{C}(x)[1 \ldots m] = 0^{m-1}1$. Then the set $\mathcal{S}$ given by*

$$\mathcal{S} = \{f_{\mathcal{C},m}(x, y) : 1 \leq y \leq q\}$$

is a q-ary ρ-pseudorandom set in $\mathbb{F}_q^m$ of size q for the class of all linear prediction tests.

Proof. It suffices to check if $\mathcal{C}$ satisfies $\rho > 1 - \delta$. Since $\mathcal{C}$ is a Reed-Solomon code, its relative distance is given by $\delta = (q - m + 1)/q = 1 - (m - 1)/q > 1 - \rho$ from our choice of $q > (m - 1)/\rho$. □

We can also define pseudorandom sets that fool restricted classes of distinguishing tests.

Definition 45 *A vector $v \in \mathbb{F}_q^m$ is a (homogeneous)* **linear distinguishing test** *with success probability ρ' for a set $\mathcal{S} \subseteq \mathbb{F}_q^m$ if v satisfies*

$$\left| \Pr_{s \in \mathcal{S}}[s \cdot v = 0] - \Pr_{x}[x \cdot v = 0] \right| \geq \rho'$$

Correspondingly a q-ary ρ'-pseudorandom set $\mathcal{S}$ that fools the class of linear distinguishing tests satisfies

$$\left| \Pr_{s \in \mathcal{S}}[s \cdot v = 0] - \Pr_{x}[x \cdot v = 0] \right| < \rho'$$

for all $v \in \mathbb{F}_q^m$.

Corollary 46 *Let $\mathcal{C}$ and $\mathcal{S}$ be as defined above in Theorem 43. Then S is a q-ary ρ'-pseudorandom set for the class of linear distinguishing tests where $\rho' = (\rho - 1/q)(q - 1)$.*

Proof. From Proposition 7 in Chapter 4 we know that a linear distinguisher with advantage ϵ implies a linear predictor with success probability $1/q + \epsilon/(q - 1)$. Conversely therefore, if there exists no linear prediction test with success probability ρ for

$f_{\mathcal{C},m}$ then there exists no linear distinguishing test with advantage $(\rho - 1/q)(q-1)$.

□

5.1.2 Obtaining ϵ-biased spaces from q-ary pseudorandom sets

From q-ary pseudorandom sets for linear tests, we shift our attention to their binary counterparts.

Definition 47 *A multiset $\mathcal{S} \subseteq \{0,1\}^m$ is an ϵ-**biased space** if for $\mathbf{s}$ chosen uniformly at random from $\mathcal{S}$ and every non-zero $\mathbf{v} \in \{0,1\}^m$, the random variable $X = \mathbf{s} \cdot \mathbf{v}$ satisfies*

$$|\Pr[X = 0] - \Pr[X = 1]| < \epsilon$$

Our construction of ϵ-biased spaces is structured exactly along the same lines as our construction of binary extractors for linear prediction tests from q-ary extractors.

Theorem 48 *Let $\mathcal{C}_1$ be an $[n_1', k_1', \delta_1 n_1']$ q-ary systematic cyclic linear code and $\mathcal{C}_2$ be an $[n_2', \log q, \delta_2 n_2']$ binary systematic linear code. Set $m = k_1' - 1$ and define $\mathcal{S} = \{f_{\mathcal{C}_1,m}(x,y) : 1 \le y \le n_1'\}$. Define*

$$\mathcal{T} = \{(\mathcal{C}_2(s_1)[z], \mathcal{C}_2(s_2)[z], \ldots, \mathcal{C}_2(s_m)[z]) : (s_1, \ldots, s_m) \in \mathcal{S}, 1 \le z \le n_2'\}$$

Then $\mathcal{T}$ is a 4ϵ-biased space for $\delta_1 > 1 - \epsilon, \delta_2 > 1/2 - \epsilon$.

Proof. As usual, we proceed by assuming the contrary. Then by definition, there exists a $\mathbf{v} \in \{0,1\}^m$ such that

$$\left| \Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 0] - \Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 1] \right| \quad \ge \quad 4\epsilon \tag{5.3}$$

$$\Rightarrow \Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 0] - \Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 1] \ge 4\epsilon \quad \text{or} \quad \Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 0] - \Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 1] \le -4\epsilon \tag{5.4}$$

Since the two events are mutually exclusive and exhaustive,

$$\Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 0] + \Pr_{\mathbf{x} \in \mathcal{T}}[\mathbf{x} \cdot \mathbf{v} = 1] = 1 \tag{5.5}$$

Combining (5.4) and (5.5), we get

$$\Pr_{\mathbf{x}\in\mathcal{T}}[\mathbf{x}\cdot\mathbf{v}=0]\geq 1/2+2\epsilon \quad \text{or} \quad \Pr_{\mathbf{x}\in\mathcal{T}}[\mathbf{x}\cdot\mathbf{v}=0]\leq 1/2-2\epsilon \tag{5.6}$$

$$\Rightarrow \left|\Pr_{\mathbf{x}\in\mathcal{T}}[\mathbf{x}\cdot\mathbf{v}=0]-\Pr_{\mathbf{x}}[\mathbf{x}\cdot\mathbf{v}=0]\right| \quad \geq \quad 2\epsilon \tag{5.7}$$

(5.7) follows from the fact that $\Pr_{\mathbf{x}}[\mathbf{x}\cdot\mathbf{v}=0]=1/2$ for any non-zero $\mathbf{v}$. This means that $\mathbf{v}$ is a linear distinguisher with success rate 2ϵ and by Proposition 7 in Chapter 4 gives an i-th bit binary linear predictor B for some i with success probability $1/2+2\epsilon$ for the m-bit random variable $\mathbf{x}$ over the distribution induced by picking uniformly at random from $\mathcal{T}$:

$$\Pr_{\mathbf{x}\in\mathcal{T}}[B(\mathbf{x}_{1,\ldots,i-1})=\mathbf{x}_i]\geq 1/2+2\epsilon \tag{5.8}$$

We can rewrite (5.8) as follows

$$\Pr_{y,z}[B(\mathcal{C}_2(f_{\mathcal{C}_1,m}(x,y)_1)[z],\ldots,\mathcal{C}_2(f_{\mathcal{C}_1,m}(x,y)_{i-1})[z])=\mathcal{C}_2(f_{\mathcal{C}_1,m}(x,y)_i)[z]] \geq 1/2+2\epsilon \tag{5.9}$$

Following a standard averaging argument, for an ϵ fraction of y

$$\Pr_{z}[B(\mathcal{C}_2(f_{\mathcal{C}_1,m}(x,y)_1)[z],\ldots,\mathcal{C}_2(f_{\mathcal{C}_1,m}(x,y)_{i-1})[z])=\mathcal{C}_2(f_{\mathcal{C}_1,m}(x,y)_i)[z]] \geq 1/2+\epsilon \tag{5.10}$$

Since $\delta_2 > 1/2-\epsilon$ from the theorem statement Lemma 8 from Chapter 4 holds and B is an errorless binary linear predictor for an ϵ fraction of y. From Lemma 9 again in Chapter 4, this implies a q-ary linear predictor P for $f_{\mathcal{C}_1,m}$ with success probability ϵ over the distribution induced by picking y uniformly. But from Theorem 31, for $\delta_1 > 1-\epsilon$ $f_{\mathcal{C}_1,m}$ is a (k,ϵ) q-ary extractor for all linear prediction tests which leads to a contradiction in our assumption. □

Corollary 49 *Fix m. Let $\mathcal{C}_1$ be a Reed-Solomon code with parameters m and $q >$*

m/ϵ and let $\mathcal{C}_2$ be a $[q, \log q, q/2]$ binary Hadamard code. Then the set $\mathcal{T}$ defined in Theorem 48 is a 4ϵ-biased space of size $O(m^2/\epsilon^2)$.

Proof. It is easy to see that the size of $\mathcal{T}$ is given by $n'_1 \cdot n'_2$ where n'_1, n'_2 are respectively the blocklengths of $\mathcal{C}_1, \mathcal{C}_2$. Hence, $|\mathcal{T}| = q^2 = m^2/\epsilon^2$. □

Corollary 50 *Fix m. Let $\mathcal{C}_1$ be a Reed-Solomon code with parameters m and $q > m/\epsilon$ and let $\mathcal{C}_2$ be an $[\bar{n} = O(\log^2 q/\epsilon^2), \log q, (1/2 - \epsilon)\bar{n}]$ binary linear code. Then the set T defined above is a 4ϵ-biased sample space of size $O(m\text{polylog}(m, 1/\epsilon)/\epsilon^3)$.*

5.2 Pseudorandom sets for low-degree prediction tests

We extend our treatment of pseudorandom sets to low-degree prediction tests.

Theorem 51 *Fix m, ρ. Let $\mathcal{C}$ be a systematic q-ary Reed-Müller code with parameters h, l satisfying $\binom{h+l-1}{h} = m$ and $q \geq dh/\rho$. Let x be such that $\mathcal{C}(x)[1 \dots m] = 0^{m-1}1$. Then $\mathcal{S} = \{f_{\mathcal{C},m}(x, y) : 1 \leq y \leq q\}$ is a q-ary ρ-pseudorandom set for the class of degree d predictors.*

Proof. Since $\mathcal{C}$ is systematic and its dimension is exactly m, the codeword $\mathcal{C}(x)$ with the aforementioned properties indeed exists. Assuming contrariwise that there exists an i-th element degree-d prediction test p for the uniform distribution on $\mathcal{S}$, we use Lemma 10 from Chapter 4 to claim that p is an errorless predictor. In particular

$$
\begin{aligned}
p(\mathcal{C}(x)[m-i], \dots, \mathcal{C}(x)[m-2]) &= \mathcal{C}(x)[m-1] && (5.11) \\
p(\mathcal{C}(x)[m-i+1], \dots, \mathcal{C}(x)[m-1]) &= \mathcal{C}(x)[m] && (5.12)
\end{aligned}
$$

but

$$(\mathcal{C}(x)[m-i], \dots, \mathcal{C}(x)[m-2]) = (\mathcal{C}(x)[m-i+1], \dots, \mathcal{C}(x)[m-1]) = (0, \dots, 0)$$

while $\mathcal{C}(x)[m-1] = 0$ and $\mathcal{C}(x)[m] = 1$. This gives a contradiction to our original assumption. $\square$

Using Reed-Solomon codes, we can obtain specific constructions of pseudorandom sets for low-degree tests.

Corollary 52 *Fix m, ρ. Let $\mathcal{C}$ be a systematic Reed-Solomon code with parameters h, q satisfying $h = m-1, q = dh/\rho$. Then the set $\mathcal{S}$ as described in Theorem 51 is a q-ary ρ-pseudorandom set in $\mathbb{F}_q^m$ of size $d(m-1)/\rho$ for the class of all degree d prediction tests.*

Proof. It suffices to comment only on the size of $\mathcal{S}$ which is given by the blocklength of $\mathcal{C}$. Since for a Reed-Solomon code, the blocklength is q, $|\mathcal{S}| = q = d(m-1)/\rho$. The rest of the proof structure is intact from above. $\square$

Chapter 6

Review

6.1 Revisiting extractors and pseudorandom sets

At the start of this work, we introduced notions of statistical and computational indistinguishability. We then connected these notions to two types of pseudorandom objects, namely extractors and pseudorandom generators. Any function with a small circuit size achieves limited success in distinguishing the output of a pseudorandom generator from that of a uniformly random distribution. In the case of the output of an extractor, no function regardless of circuit size is able to distinguish it from that of a uniformly random distribution with reasonable success.

In the pseudorandom objects that we have introduced in Chapters 4 and 5 however, this distinction is not very clear because a low-degree prediction test can be viewed both as a computationally limited function against which a pseudorandom generator is constructed and as belonging to a restricted class of functions against which an extractor is constructed.

As we saw before, a (k, ρ) q-ary extractor $f_{\mathcal{C},m} : \{0,1\}^n \times \{0,1\}^t \to \mathbb{F}_q^m$ for linear (low-degree) prediction tests satisfies the property that for x *sampled from a distribution with min-entropy at least* k over $\{0,1\}^n$ and y sampled from a uniformly random distribution over $\{0,1\}^t$, $f_{\mathcal{C},m}(x, y)$ induces a distribution over $\mathbb{F}_q^m$ that fools all linear (low-degree) prediction tests with success probability ρ. In contrast, a q-ary ρ-pseudorandom set $\mathcal{S}$ that fools all linear (low-degree) prediction tests is constructed by *fixing some suitably chosen* x and the distribution is induced by sampling y uniformly from $\{0,1\}^t$ has the property that it fools all linear (low-degree) prediction tests.

6.2 Revisiting ϵ-biased spaces

In Chapter 5 we described our construction of ϵ-biased spaces based on the binary extractors for linear tests discussed in Chapter 4. Specifically, our construction of an ϵ-biased space composed a q-ary systematic cyclic linear code with a binary systematic linear code. Depending on the choice of codes used, we obtained different parameters in our constructions as noted in Corollaries 49 and 50. While we would undoubtedly have liked to claim that the "bias" derives from how we construct the pseudorandom set by fixing a suitable codeword, it turns out that any $[\bar{n}, \bar{k}, \delta\bar{n}]$ binary linear code containing the all-ones codeword and whose codewords have relative distance $\delta \geq (1-\epsilon)/2$ automatically realizes an ϵ-biased space of size $\bar{n}$ on $\{0,1\}^{\bar{k}}$.

Lemma 11 [Bie99] *Let $\mathcal{C}$ be an $[\bar{n}, \bar{k}, \delta\bar{n}]$ binary linear code containing $1^{\bar{n}}$ where $\delta \geq (1-\epsilon)/2$. Let $\mathbf{G} = [\mathbf{x_1}, \ldots, \mathbf{x_{\bar{n}}}]$ be the $(\bar{k} \times \bar{n})$ generator matrix for $\mathcal{C}$. Then the set $\mathcal{S} = \{\mathbf{x}_i | 1 \leq i \leq \bar{n}\}$ is an ϵ-biased space in $\{0,1\}^{\bar{k}}$.*

Proof. In order to show that $\mathcal{S}$ is an ϵ-biased space we must show that for any $\mathbf{u} \in \{0,1\}^{\bar{k}}$,

$$\begin{aligned}
\left|\Pr_{\mathbf{x}\in\mathcal{S}}[\mathbf{x}\cdot\mathbf{u}=1] - \Pr_{\mathbf{x}\in\mathcal{S}}[\mathbf{x}\cdot\mathbf{u}=0]\right| &\leq \epsilon && (6.1)\\
\text{or that } -\epsilon \leq \Pr_{\mathbf{x}\in\mathcal{S}}[\mathbf{x}\cdot\mathbf{u}=1] - \Pr_{\mathbf{x}\in\mathcal{S}}[\mathbf{x}\cdot\mathbf{u}=0] &\leq \epsilon && (6.2)\\
\text{But, } \Pr_{\mathbf{x}\in\mathcal{S}}[\mathbf{x}\cdot\mathbf{u}=1] + \Pr_{\mathbf{x}\in\mathcal{S}}[\mathbf{x}\cdot\mathbf{u}=0] &= 1 && (6.3)\\
\text{So we need to show that } (1-\epsilon)/2 \leq \Pr_{\mathbf{x}\in\mathcal{S}}[\mathbf{x}\cdot\mathbf{u}=1] \leq (1+\epsilon)/2 & && (6.4)\\
\text{or equivalently that } (1-\epsilon)\bar{n}/2 \leq |\{\mathbf{x}\cdot\mathbf{u}=1|\mathbf{x}\in\mathcal{S}\}| \leq (1+\epsilon)\bar{n}/2 & && (6.5)
\end{aligned}$$

Since $\mathbf{G}$ is the generator matrix for $\mathcal{C}$, for any $\mathbf{u} \in \{0,1\}^{\bar{k}}$ $\mathbf{u}\cdot\mathbf{G}$ is a codeword in $\mathcal{C}$. Therefore $\mathbf{u}\cdot\mathbf{G}$ has weight at least $(1-\delta)\bar{n} \geq (1-\epsilon)\bar{n}/2$. On the other hand, since $1^{\bar{n}} \in \mathcal{C}$ any other codeword c in $\mathcal{C}$ can have distance at most $(1+\epsilon)\bar{n}/2$, otherwise the codeword $1^{\bar{n}} + c$ has distance less than $(1-\epsilon)\bar{n}/2$. But by construction

$$w(\mathbf{u} \cdot \mathbf{G}) = |\{\mathbf{u} \cdot \mathbf{x} | \mathbf{x} \in \mathcal{S}\}|$$

and hence

$$(1-\epsilon)\bar{n}/2 \leq w(\mathbf{u} \cdot \mathbf{G}) \leq (1+\epsilon)\bar{n}/2$$

This completes the proof. □

Therefore given a good binary linear code such as the RS-Had code obtained by concatenating a q-ary Reed-Solomon code with a Hadamard code, we can obtain an ϵ-biased space of size $O(m^2/\epsilon^2)$ where the distance of the code is $(1-\epsilon)/2$.

6.3 Pseudorandom sets for low-degree distinguishing tests

A natural variant of degree-d prediction tests discussed in Section §5.2 of Chapter 5 is the class of degree-d distinguishing tests.

Definition 53 *A* **degree-d distinguishing test** *p with success probability ρ for a random variable $X = (X_1, \ldots, X_m)$ defined on $\mathbb{F}_q^m$ and sampled according to some distribution D is a polynomial $p : \mathbb{F}_q^m \to \mathbb{F}_q$ of total degree at most d satisfying*

$$\left| \Pr[p(X_1, \ldots, X_m) = 0] - \Pr_{z \in \mathbb{F}_q^m}[p(z_1, \ldots, z_m) = 0] \right| \geq \rho \tag{6.6}$$

where $z = (z_1, \ldots, z_m)$ is chosen uniformly at random from $\mathbb{F}_q^m$.

6.3.1 Polynomial identity testing

Consider the problem of determining if a given polynomial $p : \mathbb{F}_q^m \to \mathbb{F}_q$ in m variables of total degree d is identically zero on $\mathbb{F}_q$. We refer to this problem as the polynomial identity test. By exhaustive search requiring time $O(q^m)$, we can evaluate the polynomial on its entire domain and determine if $p \equiv 0$. Alternatively, we can use the Schwartz-Zippel lemma which says that a polynomial in m variables and of total degree d is zero on at most a $d/|S|$ fraction of inputs from S^m where $S \subseteq \mathbb{F}_q$. This suggests a probabilistic polynomial-time algorithm for the polynomial identity test problem in the following manner: choose m elements at random from S and evaluate p on the m-tuple in S^m drawn from them. Since the probability that p vanishes on a random point is at most $d/|S|$, the algorithm fails with at most $d/|S|$. Is it possible however to use fewer than $m \log q$ random bits than as was needed above? Of course, if the class of decision problems bounded for which there exists a probabilistic polynomial-time denoted BPP collapses to P as is presently widely believed the answer to this question would be yes, but we do not know for sure if BPP=P. This pertains to the important question of derandomizing polynomial identity testing about which there has been some extensive research in the past [CK00, KS01, LV98, IKW01, KI03].

6.3.2 Pseudorandom sets for low-degree tests and polynomial identity testing

The previous discussion sheds some light on the existence of a connection between polynomial identity testing and low-degree distinguishing tests. Suppose that there exists a q-ary ρ-pseudorandom set $\mathcal{S}$ that could fool the class of all degree-d distinguishing tests. We construct a random variable X with distribution D over $\mathcal{S}$ by choosing an element uniformly at random from $\mathcal{S}$. Then by Definition 53, for any degree-d distinguishing test p, X satisfies

$$\left| \Pr[p(X_1, \ldots, X_m) = 0] - \Pr_{z \in \mathbb{F}_q^m}[p(z_1, \ldots, z_m) = 0] \right| < \rho \tag{6.7}$$

If furthermore it is the case that D requires fewer random bits to describe it than was needed for the probabilistic algorithm described above then we have answered the question posed in §6.3.1. We have already seen constructions of q-ary ρ-pseudorandom sets for the class of degree-d prediction tests in Theorem 51 and Corollary 52. Is it possible similarly to construct pseudorandom sets for degree-d distinguishing tests? In what follows we will attempt an answer to this question.

6.3.3 Towards constructing pseudorandom sets for low-degree distinguishing sets

Before taking the subject any further it would serve us well to revisit our pseudorandom set constructions for low-degree prediction tests. Let $\mathbb{F}_q$ be a field of order q. The construction in Corollary 52 in Chapter 5 gives a (dh/q)-pseudorandom set $\mathcal{S}$ for the class of degree-d prediction tests using a Reed-Solomon code $\mathcal{C}$ with parameters $h = m + 2, q$. $\mathcal{S}$ was defined as

$$\mathcal{S} = \{f_{\mathcal{C},m}(x, y) : 1 \leq y \leq q\} \tag{6.8}$$

where x was such that $\mathcal{C}(x)[1 \ldots m] = 0^{m-1}1$. Let α be a generator for $\mathbb{F}_q^*$, the multiplicative group of $\mathbb{F}_q$. Since $\mathcal{C}$ is a Reed-Solomon code, we can associate a univariate polynomial $g : \mathbb{F}_q \rightarrow \mathbb{F}_q$ of degree h with x such that

$$g(z) = \frac{(z-1)(z-\alpha)(z-\alpha^2)\ldots(z-\alpha^h)}{(\alpha^{h+1}-1)(\alpha^{h+1}-\alpha)\ldots(\alpha^{h+1}-\alpha^h)}$$

Our attempt will explore using $\mathcal{S}$ to obtain a pseudorandom set $\mathcal{S}'$ for degree-d distinguishing tests. Consider the following algorithm. Its input is a polynomial p in m variables of total degree d defined on $\mathbb{F}_q$. Presented below is an algorithm that takes as input an m-variate polynomial of degree d and accepts it if it identically zero and rejects otherwise.

Algorithm 1 Determine if $p \equiv 0$

```
Require: An $m$-variate polynomial $p : \mathbb{F}_q^m \to \mathbb{F}_q$ of degree $d$
    Fix $l \geq 2(m+2)d^3 d_{\mathcal{R}}$
    for $i = 0$ to $l-1$ do
        Define $r_i \in \mathbb{F}_q^m$ as $r_i = (g(\alpha^i), g(\alpha^{i+1}), \ldots, g(\alpha^{i+m-1}))$
        if $p(r_i) \neq 0$ then
            REJECT and EXIT
        end if
    end for
    ACCEPT
```

We will define $d_{\mathcal{R}}$ subsequently. In order to show that Algorithm 1 works correctly, we would only need to show that if $p \not\equiv 0$ then there exists an i such that $p(r_i) \neq 0$ because if $p \equiv 0$ then the algorithm accepts. Note that if this is true then Algorithm 1 unconditionally derandomizes polynomial identity testing. We are tasked with determining if this holds and we will do so by advancing some claims towards proving the correctness of the algorithm. In showing that an r_i exists for which $p(r_i) \neq 0$ if $p \not\equiv 0$, we state and prove the following proposition.

Proposition 12 *For some $j : 1 \leq j \leq m$ let $p' : \mathbb{F}_q^j \to \mathbb{F}_q$ be a degree-d polynomial in j variables over $\mathbb{F}_q$. Let $R = \{r_i | 0 \leq i \leq r-1\}$ be a set in $\mathbb{F}_q^m$ where r_i is defined for each i as in Algorithm 1. Then*

$$\Pr_{z \in R}[p'(z_1, \ldots, z_j) = z_{j+1}] \leq \frac{(m+2) \cdot d}{r}$$

Proof. Suppose for the sake of contradiction that

$$\Pr_{z \in R}[p'(z_1, \ldots, z_j) = z_{j+1}] > \frac{(m+2) \cdot d}{r} \tag{6.9}$$

This means that p' is a $(j+1)$-th element degree-d prediction test with success probability at least $d(m+2)/r$ for the random variable obtained by sampling z uniformly from R.

Consider the $(j+1)$-variate polynomial $q(z_1, \ldots, z_{j+1}) = z_{j+1} - p'(z_1, \ldots, z_j)$. q also is of degree d. The polynomial $q'(z) = q(g(z), g(\alpha z), \ldots, g(\alpha^j z))$ is a univariate polynomial and has total degree at most $d(m+2)$ where g is the degree-h polynomial defined in (6.8). Also, by construction q' vanishes in the points $z \in R$ where p' predicts z_{j+1} accurately and so from (6.9) q' vanishes on greater than $d(m+2)$ points in R. But q' is a polynomial of total degree $d(m+2)$ and by the Fundamental Theorem of Algebra a non-zero polynomial of degree h can have at most h zeroes. Hence, q' must be identically zero over $\mathbb{F}_q$ and in particular for all $z \in R$:

$$\begin{aligned} q'(z_1, \ldots, z_{j+1}) = z_{j+1} - p'(z_1, \ldots, z_j) &= 0 && (6.10) \\ \Rightarrow p'(z_1, \ldots, z_j) &= z_{j+1} && (6.11) \end{aligned}$$

(6.11) tells us that p' is an errorless $(j+1)$-th element predictor for the random variable obtained by sampling z uniformly from R. Therefore,

$$\begin{aligned} p'(g(\alpha^0), \ldots, g(\alpha^{j-1})) = g(0, \ldots, 0) &= g(\alpha^j) = 0 && (6.12) \\ p'(g(\alpha^{m+2-j}), \ldots, g(\alpha^{m+2-1})) = g(0, \ldots, 0) &= g(\alpha^j) = 1 && (6.13) \end{aligned}$$

which is a contradiction to our assumption. We should note that this proposition is almost identical to an analogous application of Lemma 10, the only difference being that in the latter case the predictor's success rate would be measured over $\mathbb{F}_q^j$ whereas in the former case this is measured over R. □

We will attempt showing the existence of an $r_i \in R$ for which $p(r_i) \neq 0$ by means of arriving at a contradiction. We hope to achieve this contradiction by showing that we can obtain from a low-degree distinguishing test against R, a low-degree prediction test with reasonably good success rate. Once this is achieved, we will obtain a contradiction by picking suitable parameters in the construction of R and appealing to Proposition 12 to ensure that R is a pseudorandom set that fools such a prediction test.

6.3.4 Constructing the low-degree prediction test

Lemma 13 *Suppose there exists a degree-d distinguishing test p such that $p(r_i) = 0$ for all $r_i \in R$. Then for some $t, 1 \leq t \leq m$ there exists an $(m - t + 1)$-th element predictor $f_{u^*} : \mathbb{F}_q^{m-t} \to \mathbb{F}_q$ with success probability $1/d^2$ for the random variable induced by picking r_i uniformly from R.*

Proof. We will inductively construct a series of non-zero polynomials $p_0, \ldots, p_m$ based on p in the following manner:

- p_0 is defined as:

  $$p_0 = p$$

  Note that p_0 can also be written as a univariate polynomial in x_m over the polynomial ring $\mathbb{F}_q[x_1, \ldots, x_{m-1}]$:

  $$p_0(x_m) = \sum_{j=0}^{d} c_{0,j}(x_1, \ldots, x_{m-1}) x_m^j$$

  where $c_{0,j} : \mathbb{F}_q^{m-1} \to \mathbb{F}_q, 0 \leq j \leq d$ are polynomials in $(m - 1)$ variables over $\mathbb{F}_q$ and with total degree at most $d - j$.

- Let $p_k : \mathbb{F}_q[x_1, \ldots, x_{m-k-1}] \to \mathbb{F}_q$ be the non-zero univariate polynomial in x_{m-k} of total degree at most d constructed in this manner. p_k may be written as:

  $$p_k(x_{m-k}) = \sum_{j=0}^{d} c_{k,j}(x_1, \ldots, x_{m-k-1}) x_{m-k}^j$$

  Since p_k is identically not zero, there exists some $j^*; 0 \leq j^* \leq d$ such that $c_{k,j^*}(x_1, \ldots, x_{m-k-1}) \not\equiv 0$ while $c_{k,j}(x_1, \ldots, x_{m-k-1}) \equiv 0$ for all $j > j^*$. Then p_{k+1} is defined as

  $$p_{k+1} = c_{k,j^*}$$

Observation 6.4 *The multiset of polynomials $\{p_0, \ldots, p_m\}$ constructed in this fashion has at most d distinct elements.*

Proof. First of all, we can easily observe that for any i each successive polynomial in the series contains one literal less than its predecessor. Therefore if $\deg(p_i)$ denotes the degree of p_i, then $\deg(p_i) \leq \deg(p_{i-1})$. Furthermore, suppose that $\deg(p_i) = \deg(p_{i-1}) = d'$ say. From the procedure we used to construct p_i, there is a j^* such that $p_{i+1} = c_{i,j^*}$. But p_i has degree j^* in x_{m-i} and therefore $\deg(p_{i+1}) = \deg(c_{i,j^*}) = d' - j^* = \deg(p_i) = d'$ giving $j^* = 0$. Since our procedure guarantees that $c_{i,j} \equiv 0$ for all $j > j^*$, $p_i = c_{i,j^*} = p_{i+1}$. This completes the proof since p has degree d. □

Our next two observations are self-explanatory.

Observation 6.5 *Let s denote the smallest integer for which deg$(p_s) = 0$. Then from our original hypothesis about p we get*

$$\Pr_{z \in R}[p_0(z) \neq 0] = 0$$

while from how the series of polynomials are constructed we get

$$\Pr_{z \in R}[p_s(z) \neq 0] = 1$$

Observation 6.6 *Since there are at most d distinct polynomials in $\{p_0, \ldots, p_s\}$, there exists some t for which*

$$\Pr_{z \in R}[p_t(z) \neq 0] - \Pr_{z \in R}[p_{t-1}(z) \neq 0] \geq 1/d \tag{6.14}$$

From Observation 6.6, we conclude that since $|R| = \Omega(md^4)$ there is an $r_i \in R$ such that

$$p_t(r_i) \neq 0 \wedge p_{t-1}(r_i) = 0$$

We will discuss in what follows a randomized algorithm that will determine a point r_i where $p_{t-1}(r_i) = 0$. Note that as a univariate polynomial of degree at most d in x_{m-t+1} over $\mathbb{F}_q[x_1, \ldots, x_{m-t}]$, p_{t-1} has at most d roots over the ring. So, a reasonable goal for a randomized algorithm would be to output one of these roots in $\mathbb{F}_q$ (if it exists) at random. Recall that

$$p_{t-1} = \sum_{k=0}^{d} c_{t-1,k}(x_1, \dots, x_{m-t}) x_{m-t+1}^k$$

We will now use the previous discussion to describe the final step involved in constructing our predictor. We define a family of functions $\mathcal{F} = \{f_u : \mathbb{F}_q^{m-t} \rightarrow \mathbb{F}_q | 1 \leq u \leq d\}$ with the property that on input $x_1, \dots, x_{m-t}$, an element of $\mathcal{F}$ given by f_u outputs the u-th root z^* of the univariate polynomial $p'_{t-1}(z) = \sum_{k=0}^{d} c_{k,t-1} z^k$ under some canonical ordering of the d roots of $p'_{t-1}(z)$. If there is no root then f_u outputs 0. Note that given $(x_1, \dots, x_{m-t})$ the coefficients $c_{k,t-1} = c_{k,t-1}(x_1, \dots, x_{m-t})$ are all in $\mathbb{F}_q$.

In order to simplify our notation, we shall denote $r_i|j$ to mean the first j components of $r_i = (r_i)_{1,\dots,j}$. We also denote the event $(p(r_i|m-t) \neq 0) \wedge (p(r_i|m-t+1) = 0)$ by E. We observe using simple facts about conditional probability that

$$\begin{aligned} \Pr_{r_i \in R, u}[f_u(r_i|m-t) = (r_i)_{m-t+1}] &\geq \Pr_u[f_u \text{ chooses } (r_i)_{m-t+1}|E] \cdot \Pr_{r_i \in R}[E] \\ &\geq \frac{1}{d} \cdot \frac{1}{d} \\ &= \frac{1}{d^2} \end{aligned} \tag{6.15}$$

which follows from the fact that $\Pr_{r_i \in R}[E] \geq 1/d$ from (6.14) and the randomized algorithm would choose $(r_i)_{m-t+1}$ with probability at least $1/d$. From (6.15) we get:

$$\Pr_{r_i \in R, u}[f_u \text{ chooses } (r_i)_{m-t+1}] \geq \frac{1}{d^2} \tag{6.16}$$

From an averaging argument, we can show that there exists a fixing of $u = u^*$ for which

$$\Pr_{r_i \in R}[f_{u^*}((r_i)_{1,\dots,m-t}) = (r_i)_{m-t+1}] \geq \frac{1}{d^2} \tag{6.17}$$

f_{u^*} is our predictor with success probability at least $1/d^2$ over the choice of $r_i \in R$.

□

But we set out to construct a low-degree prediction test from a degree-d distinguishing test and we ask if f_{u^*} can be described by a low-degree "root-finding" polynomial $\mathcal{R}$. All we need of $\mathcal{R}$ is that upon input $(c_0, \ldots, c_d) \in \mathbb{F}_q^{d+1}$, $\mathcal{R}$ outputs the u^*-th root in $\mathbb{F}_q$ of the polynomial $p'(z) = \sum_{k=0}^{d+1} c_k z^k$. If no such root exists in $\mathbb{F}_q$, then $\mathcal{R}$ outputs an arbitrary element κ. Namely, if p' has roots $\{\zeta_1, \ldots, \zeta_d\}$ not all in $\mathbb{F}_q$ then

$$\mathcal{R}(c_0, \ldots, c_d) = \begin{cases} \zeta_{u^*} & \text{if } \zeta_{u^*} \in \mathbb{F}_q, \\ \kappa & \text{otherwise.} \end{cases} \tag{6.18}$$

We stress here that κ is arbitrary and not even necessarily fixed. We only require that it does not lie in $\mathbb{F}_q$. We enunciate our reasons for requiring $\mathcal{R}$ to be low-degree will be clear in the following claim.

Claim 54 *Suppose $\mathcal{R} : \mathbb{F}_q^{d+1} \to \mathbb{F}_q$ is a polynomial of degree $d_{\mathcal{R}}$ such that upon inputs $(c_0, \ldots, c_d) \in \mathbb{F}_q^{d+1}$ $\mathcal{R}$ outputs the u^*-th root of the univariate polynomial $p'(z) = \sum_{k=0}^{d} c_k z^k$. Then there exists an $(m-t+1)$-th element predictor with degree $dd_{\mathcal{R}}$ for the random variable induced by sampling uniformly from R with success probability $1/d^2$.*

Proof. We define $P : \mathbb{F}_q^{m-t} \to \mathbb{F}_q$ as follows:

$$P(x_1, \ldots, x_{m-t}) = \mathcal{R}(c_0(x_1, \ldots, x_{m-t}), c_1(x_1, \ldots, x_{m-t}), \ldots, c_d(x_1, \ldots, x_{m-t}))$$

P is a polynomial in $(x_1, \ldots, x_{m-t})$ of total degree at most $dd_{\mathcal{R}}$ since each of $c_0, \ldots, c_d$ is a polynomial of degree at most d and $\mathcal{R}$ is a polynomial of degree $d_{\mathcal{R}}$ from the supposition of the claim. Since $\mathcal{R}$ outputs the u^*-th root of the polynomial $p'(z) = \sum_{k=0}^{d} c_k z^k$, P agrees with $f_{u^*}(x_1, \ldots, x_{m-t})$ on all $r_i : 0 \leq i \leq l-1$ in R. Furthermore P outputs the u^*-th root when $c_1, \ldots, c_d$ are not all 0 (it outputs 0 otherwise). Therefore from (6.16) we get

$$\Pr_{z \in R}[P(z_{1 \dots m-t}) = z_{m-t+1}] \geq \frac{1}{d^2}$$

which is as claimed. □

Let us tease out the rest of the hoped-for contradiction. By construction in Algorithm 1, $l > (m+2)d^3 d_{\mathcal{R}}$ and hence $1/d^2 > (m+2)dd_{\mathcal{R}}/l$. Therefore, P is a low-degree prediction test with success probability greater than $(m+2)\mathrm{deg}(P)/l$ which is in contravention to Proposition 12 giving us the required contradiction to our hypothesis.

The $64000-dollar question is if the degree of a "root-finding" polynomial $\mathcal{R}$ given by $d_{\mathcal{R}}$ is small. If $d_{\mathcal{R}} = \mathrm{poly}(d)$ then $l = \Omega(m \cdot \mathrm{poly}(d))$ and we will have successfully derandomized polynomial identity testing using a pseudorandom set of size polynomial in m, d.

Unfortunately for us, Lemma 14 below shows that $\mathcal{R}$ has degree $\Omega(q-d)$.

Lemma 14 *Let $\mathcal{R} : \mathbb{F}_q^{d+1} \rightarrow \mathbb{F}_q$ be defined as in (6.18). Then deg$(\mathcal{R}) \geq (q-d-1)$.*

Proof. By definition, $\mathcal{R}$ outputs the u^*-th root ζ_{u^*} of the polynomial $p'(z) = \sum_{k=0}^{d} c_k z^k$ when $\zeta_{u^*} \in \mathbb{F}_q$. p' is a univariate polynomial of total degree d. Consider the polynomial $g : \mathbb{F}_q^{d+1} \rightarrow \mathbb{F}_q$ given by

$$g(c_0, \dots, c_d) = p'(\mathcal{R}(c_0, \dots, c_d)) \qquad (6.19)$$

Claim 55 *g is a $(d+1)$-variate polynomial that has degree at least $q-1$.*

Proof. (Of claim) Consider the univariate polynomial $g'(c_0, c_1)$ given by

$$g'(c_1) = g(1, c_1, 0, \dots, 0) = p'(\mathcal{R}(1, c_1, 0, \dots, 0))$$

Since $p'(z) = 1 + c_1 z$ has degree 1, $\mathcal{R}$ must behave as follows:

$$\mathcal{R}(1, c_1, 0, \ldots, 0) = \begin{cases} -1/c_1 & c_1 \neq 0, \\ \kappa & \text{otherwise.} \end{cases}$$

Consequently,

$$g'(c_1) = \begin{cases} 0 & c_1 \neq 0, \\ \kappa & \text{otherwise.} \end{cases}$$

Therefore, for at least $(q-1)$ values of $c_1 \in \mathbb{F}_q$, $g'(c_1) = 0$ and hence $g'(c_1)$ has degree at least $(q-1)$. But g' is a restriction to one variable of the polynomial $g(c_0, c_1, \ldots, c_d)$ and hence $g(c_0, \ldots, c_d)$ must also have total degree at least $(q-1)$. □

Since p' has degree at most d and g has degree at least $q - 1$ from Claim 55, $\mathcal{R}$ must have degree at least $q - d - 1$. □

Epilogue

In this work, we looked at some intriguing connections between codes and pseudorandom objects. Specifically, we considered the family of cyclic linear codes and described a very simple and generic technique of obtaining limited pseudorandomness from them. The pseudorandomness was limited because the objects obtained fooled a limited class of low-degree polynomial tests rather than the larger class of *all* efficient tests but we are hopeful that the techniques that we addressed in this work can be used to improve on this.

Still, an interesting concomitant to pseudorandom objects for low-degree polynomial tests as we saw was how they essentially would imply derandomizing polynomial identity testing. We looked at approaching this problem via our constructions but unfortunately came up short in our attempt to convert pseudorandom objects for low-degree prediction tests to pseudorandom objects for low-degree distinguishing tests. Nonetheless we are confident that our attempt deserves a closer look and we believe that it can be used to get some partial success.

Bibliography

[ABI86] Noga Alon, László Babai, and Alon Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *Journal of Algorithms*, 7(4):567–583, 1986.

[AGHP92] N. Alon, O. Goldreich, J. Hastad, and R. Peralta. Simple constructions of almost k-wise independent random variables. *Random Structures and Algorithms*, (3):289–304, 1992.

[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, 1998.

[BF90] D. Beaver and J. Feigenbaum. Hiding instances in multioracle queries. In *STACS 90: Proceedings of the seventh annual symposium on Theoretical aspects of computer science*, pages 37–48, New York, NY, USA, 1990. Springer-Verlag New York, Inc.

[BFNW93] L. Babai, L. Fortnow, N. Nisan, and A. Wigderson. BPP has subexponential time simulations unless EXPTIME has publishable proofs. *Computational Complexity*, 3(4):307–318, 1993.

[Bie99] Jürgen Bierbrauer. Weakly biased arrays, weakly dependent arrays and error-correcting codes, 1999.

[BIW04] Boaz Barak, Russell Impagliazzo, and Avi Wigderson. Extracting randomness from few independent sources. In *Proceedings of the 45th Annual*

IEEE Symposium on Foundations of Computer Science, pages 384–393, 2004.

[BKS$^+$04] B. Barak, G. Kindler, R. Shaltiel, B. Sudakov, and A. Wigderson. Simulating independence: New constructions of condensers, ramsey graphs, dispersers and extractors, 2004. Submitted for publication.

[BLR90] M. Blum, M. Luby, and R. Rubinfeld. Self-testing/correcting with applications to numerical problems. In *STOC '90: Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 73–83, New York, NY, USA, 1990. ACM Press.

[BM84] M. Blum and S. Micali. How to generate cryptographically strong sequence of pseudo-random bits. *SIAM Journal on Computing*, 13:850–864, 1984.

[Cal04] Cristian Calude. Algorithmic randomness, quantum physics, and incompleteness. In Maurice Margenstern, editor, *MCU*, volume 3354 of *Lecture Notes in Computer Science*, pages 1–17. Springer, 2004.

[CK00] Z.-Z. Chen and M.-Y. Kao. Reducing randomness via irrational numbers. *SIAM Journal on Computing*, 29(4):1247–1256, 2000.

[GL89] O. Goldreich and L. A. Levin. A hard-core predicate for all one-way functions. In ACM, editor, *Proceedings of the twenty-first annual ACM Symposium on Theory of Computing, Seattle, Washington, May 15–17, 1989*, pages 25–32, New York, NY, USA, 1989. ACM Press.

[GS00] V. Guruswami and M. Sudan. List decoding algorithms for certain concatenated codes. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing (STOC-00)*, pages 181–190, 2000.

[Gur04] V. Guruswami. Better extractors for better codes? In *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 436–444, New York, NY, USA, 2004. ACM Press.

[IKW01] R. Impagliazzo, V. Kabanets, and A. Wigderson. In search of an easy witness: Exponential time vs. probabilistic polynomial time. In Frances M. Titsworth, editor, *Proceedings of the Sixteenth Annual Conference on Computational Complexity (CCC-01)*, pages 2–12, Los Alamitos, CA, June 18–21 2001. IEEE Computer Society.

[ILL89] R. Impagliazzo, L. Levin, and M. Luby. Pseudorandom generation from one-way functions. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 12–24, May 1989.

[INW94] R. Impagliazzo, N. Nisan, and A. Wigderson. Pseudorandomness for network algorithms. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on the Theory of Computing*, pages 356–364, Montréal, Québec, Canada, 23–25 May 1994.

[IW97] R. Impagliazzo and A. Wigderson. P = BPP if E requires exponential circuits: Derandomizing the XOR lemma. In *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing (STOC '97)*, pages 220–229, New York, May 1997. Association for Computing Machinery.

[KI03] V. Kabanets and R. Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC-03)*, pages 627–634, San Diego, May 9–11 2003. ACM Press.

[KS01] A. Klivans and D. Spielman. Randomness efficient identity testing of multivariate polynomials. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing (STOC-01)*, pages 216–223, New York, NY, USA, 2001. ACM Press.

[LV98] D. Lewin and S. Vadhan. Checking polynomial identities over any field: towards a derandomization? In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing (STOC)*, pages 438–447, 1998.

[NN93] Joseph Naor and Moni Naor. Small-bias probability spaces: Efficient constructions and applications. *SIAM Journal on Computing*, 22(4):838–856, August 1993.

[NW94] N. Nisan and A. Wigderson. Hardness vs. randomness. *Journal of Computer and System Sciences*, 49(2):149–167, October 1994.

[NZ96] N. Nisan and D. Zuckerman. Randomness is linear in space. *Journal of Computer and System Sciences*, 52(1):43–52, February 1996.

[Raz05] R. Raz. Extractors with weak random seeds. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC-05)*, May 2005.

[RR99] R. Raz and O. Reingold. On recycling the randomness of states in space bounded computation. In *Proceedings of the 31st ACM Symposium on Theory of Computing*, pages 159–168, 1999.

[RRV02] R. Raz, O. Reingold, and S. Vadhan. Extracting all the randomness and reducing the error in Trevisan's extractors. *Journal of Computer and System Sciences*, 65(1):97–128, 2002.

[RTS00] J. Radhakrishnan and A. Ta-Shma. Bounds for dispersers, extractors, and depth-two superconcentrators. *SIAM Journal on Discrete Mathematics*, 13(1):2–24, 2000.

[Sch80] J. T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27(4):701–717, 1980.

[Sha02] R. Shaltiel. Recent developments in explicit constructions of extractors. *Bulletin of the European Association for Theoretical Computer Science*, 77:67–, June 2002. Columns: Computational Complexity.

[STV01] M. Sudan, L. Trevisan, and S. Vadhan. Pseudorandom generators without the XOR lemma. *Journal of Computer and System Sciences*, 62(2):236–266, March 2001.

[SU01] R. Shaltiel and C. Umans. Simple extractors for all min-entropies and a new pseudo-random generator. In IEEE, editor, *42nd IEEE Symposium on Foundations of Computer Science: proceedings: October 14–17, 2001, Las Vegas, Nevada, USA*, pages 648–657, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, 2001. IEEE Computer Society Press.

[Sud97] Madhu Sudan. Decoding of Reed-Solomon codes beyond the error-correction bound. *Journal of Complexity*, 13(1):180–193, 1997.

[SZ99] A. Srinivasan and D. Zuckerman. Computing with very weak random sources. *SIAM Journal on Computing*, 28(4):1433–1459, August 1999.

[Tre01] L. Trevisan. Extractors and pseudorandom generators. *Journal of the ACM*, 48(4):860–879, July 2001.

[Tre03] L. Trevisan. List decoding using the XOR lemma. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, pages 126–135, 2003.

[TS96] A. Ta-Shma. On extracting randomness from weak random sources (extended abstract). In ACM, editor, *Proceedings of the twenty-eighth annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, May 22–24, 1996*, pages 276–285, New York, NY, USA, 1996. ACM Press.

[TSZS01] A. Ta-Shma, D. Zuckerman, and S. Safra. Extractors from Reed-Muller codes. In IEEE, editor, *42nd IEEE Symposium on Foundations of Computer Science: proceedings: October 14–17, 2001, Las Vegas, Nevada, USA*, pages 638–647, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, 2001. IEEE Computer Society Press.

[Uma02] C. Umans. Pseudo-random generators for all hardnesses. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC-02)*, pages 627–634, New York, May 19–21 2002. ACM Press.

[WZ93] A. Wigderson and D. Zuckerman. Expanders that beat the eigenvalue bound: Explicit construction and applications. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on the Theory of Computing*, pages 245–251, San Diego, California, 16–18 May 1993.

[Yao82] A. C. Yao. Theory and applications of trapdoor functions. In *Proceedings of the 23rd Symposium on Foundations of Computer Science (FOCS)*, pages 80–91. IEEE Computer Society Press, 1982.

[Zip79] R. E. Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of EUROSAM 79*, volume 72 of *Lecture Notes in Computer Science*, pages 216–226, 1979.

[Zuc97] D. Zuckerman. Randomness-optimal oblivious sampling. *Random Structures and Algorithms*, 11:345–367, 1997.