Minimizing the Complexity of Goldreich's Pseudorandom Generator

Alex Lombardi*
MIT

Vinod Vaikuntanathan†
MIT

March 25, 2017

Abstract

In the study of cryptography in NC^0, it was previously known that Goldreich's candidate pseudorandom generator (PRG) is insecure when instantiated with a predicate P in 4 or fewer variables, if one wants to achieve polynomial stretch (that is, stretching n bits to $n^{1+\epsilon}$ bits for some constant $\epsilon > 0$). The current standard candidate predicate for this setting is the "tri-sum-and" predicate $\mathrm{TSA}(x) = \mathrm{XOR}_3 \oplus \mathrm{AND}_2(x) = x_1 \oplus x_2 \oplus x_3 \oplus x_4 x_5$, yielding a candidate PRG of locality 5. Moreover, Goldreich's PRG, when instantiated with TSA as the predicate, is known to be secure against several families of attacks, including $\mathbb{F}_2$-linear attacks and attacks using SDP hierarchies such as the Lasserre/Parrilo sum-of-squares hierarchy.

However, it was previously unknown if TSA is an "optimal" predicate according to other complexity measures: in particular, *decision tree (DT-)complexity* (i.e., the smallest depth of a binary decision tree computing P) and $\mathbb{Q}$*-degree* (i.e., the degree of P as a polynomial over $\mathbb{Q}$), which are important measures of complexity in cryptographic applications such as the construction of an indistinguishability obfuscation scheme. In this work, we ask: *Can Goldreich's PRG be instantiated with a predicate with DT-complexity or* $\mathbb{Q}$*-degree less than* 5?

We show that this is indeed possible: we give a candidate predicate for Goldreich's PRG with DT-complexity 4 and $\mathbb{Q}$-degree 3; in particular, this candidate PRG therefore has the property that every output bit is a degree 3 polynomial in its input. Moreover, Goldreich's PRG instantiated with our predicate has security properties similar to what is known for TSA, namely security against $\mathbb{F}_2$-linear attacks and security against attacks from SDP hierarchies such as the Lasserre/Parrilo sum-of-squares hierarchy.

We also show that all predicates with either DT-complexity less than 4 or $\mathbb{Q}$-degree less than 3 yield insecure PRGs, so our candidate predicate simultaneously achieves *the best possible locality, DT-complexity,* $\mathbb{Q}$*-degree, and* $\mathbb{F}_2$*-degree* according to all known attacks.

*E-mail: alexjl@mit.edu. Supported by an Akamai Presidential Fellowship.

†E-mail: vinodv@mit.edu. Research supported in part by NSF Grants CNS-1350619 and CNS-1414119, Alfred P. Sloan Research Fellowship, Microsoft Faculty Fellowship, the NEC Corporation, a Steven and Renee Finn Career Development Chair from MIT. This work was also sponsored in part by the Defense Advanced Research Projects Agency (DARPA) and the U.S. Army Research Office under contracts W911NF-15-C-0226.

Contents

1 Introduction

While hard problems occur abundantly in nature, *useful hard problems* are somewhat rare. In particular, to be useful in cryptography, the (conjuctured) hard problems need several additional properties: at the minimum, average-case hardness and the ability to sample hard instances with their solutions (a property that is required for building one-way functions).

It is hard enough to come up with cryptographically useful hard problems, but to make our life even harder, we also often want the cryptographic constructions to be as *simple* as possible. For example, take the case of (cryptographic) pseudorandom generators (PRGs), the object of study in this work. Here, we ask for a function[1] $G : \{0,1\}^n \to \{0,1\}^m$ which is: (a) expanding, meaning that $m > n$ and ideally, m is a large polynomial in n; (b) pseudorandom, meaning that $G(U_n)$ is computationally indistinguishable from U_m, where U_n and U_m are uniform distributions on n and m bits, respectively; and (c) simple, meaning that G is computable by a (uniform) $\mathbf{NC^0}$ circuit.

In a remarkable tour-de-force, Applebaum, Ishai and Kushilevitz [AIK06] showed how to "compile" any PRG computable in a large complexity class, say $\mathbf{NC^1}$, into one that can be computed in $\mathbf{NC^0}$. Their PRGs even had locality as small as 4, meaning that each output bit depends on only 4 input bits. This gave us candidate PRGs under essentially any standard complexity assumption, e.g., the hardness of factoring, that of the decisional Diffie-Hellman problem, the worst-case hardness of approximating shortest vectors in lattices, and so forth. The main deficiency of the AIK work is that even if you start with a PRG with large, polynomial, stretch in $\mathbf{NC}^1$, you only get a PRG with sub-linear (additive) stretch in $\mathbf{NC^0}$, that is $m = n + o(n)$. Indeed, as Mossel, Shpilka and Trevisan [MST06] showed, there are no PRGs in $\mathbf{NC^0}$ with polynomial stretch and locality 4, so in a sense, the [AIK06] construction is nearly optimal.

However, we cannot help but ask for more. Polynomial stretch PRGs, also called PPRGs (where $m = n^c$ for some $c > 1$) in $\mathbf{NC^0}$ have several applications including secure two-party computation with constant overhead [IKOS08] and more recently, indistinguishability obfuscation (IO) from constant-degree multilinear maps [Lin16a, LV16, AS16, Lin16b]. Jumping ahead, we note that the IO application cares about reducing parameters of the PRG other than its locality, in particular its $\mathbb{Q}$-degree, defined as the degree of each output bit when expressed as a polynomial over the rationals. But more on this later.

PPRGs are much trickier to construct; indeed, our best hope is a candidate construction (actually, a family of constructions) first proposed by Goldreich [Gol00] in 2000. Goldreich's generator and its properties in the polynomial stretch regime are the central themes of this paper.

Goldreich's Pseudo-random Generator. Goldreich's candidate pseudorandom generator, first introduced in [Gol00] (then as a candidate one-way function), can be instantiated with any k-ary predicate $P : \{0,1\}^k \to \{0,1\}$ and any k-uniform (directed) hypergraph H on n vertices and m hyperedges. Given H and P, we define a PRG $G : \{0,1\}^n \to \{0,1\}^m$ as follows: Identify each vertex in H with an index in $[n]$ and each hyperedge with an index $i \in [m]$. For each $i \in [m]$, let $\Gamma_H(i) \in [n]^k$ be the sequence of k vertices in the ith hyperedge. Then, Goldreich's PRG is the function from $\{0,1\}^n$ to $\{0,1\}^m$ defined by

$$G_{H,P}(x) = \left(P(x|_{\Gamma_H(i)})\right)_{i\in[m]}.$$

[1] To be more precise, we ask for a *family* of functions $\{G_n\}_{n\in\mathbb{N}}$ where G_n maps n bits to $m = m(n) > n$ bits.

That is, the ith bit of $G_{H,P}(x)$ is the output of P when given the $\Gamma_H(i)$-restriction of x as input. For the rest of this paper, we think of k as an absolute constant.

Many predicates $P : \{0,1\}^k \to \{0,1\}$ are known to yield *insecure* PRGs when plugged into Goldreich's generator $G_{H,P}$. In particular, call a predicate *degenerate* if it is either affine or *not* pairwise-independent, that is it is correlated to some pair of its input bits.[2] A long sequence of results [MST06, BQ12, ABR12, OW14, KMOW17] show the following beautiful *dichotomy theorem*: every predicate is either (a) degenerate, in which case it can be broken by a $\mathbb{F}_2$-linear attack, or (b) non-degenerate, in which case it resists a large class of attacks including $\mathbb{F}_2$-linear attacks [ABR12] and attacks that can be implemented in the Lasserre/Parrilo sum-of-squares (SOS) hierarchy, a powerful SDP hierarchy which generalizes the Sherali-Adams (SA_+) and Lovász-Schrijver (LS_+) proof systems [OW14, KMOW17]. The latter class of attacks is of particular interest in the context of Goldreich's generator as the problem of breaking it is closely tied to the problem of refuting random instances of CSPs (see [OW14, KMOW17, BS16] for context on SOS and refutation of CSPs). Moreover, the security result against SOS attacks from [KMOW17] holds even when the stretch of the PRG is a large polynomial: the stretch achievable by a predicate P is entirely characterized by its t-wise independence properties.

As a special case of the dichotomy results, any predicate P with locality at most 4 (i.e., $k \leq 4$) is degenerate and therefore Goldreich's PRG using P is broken. In addition, given these results, when looking for candidate predicates that could achieve polynomial stretch, one can take the view that any non-degenerate predicate is a viable candidate. For a more detailed discussion of Goldreich's PRG, we refer the reader to Section 3.

Locality, $\mathbb{Q}$-degree and Program Obfuscation. As we already alluded, some applications of polynomial-stretch "local" PRGs do not necessarily need small locality, but rather they care about optimizing other parameters of the predicate P. In particular, a sequence of works showed how to construct program obfuscation schemes from *constant-degree* cryptographic multi-linear maps [Lin16a, LV16] assuming that P can be written as a *constant-degree* polynomial $P_{\mathbb{Q}}$ over the rationals (which agrees with P on the Boolean hypercube).

The connection between multi-linearity and $\mathbb{Q}$-degree is precise and quantitative: [Lin16a, LV16] showed that if Goldreich's PRG is secure with a predicate P with $\mathbb{Q}$-degree d, and cryptographic multilinear maps with degree $D = 3d + 2$ (and a secure Diffie-Hellman-like problem) exist, so does program obfuscation. Indeed, this connection is even sharper, but we will defer a discussion of the sharper statement to the end of the introduction. Minimizing the "degree of the cryptographic multilinear maps" is a vital research direction; indeed, achieving multi-linearity 2 will imply a construction of program obfuscation, and indeed nearly all of cryptography, from hard Diffie-Hellman-like problems on elliptic curves.

This inspires the question:

What is the smallest $\mathbb{Q}$-degree for a predicate P that makes Goldreich's function a PPRG?

Since $\mathbb{Q}$ degree of a predicate can be no more than its locality, the answer is at most 5, but how small can it be?

[2] This means that there are input locations i and j such that $\Pr[P(x) = x_i \oplus x_j] \neq \frac{1}{2}$.

1.1 Our Results

We study the complexity of predicates P in Goldreich's PRG constructions that give us polynomial-stretch PRGs. In particular, we look at two measures of complexity of a predicate P, namely its $\mathbb{Q}$-degree and the minimum depth of a decision tree that computes it (we call this the DT-complexity of P), and show positive and negative results.

On the negative side, we show that no predicates with $\mathbb{Q}$-degree at most 2 or decision tree depth at most 3 can result in a secure PPRG.[3] Previously, Mossel, Shpilka and Trevisan [MST06] who show that locality-4 predicates cannot result in a secure PRG.

Theorem 1.1 (Informal)**.** *All predicates computable by a degree-2 polynomial over $\mathbb{Q}$ or a depth 3 decision tree are either affine or fail to be pairwise-independent. Subsequently, Goldreich's PRG is insecure when instantiated with such predicates for $m = \Omega(n^{1+\epsilon})$.*

On the positive side, we construct a predicate P which has locality decision-tree depth 4, $\mathbb{Q}$-degree 3 and $\mathbb{F}_2$-degree 2 which is "Goldreich-friendly". That is, Goldreich's PRG instantiated with P resists precisely the class of attacks considered in [ABR12, OW14, KMOW17], namely $\mathbb{F}_2$-linear attacks and attacks that can be implemented in the Lasserre/Parrilo sum-of-squares (SOS) hierarchy.

Our predicate TSPA ("tri-sum-paired-and") is defined as follows

$$f(x_1, x_2, x_3, x_4, x_5) = x_1 \oplus x_2 \oplus x_3 \oplus (x_2 \oplus x_4)(x_3 \oplus x_5).$$

and is inspired by work of Ambainis [Amb06] who constructs a similar predicate in the context of quantum query complexity.

Theorem 1.2. *There is a non-degenerate predicate $Q : \{0,1\}^5 \to \{0,1\}$ on 5 variables which is computable by a degree 3 polynomial over $\mathbb{Q}$ as well as by a depth 4 decision tree. When instantiated with Q, Goldreich's PRG is (subexponentially) secure against $\mathbb{F}_2$-linear attacks for all $m = O(n^{1.25-\epsilon})$ as well as attacks from the Lasserre/Parrilo sum-of-squares hierarchy for all $m = O(n^{1.5-\epsilon})$.*

1.2 An Open Problem

For applications to obfuscation, however, our results are not the end of the story. Lin's result in [Lin16b] explicitly proves the "d vs. $3d+2$" tradeoff, but implicit in [Lin16b] is the following result.

Theorem 1.3. *Suppose that there exists a PRG with $\mathbb{Q}$-degree d and stretch $n^{s+\epsilon}$ for some $\epsilon > 0$. Then, indistinguishability obfuscation can be constructed from cryptographic D-linear maps for any $D \geq \frac{3d}{\lfloor s \rfloor}$.*

Therefore, what we are interested in optimizing is the *ratio* of the degree d of a candidate PRG to the exponent s of its polynomial stretch. In particular, we ask a generalization of our earlier question:

For a fixed constant d, what is the largest stretch a $\mathbb{Q}$-degree d PRG can attain?

For example, does there exist a degree d PRG obtaining stretch $n^{\lceil \frac{3}{4}d \rceil + \epsilon}$? If so, then Theorem 1.3 implies that IO can be constructed from 4-linear maps. Both of the above questions remain entirely open.

[3] This also yields a negative result for secure predicates with $\mathbb{F}_p$-degree 2 for $p > 2^k$.

Organization. In Section 2, we review the definitions and concepts relevant to the paper, and in Section 3 we formally introduce Goldreich's PRG and discuss previous results related to its security. In Section 4, we present new candidate predicates for Goldreich's PRG achieving $\mathbb{Q}$-degree and decision tree-complexity less than 5, and in Sections 5 and 6 we prove the negative results stated in Theorem 1.1.

2 Preliminaries

Notation. We let U_n denote the uniform distribution on n bits, i.e., on the set $\{0,1\}^n$. Let $\mathsf{negl}(n) : \mathbb{N} \to \mathbb{R}$ denote any function that is smaller than any inverse polynomial in n. That is, we require that for every polynomial p, there is an $n_p \in \mathbb{N}$ such that for all $n > n_p$, $\mathsf{negl}(n) < 1/p(n)$.

2.1 Pseudorandom generators

We say that a function $G : \{0,1\}^n \to \{0,1\}^m$ is a *pseudorandom generator* (PRG) if it has the following properties: (1) G is computable in (uniform) time $\mathrm{poly}(n)$, and (2) any probabilistic polynomial time adversary $A : \{0,1\}^m \to \{0,1\}$ has the property that

$$\left| \mathop{\mathbf{E}}_{x \leftarrow U_n} [A(G(x))] - \mathop{\mathbf{E}}_{y \leftarrow U_m} [A(y)] \right| = \mathsf{negl}(n)$$

We say that a PRG $G : \{0,1\}^n \to \{0,1\}^m$ has *stretch* $m = m(n)$. In this paper, we focus on the *polynomial stretch* regime, namely where $m = O(n^c)$ for some constant $c > 1$.

If G is computable in NC^0, then we define the *locality* of G to be the maximum number of input bits on which any output bit of G depends.

As positive evidence for a particular function $G : \{0,1\}^n \to \{0,1\}^m$ to be a PRG, one can prove that G resists attacks by more specific kinds of adversaries (instead of all polynomial-time adversaries). We focus on two kinds of attacks here. First, we consider $\mathbb{F}_2$-linear attacks.

Definition 1 (small-bias generator)**.** A polynomial-time computable function $G : \{0,1\}^n \to \{0,1\}^m$ is a *small-bias generator* if the advantage of every $\mathbb{F}_2$-linear function $A : \{0,1\}^m \to \{0,1\}$ in distinguishing between the distributions $G(U_n)$ and U_m is $\mathsf{negl}(n)$. See [NN93] for a general exposition on small-bias generators.

The other class of attacks we consider are those from the **Lasserre/Parrilo sum-of-squares (SOS) hierarchy**, a powerful SDP hierarchy which generalizes the Sherali-Adams (SA_+) and Lovász-Schrijver (LS_+) proof systems. See [BS16] for a general introduction to the SOS hierarchy. As we will see, SDP-based attacks are of particular interest to us because the security of Goldreich's PRG is closely tied to the problem of refuting random instances of CSPs.

2.2 Analysis of Boolean Functions

Let $f : \{0,1\}^k \to \{0,1\}$ be a function on k variables.

Definition 2 (bias)**.** f is *balanced*, or *unbiased*, if $\mathop{\mathbf{E}}_{x \sim U_k} f(x) = \frac{1}{2}$; otherwise, we say that f is *biased*.

Definition 3 (t-wise independence)**.** For $t \geq 0$, f is t-wise independent if for all $s \leq t$ and all sets of s variables $\{x_{i_1}, ..., x_{i_s}\}$, the function $f(x) \oplus x_{i_1} \oplus ... \oplus x_{i_s}$ is balanced.

Finally, we define two complexity measures of f besides its locality: degree and decision tree complexity.

Definition 4 (degree)**.** For any field K, the *K-degree* of f is the degree of the unique multilinear polynomial over K computing f.

For our purposes, the fields of interest are $K = \mathbb{Q}$ and $K = \mathbb{F}_2$, giving us notions of $\mathbb{Q}$-degree and $\mathbb{F}_2$-degree.

Definition 5 (decision tree)**.** A *decision tree* is a directed binary tree T in which each non-leaf vertex is labeled by a variable x_i and each leaf is labeled by a bit $b \in \{0,1\}$. Any $v \in \{0,1\}^k$ defines a path in T which starts at the root of T and traverses from a vertex labeled by x_i to its *left child* if $v_i = 0$ and to its *right child* if $v_i = 1$. We say that a decision tree T computes a function f if the following holds: for each $v \in \{0,1\}^k$, the leaf at the end of the path in T defined by v is labeled by $f(v)$.

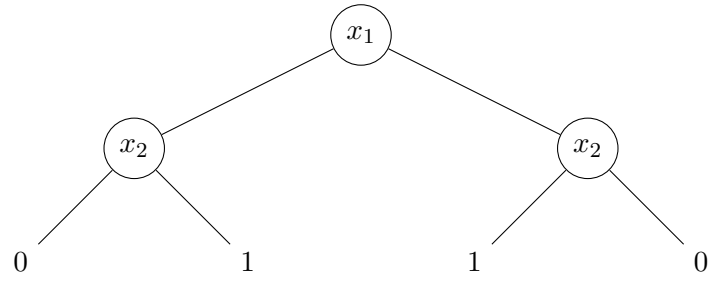


Figure 1: This is a depth 2 decision tree computing the function $f(x_1, x_2) = x_1 \oplus x_2$. We only consider nodes labeled by variables when calculating the depth of a decision tree

Definition 6 (decision tree complexity)**.** Let $f : \{0,1\}^k \to \{0,1\}$ be a boolean function. The *decision tree complexity (DT-complexity)* of f is the smallest depth of a decision tree computing f.

Our three main complexity measures – $\mathbb{Q}$-degree, DT-complexity, and locality – are related to each other in the following way.

Lemma 2.1. *Let $f : \{0,1\}^k \to \{0,1\}$ be a boolean function with $\mathbb{Q}$-degree d and DT-complexity D. Then $d \leq D \leq k$.*

Proof. To see that $D \leq k$, note that there exists a (complete) depth k decision tree T computing f in which the ith layer of T consists only of variables x_i and the "$k+1$th layer" of T consists of 2^k leaves spelling out the truth table for f. To see that $d \leq D$, note that a decision tree with root x_i and children T_L and T_R has the property that $T(x) = x_i T_R(x) + (1 - x_i)T_L(x)$ (where $T(x) := f(x)$ when T computes f); a quick induction allows us to conclude that $T(x)$ can be written as a rational polynomial of degree at most D, as desired. □

3 A Review of Goldreich's PRG and its Security

Goldreich's candidate pseudorandom generator, first introduced in [Gol00] (then as a candidate one-way function), can be instantiated with any k-ary predicate $P : \{0,1\}^k \to \{0,1\}$ and any k-uniform (directed) hypergraph H on n vertices and m hyperedges. Given H and P, we identify each vertex in H with an index in $[n]$ and each hyperedge with an index $i \in [m]$. For each $i \in [m]$, let $\Gamma_H(i) \in [n]^k$ be the sequence of k vertices in the ith hyperedge. Then, Goldreich's PRG is the function from $\{0,1\}^n$ to $\{0,1\}^m$ defined by

$$G_{H,P}(x) = \left(P(x|_{\Gamma_H(i)})\right)_{i\in[m]}.$$

That is, the ith bit of $G_{H,P}(x)$ is the output of P when given the $\Gamma_H(i)$-restriction of x as input.

Goldreich's generator is often instantiated with a *uniformly random* choice of hypergraph H; in this setting, we say that "Goldreich's generator instantiated with P is a PRG" for some predicate P if for a random k-uniform hypergraph H, $G_{H,P}$ is a PRG with high probability (say, probability $1 - o(1)$). Often (see [AL16, OW14, ABR12]) instead of proving results for random hypergraphs it suffices to use hypergraphs with "good expansion" for varying definitions of expansion. Unless otherwise stated, references to "Goldreich's PRG" assume the uniformly random hypergraph setting.

It is sometimes useful to think of the hypergraph H and predicate P as defining a constraint satisfaction problem (CSP) using predicates P and $\neg P$ as constraints: the task of breaking Goldreich's PRG can be thought of distinguishing a random planted instance of this CSP from a truly random instance of the CSP. From this point of view, it is natural to consider SDP-based attacks and desirable to have security results against such attacks. For a more in-depth survey and discussion of Goldreich's PRG, see [App16].

3.1 The Choice of Predicates in Goldreich's PRG

Many predicates $P : \{0,1\}^k \to \{0,1\}$ are known to yield *insecure* PRGs when plugged into Goldreich's generator $G_{H,P}$. Define a predicate P to be degenerate if it is either affine or correlated to some pair of its input bits.

Definition 7. A predicate $P : \{0,1\}^k \to \{0,1\}$ is called *degenerate* if it is either

(a) *affine*: that is, the $\mathbb{F}_2$-degree of P is at most 1; or

(b) *not pairwise-independent*: that is, there is a pair of input locations i and j such that $\Pr[P(x) = x_i \oplus x_j] \neq \frac{1}{2}$.

If neither of the above conditions hold, P is called *non-degenerate*.

There have been many attacks on degenerate predicates [MST06, BQ12, ABR12]. Perhaps most interestingly, Applebaum, Bogdanov, and Rosen [ABR12] show the following result.

Theorem 3.1 ([ABR12], Theorem 2)**.** *If P is degenerate, then Goldreich's PRG is insecure when instantiated with P as long as $m = n + \Omega(n)$. In fact, the PRG has $\Omega(\frac{1}{\log(n)})$ bias with probability $1 - o(1)$.*

So when using a degenerate predicate, Goldreich's PRG cannot even pass all $\mathbb{F}_2$-linear tests. Theorem 3.1 is a result in the "random hypergraph" model, but in some cases (e.g., when P has locality at most 4), even stronger insecurity results are known. In [MST06], an efficient (but non-linear) attack is described on Goldreich's PRG when instantiated with any predicate of locality at most 4 (which is necessarily degenerate) and *any* hypergraph H.

On the other hand, the "tri-sum-and" predicate

$$\text{TSA}(x_1, x_2, x_3, x_4, x_5) = x_1 \oplus x_2 \oplus x_3 \oplus x_4 x_5$$

is an oft-cited candidate predicate for Goldreich's PRG. In [MST06], Mossel, Shpilka, and Trevisan show that when instantiated with the TSA predicate, a variant of Goldreich's PRG is secure against $\mathbb{F}_2$-linear attacks (that is, the PRG has subexponentially small bias) for $m = O(n^{1.25-\epsilon})$, and in [OW14], O'Donnell and Witmer extend this result (again for TSA) to $m = O(n^{1.5-\epsilon})$ in the uniformly random hypergraph setting. Moreover, they show that up to $m = O(n^{1.5-\epsilon})$, the PRG is subexponentially secure against the SOS hierarchy (in a slightly modified random hypergraph setting).

There have also been results proving that Goldreich's PRG is secure against various attacks as long as the chosen predicate P is non-degenerate. Applebaum, Bogdanov, and Rosen [ABR12] show the following converse to Theorem 3.1.

Theorem 3.2 ([ABR12], Theorem 1)**.** *If P is non-degenerate, then for $m = O(n^{1.25-\epsilon})$ Goldreich's PRG has subexponentially small bias with probability $1 - o(1)$.*

This does not match the stretch of $n^{1.5-\epsilon}$ achieved in [OW14] for the TSA predicate; however, there are no known attacks against any non-degenerate predicate in this setting for $m = o(n^{1.5})$, and it remains an open problem to extend Theorem 3.2 to stretch $m = O(n^{1.5-\epsilon})$. However, it does show that any non-degenerate predicate gives rise to a small bias generator with polynomial stretch.

As for SOS lower bounds, a recent result of Kothari, Mori, O'Donnell, and Witmer [KMOW17] matches the lower bounds in [OW14] for any pairwise-independent predicate (and hence for any non-degenerate predicate).

Theorem 3.3 ([KMOW17], Theorem 1.2, rephrased)**.** *If P is pairwise-independent, then Goldreich's PRG instantiated with P is secure against the SOS hierarchy up to degree $\tilde{\Omega}(\frac{n^3}{m^2})$. Therefore, against SOS attacks the PRG has subexponential security up to stretch $m = n^{1.5-\epsilon}$.*

In summary, given these results, when looking for candidate predicates that could achieve polynomial stretch (in fact, stretch up to $n^{1.5-\epsilon}$), we take the view that any non-degenerate predicate is a viable candidate.

4 New Candidate Predicates for Goldreich's PRG

In this section, we focus on the problem of finding non-degenerate predicates of minimal decision tree (DT)-complexity and $\mathbb{Q}$-degree, respectively. Previously, no such predicates were known to have either DT-complexity or $\mathbb{Q}$-degree less than 5, while the commonly used non-degenerate predicate

$$\text{TSA}(x_1, ..., x_5) = x_1 \oplus x_2 \oplus x_3 \oplus x_4 x_5$$

has DT-complexity 5 and $\mathbb{Q}$-degree 5.

4.1 A Predicate with Decision Tree-Complexity 4

As a first step, we come up with a predicate that has DT-complexity 4 and $\mathbb{Q}$-degree 4. Indeed, the TSA predicate can be modified in a simple way to decrease its DT-complexity down to 4 while maintaining pairwise independence. Consider the predicate

$$P(x_1, ..., x_5) = x_1 \oplus x_2 \oplus \overline{x_5}x_3 \oplus x_5x_4$$

Theorem 4.1. *The predicate P defined above has a decision tree of depth 4, has $\mathbb{Q}$-degree 4, and is non-degenerate (namely, pairwise independent and not affine).*

Proof. We first show that P can be computed by a decision tree of depth 4. First, note that the functions $x_1 \oplus x_2 \oplus x_3$ and $x_1 \oplus x_2 \oplus x_4$ both have DT-complexity 3, as they are functions of three variables each. Now P can be computed by a depth-4 decision tree which first queries for x_5 and, depending on whether x_5 is 0 or 1, computes the function $x_1 \oplus x_2 \oplus x_3$ or $x_1 \oplus x_2 \oplus x_4$ respectively.

Since $\mathbb{Q}$-degree is bounded above by DT-complexity, this means that $\deg_{\mathbb{Q}}(P) \leq 4$ as well. In fact, P has degree exactly 4, as P can be written as

$$P(x) = x_1 \oplus x_2 \oplus (x_5x_4 + (1 - x_5)x_3)$$

It is easy to see that P has a nonzero $x_1x_2x_4x_5$ term (as well as a nonzero $x_1x_2x_3x_5$ term) when written as a multilinear polynomial over $\mathbb{Q}$.

To see that P is pairwise independent, note that P can be written as

$$P(x_1, ..., x_5) = x_1 \oplus x_2 \oplus x_3 \oplus x_5(x_3 \oplus x_4).$$

It suffices to show that all the functions P, $P \oplus x_i$ ($i \in \{1, 2, 3, 4, 5\}$) and $P \oplus x_i \oplus x_j$ ($i, j \in \{1, 2, 3, 4, 5\}$) are all balanced.

First, note that P is balanced. Secondly, note that $P(x) \oplus x_i$ is also balanced because it will have either an independent x_1 summand or an independent x_2 summand remaining. Finally, note that the same clearly holds for $P(x) \oplus x_i \oplus x_j$ except for the case of $P(x) \oplus x_1 \oplus x_2 = x_3 \oplus x_5(x_3 \oplus x_4)$. In this last case, choosing $(x_3, x_4, x_5) \in \{0, 1\}^3$ independently at random is the same as choosing $(x_3, x_3 \oplus x_4, x_5) \in \{0, 1\}^3$) independently at random. With this change of variables, one sees that $P(x) \oplus x_1 \oplus x_2$ is balanced as well. Finally, we note that P is clearly not affine. □

4.2 A Predicate with Decision-Tree Complexity 4 and $\mathbb{Q}$-degree 3

One way to try to find a predicate with $\mathbb{Q}$-degree smaller than 4 is to find one with decision trees of depth smaller than 4. Unfortunately, as we will see in Section 5, there are *no non-degenerate* predicates with DT-complexity 3, so we cannot achieve $\mathbb{Q}$-degree 3 by looking for shallower decision trees.

Moreover, the predicate P defined in Section 4.1 has $\mathbb{Q}$-degree 4. Indeed, at least in low degree, there are not too many examples of Boolean functions exhibiting a gap between DT-complexity and $\mathbb{Q}$-degree. One function which *does* exhibit such a gap, studied by Ambainis [Amb06] for its applications to quantum query complexity, is the function

$$f(x_1, x_2, x_3, x_4) = x_1 + x_2 + x_3x_4 - x_1x_2 - x_2x_3 - x_1x_4.$$

The function f, which has $\mathbb{Q}$-degree 2 and DT-complexity 3, is our starting point. In [Amb06], the sensitivity properties of f were used to obtain nontrivial quantum query complexity lower bounds for functions obtained as iterated compositions of f with itself, which leads to a gap between quantum query complexity and $\mathbb{Q}$-degree; see [Amb06] for a longer discussion.

Lemma 4.2 ([Amb06])**.** *The function f defined above has $\mathbb{Q}$-degree 2 and DT-complexity 3.*

For our purposes, the key additional insight is that f can be rewritten in the form

$$f(x_1, x_2, x_3, x_4) = x_1 \oplus x_2 \oplus (x_1 \oplus x_3)(x_2 \oplus x_4).$$

This leads to the following result.

Lemma 4.3. *The function f defined above is 1-wise independent.*

Proof. (of Lemma 4.3.) Note that f can be written as

$$\begin{aligned} f(x_1, x_2, x_3, x_4) &= x_1 + x_2 + x_3x_4 - x_1x_2 - x_2x_3 - x_1x_4 \\ &\equiv x_1 + x_2 + (x_1 + x_3)(x_2 + x_4) \pmod 2. \end{aligned}$$

Note also that choosing $(x_1, x_2, x_3, x_4) \in \{0,1\}^4$ independently at random is equivalent to choosing $(x_1, x_2, x_1 \oplus x_3, x_2 \oplus x_4)$ independently at random, so $f(x)$ is balanced and $f(x) \oplus x_i$ is balanced for any choice of i (for $i = 1$ and $i = 3$, we have an "independent" x_2 summand, while for $i = 2$ and $i = 4$, we have an "independent" x_1 summand). Thus, f is 1-wise independent. □

Since f is 1-wise independent, $f(x) \oplus x_5$ is a pairwise-independent function of 5 variables. Renaming variables for convenience, we have our candidate predicate:

$$\begin{aligned} \text{TSPA}(x) &:= x_1 \oplus (x_2 + x_3 + x_4x_5 - x_2x_3 - x_3x_4 - x_2x_5) \\ &\equiv x_1 + x_2 + x_3 + (x_2 + x_4)(x_3 + x_5) \pmod 2. \end{aligned}$$

By analogy to the usual TSA predicate, we call this the *TSPA predicate*, or "tri-sum-paired-and" predicate, as x_4 and x_5 are each paired (via XOR) with an earlier variable before being AND-ed together. TSPA(x) has $\mathbb{F}_2$-degree 2, $\mathbb{Q}$-degree 3, DT-complexity 4, locality 5, where the statements about the $\mathbb{Q}$-degree and DT-complexity follow immediately from Lemma 4.2. Thus, we have:

Theorem 4.4. *There is a predicate TSPA which is non-degenerate and has:*

- *$\mathbb{F}_2$-degree 2;*
- *$\mathbb{Q}$-degree 3;*
- *DT-complexity 4; and*
- *locality 5.*

We will see in Sections 5 and 6 that TSPA(x) is optimal according to all four of these complexity measures.

5 Predicates with Depth-3 Decision Trees Yield Insecure PRGs

To show that predicates with DT-complexity at most 3 lead to insecure PRGs, by [BQ12] and [ABR12] it suffices to show that any P computable by a depth 3 decision tree is degenerate, i.e. either (1) affine or (2) *not* pairwise-independent. In order to do this, we first suppose that P is an unbiased predicate computable by a depth 3 decision tree and argue that the further condition of 1-wise independence constrains the number of variables that can appear in the decision tree. Namely, 1-wise independence constrains the "leaf variables" as defined below.

Definition 8. Let T be a decision tree computing a predicate P on variables $x_1, ..., x_n$ which is *minimal* in the sense that both 0s and 1s appear as descendants of any variable node in T. We say that x_i is a *leaf variable* if all nodes labeled by x_i have no variable children.

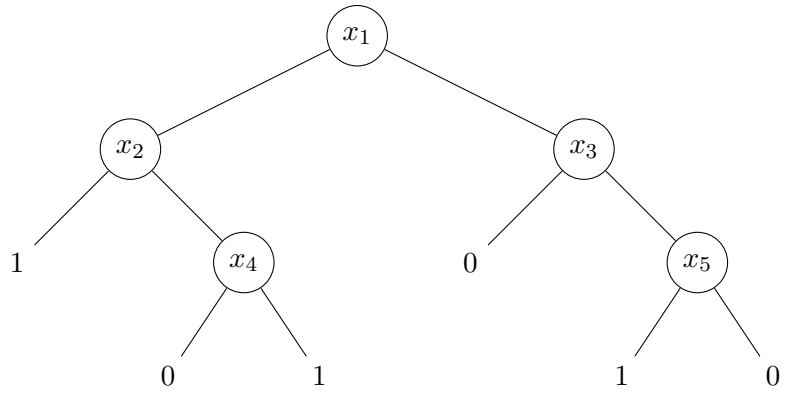


Figure 2: This is a depth 3 decision tree computing a balanced predicate P on five variables. The variables x_4 and x_5 are "leaf variables" in this tree. Note that this predicate P is correlated with the values of x_4 and x_5, respectively.

Lemma 5.1. *Suppose that P is a balanced predicate on n variables and T is a minimal decision tree (in the sense of Definition 8) computing P. If x_i is a leaf variable which occurs only once in T, then $P \oplus x_i$ is biased.*

Proof. Suppose that the variable x_i occurs at depth d within the tree and has left child b and right child $1 - b$. Then, we have that

$$\mathbf{E}[P \oplus x_i] = \frac{1}{2} \cdot \mathbf{E}[P(x) \mid x_i = 0] + \frac{1}{2} \cdot \mathbf{E}[1 - P(x) \mid x_i = 1]$$

$$= \frac{1}{2}\left(2^{-d}b + (1 - 2^{-d}) \cdot \frac{1}{2}\right) + \frac{1}{2} - \frac{1}{2}\left(2^{-d}(1-b) + (1 - 2^{-d}) \cdot \frac{1}{2}\right)$$

$$= \frac{1}{2} + \frac{1}{2}2^{-d}(2b - 1) \neq \frac{1}{2}.$$

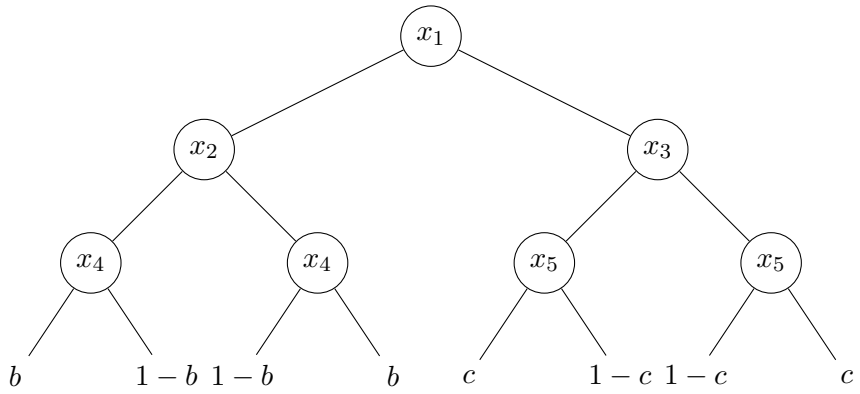
The evaluation of $\mathbf{E}[P(x) \mid x_i = 0]$ follows from further conditioning on the event that the node containing x_i is reached when evaluating the decision tree. We conclude that under the given hypotheses, $P \oplus x_i$ is biased, as claimed. □

As a result, we have the following corollary

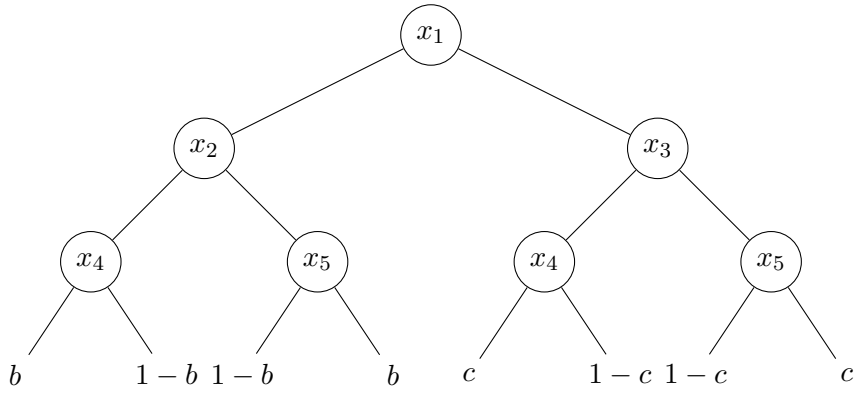
Corollary 5.2. *Suppose that P is a 1-wise independent predicate on n variables and T is a minimal decision tree computing P. Then every leaf variable x_i occurs at least twice in T.*

In the particular case we are considering, i.e. when T has depth at most 3, this gives an upper bound on the number of variables occurring in T: it is not hard to see that if P is 1-wise independent, T can have at most 5 distinct variables. We already know from [MST06] that predicates in at most 4 variables are degenerate, so all we have to do is analyze the depth 3, locality 5 case. Up to renaming of variables, there are only three possible tree structures for such a 1-wise independent predicate, shown below.

Case 1:



Case 2:



Case 3:

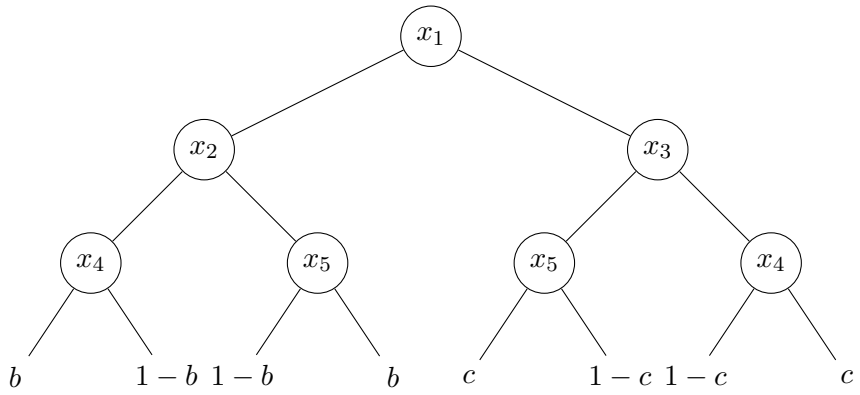


Figure 3: Locality 5, 1-wise independent predicates computable by depth 3 decision trees.

These are the only possible structures because (1) space constraints tell us that x_4 and x_5 (the leaf variables) must each occur exactly twice, and (2) the fact that $P \oplus x_4$ must be unbiased tells us that if b is the left child of one occurrence of x_4, then $1-b$ must be the left child of the other occurrence of x_4 (and similarly for x_5). We now analyze these three cases to show that none of these predicates are 2-wise independent.

In **Case 1**, the predicate P is correlated with $x_2 \oplus x_4$ (the left branch of x_1 is exactly the function $x_2 \oplus x_4 \oplus b$, so the same analysis as in Lemma 5.1 works), so P is never 2-wise independent.

In **Case 2** and **Case 3**, P is also correlated with $x_2 \oplus x_4$. To see this, we compute

$$\mathbf{E}[P(x) \oplus x_2 \oplus x_4] = \frac{1}{2}\mathbf{E}[P(x) \oplus x_2 \oplus x_4 \mid x_1 = 0] + \frac{1}{4}$$

(as when $x_1 = 1$, $P(x)$ is independent of x_2 and so $P(x) \oplus x_2 \oplus x_4$ is unbiased)

$$= \frac{1}{4}\mathbf{E}[P(x) \oplus x_2 \oplus x_4 \mid x_1 = x_2 = 0] + \frac{3}{8}$$

(as when $x_1 = 0$ and $x_2 = 1$, $P(x)$ is independent of x_4, and so $P(x) \oplus x_2 \oplus x_4$ is unbiased)

$$= \frac{1}{4}b + \frac{3}{8} \neq \frac{1}{2}.$$

This completes the case analysis, showing that if P is computable by a depth 3 decision tree, then P is degenerate, in which case Goldreich's PRG is insecure when instantiated with P.

6 Predicates with $\mathbb{Q}$-degree 2 Yield Insecure PRGs

In this section, we show that all predicates of $\mathbb{Q}$-degree 2 lead to insecure PRGs. We do this by bounding the locality of a predicate as a function of its $\mathbb{Q}$-degree. In particular, we show that any predicate P expressible as a rational polynomial of degree 2 has locality at most 4. Our bound sharpens the best previous result along these lines due to Nisan and Szegedy in [NS94], in which

it is shown that a predicate of degree d has locality at most $d \cdot 2^d$, giving an upper bound of locality 8 in our case. It may be of independent interest to tighten the result of [NS94] for larger values of d.

Theorem 6.1. *Any predicate P expressible as a rational polynomial of degree at most 2 depends on at most 4 input bits.*

Theorem 6.1 implies the desired result because we already know that predicates of locality at most 4 are degenerate (and so yield insecure PRGs – even, by [MST06], with an arbitrary hypergraph H instead of a random hypergraph), so we proceed directly to the proof of Theorem 6.1.

Proof. Since boolean functions of degree at most 1 are all of the form $P(x) = 0$, $P(x) = 1$, $P(x) = x_i$, or $P(x) = 1 - x_i$ for some i, we restrict to predicates of degree exactly 2. We prove by induction on $k \geq 5$ the following statement: a k-ary predicate P of degree 2 depends on at most 4 variables. We prove the base case of $k = 5$ using a computer search and then prove the inductive step by hand.

The case $k = 5$:

For $0 \leq j \leq 5$, let N_j denote the number of boolean functions on j variables which are also polynomials of degree at most 2, and let $\tilde{N}_j$ denote the number of such boolean functions that depend on *exactly* j variables. Then, we see by a counting argument that

$$N_j = \sum_{i=0}^{j} \binom{j}{i} \tilde{N}_i.$$

We calculate by computer search that $N_0 = 2, N_1 = 4, N_2 = 16, N_3 = 70, N_4 = 222$, and $N_5 = 552$. Using these values, we calculate that $\tilde{N}_5 = 0$, proving the base case.

The inductive step:

Suppose that for some $k \geq 5$ we know that all k-ary predicates of degree 2 have locality at most 4. We now want to show that the same is true for $k+1$-ary predicates. Consider any $k+1$-ary predicate $P(x_0, ..., x_k)$ of degree 2, which (assuming without loss of generality that P depends on x_0) can be written as a polynomial expression

$$P(x_0, ..., x_k) = x_0 P_1(x_1, ..., x_k) + P_2(x_1, ..., x_k)$$

where $P_1(x_1, ..., x_k)$ is some polynomial of degree at most 1 and $P_2(x_1, ..., x_k)$ is some polynomial of degree at most 2. Because we have that

$$P(0, x_2, ..., x_k) = P_2(x_1, ..., x_k),$$

we see that P_2 – a degree 2 polynomial – computes a boolean function on k variables.

Furthermore, we claim the following about P_2: either every present variable in P_2 occurs in a degree 2 term of P_2, or P_2 is has degree at most 1. This is true for the following reason: if the variable x_1 (without loss of generality) occurs only as a degree 1 term in P_2, then we can write $P_2(x) = ax_1 + P_2'(x_2, ..., x_k)$ where $P_2'(x_2, ..., x_k) = P_2(0, x_2, ..., x_k)$ is a boolean function and

$ax_1 = P_2(x_1, 0, ..., 0) - P_2'(0, ..., 0)$ implies that $a = \pm 1$. If P_2' is nonconstant (i.e. if P_2 has degree 2), then we obtain a contradiction because we can force P_2 to take either the value -1 or 2 (depending on whether $a = 1$ or $a = -1$) on a suitable choice of x. Thus, either P_2 has degree at most 1 or every variable in P_2 occurs in a degree 2 term.

By similar argument to the previous paragraph (applied to P instead of P_2), we see that either P has degree at most 1 (and so depends on at most one variable, so we are done) or x_0 appears in a degree 2 term in P, i.e. P_1 has degree exactly 1.

Finally, we use the fact that $P_1(x_1, ..., x_k) + P_2(x_1, ..., x_k) = P(0, x_1, ..., x_k)$ is also a boolean function on k variables, and hence depends on at most 4 variables by the inductive hypothesis. In the case that P_2 has degree 2, we note that any variable occurring in the expression for P_1 *or* occurring in the expression for P_2 necessarily occurs in the expression for $P_1 + P_2$, because P_1 has degree 1 and all variables occurring in P_2 occur in a degree 2 term, so there can be no cancellation when adding P_1 and P_2. This allows us to conclude that $P(x) = x_0 P_1(x) + P_2(x)$ depends on at most 5 variables (the variables that P_1 and P_2 depend on along with x_0), implying that P depends on at most 4 variables by the base case.

On the other hand, if P_2 has degree exactly 0 then the same argument as above applies, while if P_2 has degree exactly 1, then without loss of generality $P_2(x) = x_1$ (applying a logical negation to P and changing variable names if necessary). If $P_1(x_1, ..., x_k)$ does not contain the exact term "$-x_1$", then the argument of the previous paragraph applies, while in this special case we have

$$P(x) = -x_0 x_1 + x_1 + x_0 P_3(x_2, ..., x_k)$$

where $P_3(x_2, ..., x_k) = P(1, 0, x_2, ..., x_k)$ is a boolean function of degree at most 1. We conclude that P_3 depends on at most one variable, and so P depends on at most 3 variables in this degenerate case. Having covered all cases, this completes the inductive step and hence proves Theorem 6.1. □

7 Conclusion

We are left with many interesting and unresolved questions. Firstly, we ask,

Question 7.1. *What additional security properties can we prove (or disprove) about Goldreich's PRG when instantiated with the TSPA predicate and stretch $n^{1.5-\epsilon}$? What about with stretch $n^{1.01}$?*

More concretely, there remains a gap between the stretch ($O(n^{1.25-\epsilon})$) at which we know that Goldreich's PRG (instantiated with non-degenerate predicates) has small bias and the stretch ($\tilde{\Omega}(n^{1.5})$) at which we know efficient attacks on the PRG. In particular, small bias guarantees up to stretch $O(n^{1.5-\epsilon})$ would provide stronger evidence that Goldreich's PRG is secure when instantiated with the TSPA predicate.

Additionally, in [AL16], another class of attacks on local PRGs is studied, namely *algebraic attacks* (attacks from the Polynomial Calculus proof system). The lower bounds proved in [AL16] are insufficient to establish the security of Goldreich's PRG with the TSPA predicate against these attacks, but the established upper bounds do not rule out the PRG. Closing the gap between upper and lower bounds here would be extremely interesting and would also be relevant to higher degree candidate PRGs, whose security properties are more uncertain.

Furthermore, while this paper focused on achieving any polynomial stretch, it is both intrinsically interesting and and potentially useful for cryptographic applications to understand the

tradeoff between the $\mathbb{Q}$-degree of a PRG and the maximum stretch it can attain. In particular, a positive answer to the following question would yield an IO construction from 4-linear maps.

Question 7.2. *Can a $\mathbb{Q}$-degree d PRG achieve stretch $n^{\lceil \frac{3}{4}d \rceil + \epsilon}$?*

This degree-stretch tradeoff is not currently well understood; even the fact that $\mathbb{Q}$-degree 3 is potentially achievable was not known before this work.

Question 7.3. *What is the maximum stretch achievable by a PRG of $\mathbb{Q}$-degree 3?*

Question 7.4. *Does there exist a 3-wise independent boolean function with $\mathbb{Q}$-degree 3 and $\mathbb{F}_2$-degree 3?*

Even an answer to this elementary question would have interesting implications. If $f : \{0,1\}^k \to \{0,1\}$ were such a predicate, then the predicate $g : \{0,1\}^{k^2} \to \{0,1\}$ defined by

$$g(x_1, ..., x_{k^2}) = f(f(x_1, ..., x_k), f(x_{k+1}, ..., x_{2k}), ..., f(x_{k^2-k+1}, ..., x_{k^2}))$$

would have $\mathbb{Q}$-degree 9, $\mathbb{F}_2$-degree 9, and would be 15-wise independent. This would be a viable candidate predicate to achieve stretch $n^{7+\epsilon}$ in degree 9, which – if secure – would positively answer Question 7.2. It remains unclear whether or not such a degree 3 predicate should exist, or how to find one if it does exist.

References

[ABR12] Benny Applebaum, Andrej Bogdanov, and Alon Rosen. A dichotomy for local small-bias generators. In *Theory of Cryptography Conference*, pages 600–617. Springer, 2012.

[AIK06] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography in NC^0. *SIAM Journal on Computing*, 36(4):845–888, 2006.

[AL16] Benny Applebaum and Shachar Lovett. Algebraic attacks against random local functions and their countermeasures. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1087–1100. ACM, 2016.

[Amb06] Andris Ambainis. Polynomial degree vs. quantum query complexity. *Journal of Computer and System Sciences*, 72(2):220–238, 2006.

[App16] Benny Applebaum. Cryptographic hardness of random local functions. *Computational complexity*, 25(3):667–722, 2016.

[AS16] Prabhanjan Ananth and Amit Sahai. Projective arithmetic functional encryption and indistinguishability obfuscation from degree-5 multilinear maps. *IACR Cryptology ePrint Archive*, 2016:1097, 2016.

[BQ12] Andrej Bogdanov and Youming Qiao. On the security of Goldreich's one-way function. *Computational complexity*, 21(1):83–127, 2012.

[BS16] Boaz Barak and David Steurer. Proofs, beliefs, and algorithms through the lens of sum-of-squares. *Course notes: http://www.sumofsquares.org/public/index.html*, 2016.

[Gol00] Oded Goldreich. Candidate one-way functions based on expander graphs. *IACR Cryptology ePrint Archive*, 2000:63, 2000.

[IKOS08] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Cryptography with constant computational overhead. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 433–442, 2008.

[KMOW17] Pravesh K. Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer. Sum of squares lower bounds for refuting any CSP. *arXiv preprint arXiv:1701.04521*, 2017.

[Lin16a] Huijia Lin. Indistinguishability obfuscation from constant-degree graded encoding schemes. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, pages 28–57, 2016.

[Lin16b] Huijia Lin. Indistinguishability obfuscation from DDH on 5-linear maps and locality-5 PRGs. *Preprint: http://eprint.iacr.org/2016/1096.pdf*, 2016.

[LV16] Huijia Lin and Vinod Vaikuntanathan. Indistinguishability obfuscation from DDH-like assumptions on constant-degree graded encodings. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 11–20. IEEE, 2016.

[MST06] Elchanan Mossel, Amir Shpilka, and Luca Trevisan. On ε-biased generators in NC^0. *Random Structures & Algorithms*, 29(1):56–81, 2006.

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

[NS94] Noam Nisan and Mario Szegedy. On the degree of boolean functions as real polynomials. *Computational complexity*, 4(4):301–313, 1994.

[OW14] Ryan O'Donnell and David Witmer. Goldreich's PRG: evidence for near-optimal polynomial stretch. In *Computational Complexity (CCC), 2014 IEEE 29th Conference on*, pages 1–12. IEEE, 2014.