Limits on the Locality of Pseudorandom Generators and Applications to Indistinguishability Obfuscation

Alex Lombardi*
MIT

Vinod Vaikuntanathan†
MIT

Abstract

Lin and Tessaro (ePrint 2017) recently proposed indistinguishability obfuscation (IO) and functional encryption (FE) candidates and proved their security based on two assumptions: a standard assumption on bilinear maps and a non-standard assumption on "Goldreich-like" pseudorandom generators. In a nutshell, their second assumption requires the existence of pseudorandom generators $G : [q]^n \to \{0,1\}^m$ for some $\mathsf{poly}(n)$-size alphabet q, each of whose output bits depend on at most two in put alphabet symbols, and which achieve sufficiently large stretch.

We show polynomial-time attacks against such generators, invalidating the security of the IO and FE candidates. Our attack uses tools from the literature on two-source extractors (Chor and Goldreich, SICOMP 1988) and efficient refutation of random 2-XOR instances (Charikar and Wirth, FOCS 2004).

*Supported by an Akamai Presidential Fellowship and the grants of the second author.

†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 and W911NF-15-C-0236.

Contents

1 Introduction

There has been much recent progress on constructing indistinguishability obfuscation (IO) schemes [BGI+01, GR07] starting from the work of Garg, Gentry, Halevi, Raykova, Sahai and Waters [GGH+16]. Most recently, Lin [Lin16a] and others [LV16, Lin16b, AS16, LT17] showed a pathway to constructing IO schemes using two ingredients: multilinear maps of constant degree and pseudorandom generators of constant locality. In particular, Lin and Tessaro [LT17] construct an IO candidate from standard assumptions on *bilinear maps* and non-standard assumptions on "Goldreich-like" pseudorandom generators [Gol00] with "blockwise" locality 2.

This is a remarkable development: until recently, we had IO candidates based on constant degree (most recently, degree 5) multilinear maps and constant locality (most recently, locality 5) PRGs. We did not have any candidates for the degree 5 multilinear maps that satisfied the required assumptions (namely, a version of the decisional Diffie-Hellman assumption); however, we did have candidates for locality 5 PRGs that are known to resist a large class of attacks [OW14, AL16]. The Lin-Tessaro result dramatically changed the landscape by shifting the burden of existence from degree 5 multilinear maps to pseudorandom generators with (so-called) blockwise locality 2 and polynomial stretch. In other words, we have candidates for degree 2 multilinear maps (also known as bilinear maps) [BF03, Jou02, Jou00]; however, there are no locality 2 PRGs, and the security of blockwise locality 2 PRGs is highly questionable. (For the formal definitions of all these technical terms, see below and Section 2.)

In this work, we show a polynomial-time attack against the pseudorandom generators required for the Lin-Tessaro construction. As such, this constitutes a break of the Lin-Tessaro IO (as well as functional encryption) constructions that use bilinear maps.

We remark that our attacks do not apply to the Lin-Tessaro IO construction based on 3-linear maps. This leaves us in a curious state of affairs regarding constructions of IO from multilinear maps.

- There is a construction [LT17] of IO from trilinear maps (whose existence is questionable) and "blockwise 3-local PRGs" (for which we have plausible candidates); and
- There is a construction [LT17] of IO from bilinear maps (for which we have candidates that have been around for almost two decades) and "blockwise 2-local PRGs" (which are broken in this work).

Since cryptographically secure trilinear maps have so far eluded us, it is not surprising that the difficulty of achieving IO arises from the gap between bilinear and trilinear maps. However, we find it quite surprising that this transition appears to be related to the pseudorandomness of 2-local functions and 3-local functions (over a large alphabet, no less)!

Goldreich's PRGs with Blockwise 2-Local Predicates. We start by describing the object we attack. Let P be a predicate from Σ^2 to $\{0,1\}$, for some polynomial size alphabet $|\Sigma| = q = \mathsf{poly}(n)$. Let H be a (directed) graph with n vertices and m edges; we will refer to H as the constraint graph. The pseudorandom generator $G_{H,P} : \Sigma^n \to \{0,1\}^m$ is defined in the following way. Let $e = (i,j)$ be a directed edge in G. Then, the e^{th} bit of the output of the generator is computed as $P(x_i, x_j)$. We call this an (n,m,q)-*Goldreich-like* pseudorandom generator since it uses predicates over a large alphabet.

This construction can also be thought of as a "blockwise local" pseudorandom generator, a terminology that Lin and Tessaro introduce and use [LT17]. In an (L, w)-block-wise PRG, the nw-bit input is divided into blocks of size w bits each, and each output bit of the PRG can depend on at most L blocks. It is easy to see that a Goldreich PRG as defined above with alphabet size q is a $(2, \lceil \log q \rceil)$-block-wise PRG. In fact, Lin and Tessaro's definition of block-wise PRGs is more general in that it allowed each output bit to be computed using a different (publicly known) predicate. However, their candidate PRG used the same predicate to compute all the output bits.

With this terminology, we are ready to state our main result.

Theorem 1.1. *There is a poly(n, q) time algorithm $\mathcal{D}$ which, for any $m \geq \tilde{\Omega}(q \cdot n)$, any predicate $P : [q]^2 \to \{0, 1\}$, and any graph H with n vertices and m edges, distinguishes between a uniformly random string $z \leftarrow U_m$ and a random output $z \leftarrow G_{H,P}(U_{n,q})$ (with a constant distinguishing advantage).*

The Lin-Tessaro Theorem and Connection to Goldreich-like PRGs. Lin and Tessaro [LT17], building on earlier work [BV15, AJ15, Lin16a, LV16, Lin16b, AS16] showed an IO candidate based on the hardness of standard assumptions on bilinear maps and the existence of a Goldreich-like PRG with blockwise locality 2 and *sufficiently large* stretch. That is, they show:

> Under standard assumptions on bilinear maps and the existence of a subexponentially secure (n, m, q)-Goldreich-like PRG with $q = \mathsf{poly}(n)$ and $m = (nq^3)^{1+\epsilon}$ for some constant $\epsilon > 0$, there is an IO scheme. Assuming the existence of such a generator with quasipolynomial security, there is a compact FE scheme.

In a nutshell, they utilize the reductions of Ananth and Jain [AJ15] and Bitansky and Vaikuntanathan [BV15] who show how to construct an IO scheme from any sub-exponentially secure *compact* FE scheme. By compact FE, roughly speaking, we mean a functional encryption scheme for functions of large *output* size k with ciphertexts of size $k^{1-\epsilon}$ for some absolute constant $\epsilon > 0$. Such ciphertexts simply do not have enough space to hold the function output, so they, in a sense, have to do non-trivial computation as part of the FE decryption process. Since IO is the ultimate truth-table compression algorithm, the moral bottomline of [AJ15, BV15], formalized in [LPST16], is that "any compression implies the ultimate compression". On the other hand, non-compact FE schemes can be constructed essentially from any public-key encryption scheme [SS10, GVW12].

Thus, [LT17] construct a compact functional encryption scheme using their ingredients. Using their bootstrapping theorem, it turns out to be sufficient to construct an FE scheme that encrypts the seed of a PRG (which they instantiate with a Goldreich-like PRG as defined above) and whose functional key corresponds to the computation of the PRG itself (plus some). In a high level, their encryption algorithm takes as input the seed $\mathbf{x} = x_1 x_2 \dots x_n \in [q]^n$, pre-computes all possible monomials on the bits of each alphabet symbol $x_i \in [q]$ $(i = 1, \dots, n)$, of which there are roughly q, and includes it in the ciphertext. Computing the PRG output, then, can be written as a degree-2 computation which can be performed using a bilinear map (leveraging on an earlier result of Lin [Lin16b]). Thus, the number of bits being encrypted is $n \cdot q$. To achieve sublinear compactness which, by the above discussion, is necessary to apply the FE-to-IO transformations, they need the output length of the PRG m to be large enough, namely $m = \Omega((nq)^{1+\epsilon})$ for some constant $\epsilon > 0$. In fact, since they need to support computations that are a bit more complex than simply computing the PRG, they need the stretch to be $\Omega((nq^3)^{1+\epsilon})$.

Stretch	Worst-case vs. Random Predicate	Worst-case vs. Random Graph	Different vs. Same Predicate Per Output	Reference
$m = \tilde{\Omega}(q \cdot n)$	Worst-case	Random	Same	This Work [LV17], as originally posted
$m = \tilde{\Omega}(q \cdot n)$	Random	Random	Same	[BBKK17]
$m = \tilde{\Omega}(q^2 \cdot n)$	Worst-case	Worst-case	Different	[BBKK17]
$m = \tilde{\Omega}(q \cdot n)$	Worst-case	Worst-case	Same	This Work
$m = \tilde{\Omega}(q \cdot n)$	Worst-case	Worst-case	Different	Open

Figure 1: The State of the Art on Attacks against Blockwise 2-local PRGs.

Our main theorem (Theorem 1.1) now implies that a natural class of candidates for such PRGs, proposed and analyzed in [LT17], can be broken in polynomial-time. In fact, we show something stronger: even a potential extension of the Lin-Tessaro theorem that requires only blockwise 2-local PRGs with minimal expansion, namely $m = \tilde{\Omega}(nq)$, can be broken using our attack.

Comparison to *[BBKK17].* The presentation of this work has changed significantly since the original preprint [LV17]. We originally proved the following weaker version of Theorem 1.1; see also the first line of Figure 1.

Theorem 1.2. *There is a poly(n, q) time algorithm $\mathcal{D}$ which, for any $m \geq \tilde{\Omega}(q \cdot n)$, any predicate $P : [q]^2 \to \{0, 1\}$, and a $(1 - o(1))$ fraction of graphs H with n vertices and m edges, distinguishes between a uniformly random string $z \leftarrow U_m$ and a random output $z \leftarrow G_{H,P}(U_{n,q})$ (with a constant distinguishing advantage).*

In a concurrent work, Barak, Brakerski, Komargodski and Kothari [BBKK17] showed a completely different attack on a blockwise 2-local PRG with different parameter settings. Barak et al. show how to attack blockwise 2-local PRGs for *worst-case graphs* and a *worst-case collection of possibly different predicates* for each output bit. However, they need to assume that the PRG had a larger stretch, namely that $m = \tilde{\Omega}(q^2 \cdot n)$. They also achieve a threshold of $m = \tilde{\Omega}(q \cdot n)$ for the restricted case of *random* graphs and *random, single* predicate. See the second and third line of Figure 1.

Our main theorem draws inspiration from [BBKK17] and applies our main technique that we refer to as *alphabet reduction* in a different way than we originally conceived. See the fourth line of Figure 1.

There is a gap between our main theorem, namely Theorem 1.1, and a complete break of the [LT17] candidate: Theorem 1.1 breaks blockwise 2-local PRGs in which the predicate computing each output bit is the same. This breaks Lin and Tessaro's concrete PRG candidate. However, their theorem can be instantiated with more general block-wise local PRGs where each output bit is computed using a different predicate, a setting that [BBKK17] break. We remark here that our techniques (to be described below) can also be used to break this multiple-predicate variant at the cost of the same worse distinguishing threshold, namely $m \geq \tilde{\Omega}(nq^2)$.

On the negative side, we provide evidence that our own technique is unlikely to achieve a better threshold than $m \geq \tilde{\Omega}(q^2 \cdot n)$ for worst-case graphs and worst-case multiple predicates; it would be very interesting to understand the limits of the techniques in [BBKK17].

The current state of attacks against blockwise 2-local PRGs is summarized in Figure 1. As one can see from the table, there is a very narrow set of possibilities that neither our attack nor [BBKK17] rule out just yet. Namely, we cannot rule out the possibility that (a) the Lin-Tessaro theorem could be improved to work with stretch $\Omega((nq)^{1+\epsilon})$; and (b) there is a PRG with such a stretch that necessarily has to employ a specially tailored graph and different predicates for each output bit. An exceedingly narrow window, indeed!

1.1 Outline of Our Attack

We start with a description of our original attack, namely the proof of Theorem 1.2, which exploited the well-known connection between our problem of distinguishing a Goldreich PRG output from a uniform string and problems studied in the setting of random constraint satisfaction (CSP). In particular, we utilized a result of Allen, O'Donnell, and Witmer [AOW15] who developed a polynomial-time algorithm for a problem related to ours, namely that of *refutation of random CSPs*.

Any graph H with n nodes and m edges, any predicate P, and any string $z \in \{0,1\}^m$ together define an instance $\mathcal{I}$ of the following constraint satisfaction problem with predicates P and $\neg P$.

$$\begin{aligned} P(x_i, x_j) &= 1 \qquad \text{for every } e = (i,j) \text{ where } z_e = 1 \\ \neg P(x_i, x_j) &= 1 \qquad \text{for every } e = (i,j) \text{ where } z_e = 0 \end{aligned}$$

The task of breaking the PRG $G_{H,P}$ can be thought of as distinguishing CSP instances $\mathcal{I}$ in which the negations of P are chosen uniformly at random from instances $\mathcal{I}$ in which the negations of P are determined by a random planted solution $x \in [q]^n$.

Allen, O'Donnell, and Witmer [AOW15] developed a polynomial time algorithm for a different problem, namely that of *random CSP refutation*. In their setting (specialized to 2-CSPs), a random instance $\mathcal{I}$ is generated by choosing a random graph H along with *random negation patterns* $(a_e, b_e) \in \mathbb{Z}_q^2$ for each edge $e = (i,j) \in E(H)$, and including constraints

$$P(x_i + a_e, x_j + b_e) = 1$$

Their algorithm can certify that $\mathrm{Opt}(\mathcal{I})$, the largest fraction of constraints satisfied by any input, is less than 1 provided at least $\tilde{\Omega}(n \cdot \mathrm{poly}(q))$ constraints (for an unspecified polynomial poly). Namely, their algorithm outputs 1 with probability $1-o(1)$, but never outputs 1 if $\mathcal{I}$ is satisfiable. Clearly, this suffices to distinguish satisfiable instances $\mathcal{I}$ from uniformly random instances.[1] In fact, they achieve a much stronger property called *strong refutation* which will turn out to be crucial for us: given $\tilde{\Omega}(\frac{n}{\epsilon^2} \cdot \mathrm{poly}(q))$ constraints, their algorithm outputs 1 with probability $1 - o(1)$, but never outputs 1 if $\mathcal{I}$ is "somewhat close" to being satisfiable, that is, if $\mathrm{Opt}(\mathcal{I}) \geq 1/2 + \epsilon$ (when P is balanced). Finally, we note that their result only holds over random graphs H, but analogous results in the so-called *semi-random setting*, in which the graph H is worst-case but negation patterns are still random, have been shown in, e.g., [Wit17].

The most glaring difference between our setting and that of random CSP refutation [AOW15] is that our CSP instance has an "outer negation pattern" (randomly negating the predicate P) while theirs have an "inner negation pattern" as described above. However, it turns out that a

[1] We note that refutation seems to give us a significantly stronger guarantee than distinguishing. An analogous "refutation algorithm" in our PRG setting would be able to distinguish a random string $z \leftarrow \{0,1\}^m$ from $z \leftarrow G_{H,P}(x)$ for *any* distribution of the input x, not just the uniformly random one.

refutation algorithm for the random CSP model of [AOW15] can nevertheless be turned into a distinguishing algorithm, but at a cost; the resulting algorithm requires $m \geq \tilde{\Omega}(n \cdot \mathrm{poly}(q))$ for some large polynomial (roughly q^2 times the unspecified polynomial in [AOW15]).

Such a result is already nontrivial in the PRG setting, although it is far from the $m \geq \tilde{\Omega}(q \cdot n)$ threshold that we would like to achieve. This is the major challenge that this paper overcomes: how can we reduce this potentially large $\mathsf{poly}(q)$-dependence to an explicit, small $\mathsf{poly}(q)$-dependence?

Our main idea called alphabet reduction now comes to the rescue. Alphabet reduction is a way to convert our CSP on an alphabet of size q to a related CSP on a *new alphabet whose size is an absolute constant independent of* q. If the original CSP is random, so is the new CSP. If the original CSP is satisfiable, the new one is "somewhat close to being satisfiable", that is, there is an assignment that satisfies at least $1/2+\Omega(1/\sqrt{q})$ of its clauses. We then leverage the "strong refutation" property of the algorithm in [AOW15] to break the pseudorandomness of $G_{H,P}$ by certifying that a random CSP with *constant-sized predicate* Q is not $1/2 + \Omega(1/\sqrt{q})$-satisfiable, which can be done using the algorithm of [AOW15] with only $\tilde{\Omega}(n \cdot q'/\epsilon^2) = \tilde{\Omega}(n \cdot q)$ clauses, since $q' = O(1)$ and $\epsilon = \Omega(1/\sqrt{q})$. In other words, we traded a dependence on q in the number of required clauses for a dependence on q in the error parameter $\epsilon = \Omega(1/\sqrt{q})$; since the required number of clauses is proportional to $1/\epsilon^2$, this reduces the overall dependence on q to linear.

We achieve alphabet reduction by showing that any predicate $P : [q]^2 \to \{0,1\}$ is $(1/2 + \Omega(1/\sqrt{q}))$-correlated to another predicate $P' : [q]^2 \to \{0,1\}$ which "depends on only one bit of each input". This uses, and refines, a classical lower bound due to Chor and Goldreich [CG88] on 2-source extractors.

If our alphabet reduction produced a CSP instance whose alphabet size was some large constant, then this would be the end of the story. However, we can actually reduce to the *binary alphabet*. In the binary alphabet setting, it turns out that we can use the old MAX-2-XOR approximation algorithm of Charikar and Wirth [CW04] which achieves the following guarantee: for stretch $m = \tilde{\Omega}(\frac{n}{\epsilon^2})$, it can distinguish between a random string $z \leftarrow U_m$ and *any string* $z \in \{0,1\}^m$ which is within $\frac{1}{2} - \epsilon$ (fractional) Hamming distance of the image $G(\{0,1\}^n)$ of the PRG.[2] This allows us to obtain a much simpler algorithm (making a single black box call to the [CW04] algorithm instead of the [AOW15] algorithm) achieving the same $m = \tilde{\Omega}(n \cdot q)$ threshold, even for worst-case graphs.

Organization of the Paper. We start with some basic preliminaries in Section 2. Our alphabet reduction technique is presented in Section 3, and our attack which combines alphabet reduction with the 2-XOR algorithm of [CW04] is presented in Section 4.

2 Preliminaries

Notation. We let U_n denote the uniform distribution on $\{0,1\}^n$. Additionally, we let $U_{n,q}$ denote the uniform distribution on the set $[q]^n$. Let $\mathsf{negl}(n) : \mathbb{N} \to \mathbb{R}$ denote a 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 $\mathsf{negl}(n) < 1/p(n)$ for all $n > n_p$.

[2] The problem in the PRG setting that Charikar-Wirth solves is called the *image refutation* problem for G.

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 poly(n), and (2) for any probabilistic polynomial time adversary $A : \{0,1\}^m \to \{0,1\}$, there is a negligible function negl such that

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

We say that a PRG $G : \{0,1\}^n \to \{0,1\}^m$ has *stretch* $m - n = m(n) - n$. 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, we define the *locality* of G to be the maximum number of input bits on which any output bit of G depends.

2.2 Goldreich's Candidate (Blockwise) Local PRG

Goldreich's candidate pseudorandom generator, first introduced in [Gol00] (then as a candidate one-way function), can be instantiated with any k-ary predicate $P : [q]^k \to \{0,1\}$ and any k-uniform (directed) hypergraph H on n vertices and m hyperedges. (To the best of our knowledge, the generalization to predicates P that take symbols from a larger alphabet was first considered by Lin and Tessaro under the name of "block-wise local" PRGs). 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.

Definition 1. Given a predicate P and hypergraph H, Goldreich's PRG is the function from $[q]^n$ 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 function is often instantiated with a *uniformly random* choice of hypergraph H; in this setting, we say that "Goldreich's function 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 probability $1 - o(1)$. Often, instead of proving hardness results for random hypergraphs it suffices to use hypergraphs with "good expansion" for varying definitions of expansion [AL16, OW14, ABR12]. For a more in-depth survey and discussion of Goldreich's PRG, see [App16].

For the rest of the paper, we specialize to the case of $k = 2$, that is, blockwise 2-local PRGs. Ultimately, the attacks on $G_{H,P}$ that we describe in this paper hold for *all* graphs H, rather than just random graphs.

Finally, we note that one can analogously define $G_{H,\vec{P}}$ for a collection of m predicates $P_1, ..., P_m$ (in which the ith output bit of $G_{H,\vec{P}}$ is obtained using P_i).

3 Alphabet Reduction

Our main result relies on a technique that we call *alphabet reduction* which reduces the problem of distinguishing the Goldreich PRG that uses a predicate $P : \Sigma^2 \to \{0,1\}$ to that of distinguishing the PRG that uses a different predicate $Q : \Sigma'^2 \to \{0,1\}$ that acts on a smaller alphabet Σ'. In this section, we show the existence of such a suitable predicate Q (for every predicate P) and in the next, we use it to break the PRG. We start with the definition of alphabet reduction.

Definition 2 ($(\Sigma, \Sigma', \delta)$-Alphabet Reduction)**.** Let $P : \Sigma^2 \to \{0,1\}$ be a balanced predicate in two variables. A $(\Sigma, \Sigma', \delta)$-alphabet reduction for P is defined to be a tuple (Q, f, g) where $Q : \Sigma'^2 \to \{0,1\}$ is a balanced predicate defined on an alphabet Σ', and f and g are (exactly) $\frac{|\Sigma|}{|\Sigma'|}$-to-one maps from Σ to Σ', and

$$\mathop{\mathbf{E}}_{(x,y)\xleftarrow{\$}\Sigma^2} [P(x,y) \oplus Q(f(x), g(y))] < \delta.$$

In other words, $P(x,y)$ is nontrivially correlated to the *decomposed predicate* $Q(f(x), g(y))$. We use the shorthand "(q', δ)-alphabet reduction" when $|\Sigma'| = q'$.

Note that if $P(x,y)$ is perfectly correlated to $P'(x,y) := Q(f(x), g(y))$, then the expectation defined above is 0, and if they are perfectly uncorrelated, it is $1/2$. In words, this definition asks for a way to approximately compute P by first compressing the two inputs x and y independently, and then computing a different predicate Q on the compressed inputs.

In this section, we prove a feasibility result for alphabet reduction: namely, that any predicate $P : \Sigma^2 \to \{0,1\}$ has a $(\Sigma, \Sigma', 1/2 - \Omega(1/\sqrt{|\Sigma|})$-alphabet reduction where $\Sigma' = \{0,1\}$ is an alphabet of size two. In other words $Q(f(x), g(y))$ is mildly, but non-trivially, correlated to P. The predicate Q as well as the compression functions f and g are efficently computable given the truth table of P. Our result is a refinement of a lower bound on the possible error of two-source extractors, due to Chor and Goldreich [CG88].

Theorem 3.1. *Suppose that $P : \Sigma^2 \to \{0,1\}$ is a balanced predicate and $|\Sigma|$ is divisible by 2. Then, there exists a $(\Sigma, \Sigma', 1/2 - c/\sqrt{|\Sigma|})$-alphabet reduction (Q, f, g) for P, for some universal constant c. Moreover, given P we can find such a triple (Q, f, g) in (expected) poly$(|\Sigma|)$ time.*

Proof. Throughout this proof, we will equate Σ with the set $[q]$ (so that $|\Sigma| = q$) and Σ' with the set $\{0,1\}$ (so that $|\Sigma'| = 2$). Also, for ease of exposition, we consider P taking values in $\{\pm 1\}$ instead of $\{0,1\}$.

Given $P : [q]^2 \to \{\pm 1\}$, consider P as a ± 1-valued $q \times q$ matrix. At a high level, the proof goes as follows: we first find a $\frac{q}{2} \times \frac{q}{2}$ submatrix of P with substantially more $+1$s than -1s in it (or vice-versa). Such a submatrix is not hard to construct: picking a random collection T of $\frac{q}{2}$ columns and then choosing the collection S of $\frac{q}{2}$ rows optimizing the number of $+1$s (or -1s) in the $S \times T$ submatrix suffices. Then, we pick f (a function of the q rows) and g (a function of the q columns) to be indicator functions for S and T respectively; there then turns out to be a choice of function $Q : \{0,1\} \times \{0,1\} \to \{\pm 1\}$ (in particular, with $Q(1,1)$ set to be the majority value of P in the submatrix $S \times T$) such that (Q, f, g), with outputs transformed back to $\{0,1\}$, is a valid alphabet reduction for P.

We now proceed with the full proof. For each $x \in [q]$ and subset $T \subset [q]$, define

$$B(x, T) = \left| \sum_{y \in T} P(x, y) \right|,$$

that is, the absolute value of the T-partial row sum of row x. In [CG88] (Lemma 2), Chor and Goldreich show that if we choose T to be a uniformly random subset of $\frac{q}{2}$ columns, then for every x,

$$\Pr_{T \subset [q], |T| = \frac{q}{2}} \left[B(x, T) \geq \sqrt{\frac{q}{2}} \right] \geq \frac{1}{8}.$$

Therefore, we have that

$$\mathop{\mathbf{E}}_{T\subset[q],|T|=\frac{q}{2}}\left[\frac{1}{q}\cdot\#\left\{x\in[q]:B(x,T)\geq\sqrt{\frac{q}{2}}\right\}\right]\geq\frac{1}{8}.$$

Since the random variable $\frac{1}{q}\cdot\#\left\{x\in[q]:B(x,T)\geq\sqrt{\frac{q}{2}}\right\}$ takes values in the interval $[0,1]$ and has expectation at least $\frac{1}{8}$, we conclude by Markov's inequality that

$$\Pr_{T\subset[q],|T|=\frac{q}{2}}\left[\frac{1}{q}\cdot\#\left\{x\in[q]:B(x,T)\geq\sqrt{\frac{q}{2}}\right\}\leq\frac{1}{16}\right]\leq\frac{14}{15},$$

so that with probability $\geq\frac{1}{15}$ over the choice of T, there will be at least $\frac{q}{16}$ rows $x\in[q]$ such that $B(x,T)\geq\sqrt{\frac{q}{2}}$. Fixing any such set T, we then have that

$$\sum_{x\in[q]}B(x,T)\geq\frac{q\sqrt{q}}{16\sqrt{2}}.$$

Now, let $S\subset[q]$ be the set of $\frac{q}{2}$ rows $x_1,...,x_{\frac{q}{2}}$ with the largest values of $\tilde{B}(x,T):=\sum_{y\in T}P(x,y)$. We claim that

$$\left|\sum_{x\in S}\tilde{B}(x,T)\right|+\left|\sum_{x\notin S}\tilde{B}(x,T)\right|\geq\frac{q\sqrt{q}}{48\sqrt{2}},$$

that is, we claim that a significant fraction of the $\tilde{B}(x,T)$ terms do not cancel with each other when we sum over $x\in S$ and $x\notin S$ separately. To see this, let

$$C_1=\sum_{x:\tilde{B}(x,T)\geq0}B(x,T)$$

and

$$C_2=\sum_{x:\tilde{B}(x,T)<0}B(x,T)$$

so that $C_1+C_2=\sum_{x\in[q]}B(x,T)$. We note that without loss of generality, we have that $\tilde{B}(x,T)\geq0$ for all $x\in S$, so that

$$\left|\sum_{x\in S}\tilde{B}(x,T)\right|+\left|\sum_{x\notin S}\tilde{B}(x,T)\right|=\sum_{x\in S}B(x,T)+\max(\sum_{x\notin S}\tilde{B}(x,T),-\sum_{x\notin S}\tilde{B}(x,T))$$

$$\geq\max(C_1-C_2,C_2)\geq\frac{1}{3}(C_1-C_2)+\frac{2}{3}C_2\geq\frac{q\sqrt{q}}{48\sqrt{2}}.$$

as desired. Thus, either the submatrix $S\times T$ or $([q]-S)\times T$ has the intermediate property we were looking for.

Finally, we can construct Q,f, and g as follows: let $S_0=S,S_1=[q]-S$, $T_0=T$, $T_1=[q]-T$, and for $i,j\in\{0,1\}$ we define

$$E_{ij}=\frac{1}{q^2}\sum_{(x,y)\in S_i\times T_j}P(x,y).$$

For $i, j \in \{0,1\}$, define $Q(i,j) = 1$ if E_{ij} is one of the two largest elements of the (multi)set $\{E_{ij}, i, j \in \{0,1\}\}$ (and $Q(i,j) = -1$ otherwise). Moreover, we define $f(x) = i$ if and only if $x \in S_i$, and we define $g(y) = j$ if and only if $y \in T_j$. Intuitively, for $(x,y) \in S_i \times T_j$ we want to set $Q(f(x), g(y))$ to be the majority value of $P(x', y')$ for $(x', y') \in S_i \times T_j$, but to make Q a balanced predicate we may have to disagree with this majority function on some inputs.

By essentially the same argument about cancellation as before, we will show that $P(x,y)$ is $\frac{1}{2} + \Omega(\frac{1}{\sqrt{q}})$-correlated to $Q(f(x), g(y))$. That is, we show that

$$\mathop{\mathbf{E}}_{(x,y) \leftarrow U_{2,q}} [P(x,y)Q(f(x),g(y))] \geq \frac{1}{2}\left(|E_{00}| + |E_{01}| + |E_{10}| + |E_{11}|\right) = \Omega(\frac{1}{\sqrt{q}}).$$

To see this, re-order the four numbers E_{ij} into $E_1 \leq E_2 \leq E_3 \leq E_4$; we know that $E_1 + E_2 + E_3 + E_4 = 0$ since $P(x,y)$ is balanced. If exactly two of these four numbers are negative, then the expected value above is exactly equal to $|E_1| + |E_2| + |E_3| + |E_4|$, so we are done. On the other hand, it may be that three of $\{E_1, E_2, E_3, E_4\}$ have the same sign; suppose without loss of generality that $E_3 \leq 0$. Then, we see that

$$\mathop{\mathbf{E}}_{(x,y) \leftarrow U_{2,q}} [P(x,y)Q(f(x),g(y))] = |E_1| + |E_2| - |E_3| + |E_4|$$

$$\geq |E_4| = \frac{1}{2}\left(|E_1| + |E_2| + |E_3| + |E_4|\right),$$

completing the existence proof. Moreover, our existence proof above is constructive: to find a valid triple (Q, f, g), we repeatedly choose $T \subset [q]$ of size $\frac{q}{2}$ uniformly at random, check if $\sum_{x \in [q]} B(x,T) = \Omega(q\sqrt{q})$ (for suitably chosen constant c), and proceed to construct S, Q, f, and g as described. In expectation only a constant number of sets T will be selected before S, Q, f, and g are successfully constructed, so we are done. □ □

3.1 Limits of Alphabet Reduction

Alphabet reduction is one of the two main ingredients to our distinguishing algorithm. In order to obtain distinguishers for an even larger class of PRGs, namely, instantiations of Goldreich's PRG in which m possibly different predicates $P^{(1)}, P^{(2)}, ..., P^{(m)}$ are used instead of a repeated single predicate, one can analogously define an "average case alphabet reduction" for an m-tuple of predicates $\vec{P}$.

Definition 3 (Average Case $(\Sigma, \Sigma', \delta)$-Alphabet Reduction)**.** Let $P^{(1)}, P^{(2)}, \ldots, P^{(m)} : \Sigma^2 \to \{0,1\}$ be a collection of balanced predicates in two variables. A $(\Sigma, \Sigma', \delta)$-*average case alphabet reduction* for $\vec{P}$ is defined to be a tuple $(\vec{Q}, f, g)$ such that each $Q^{(i)} : \Sigma'^2 \to \{0,1\}$ is a balanced predicate defined on an alphabet of size q', f and g are (exactly) $\frac{q}{q'}$-to-one maps from $\Sigma \to \Sigma'$, and

$$\mathop{\mathbf{E}}_{(x,y) \xleftarrow{\$} \Sigma^2, i \xleftarrow{\$} [m]} [P^{(i)}(x,y) \oplus Q^{(i)}(f(x), g(y))] < \delta.$$

In other words, $P^{(i)}(x,y)$ is nontrivially correlated to $Q^{(i)}(f(x), g(y))$ on average over the choice of i. We use the shorthand "(q', δ)-average case alphabet reduction" for $\vec{P}$ when $|\Sigma'| = q'$.

Note that we require the same alphabet reduction functions f and g to work for all the predicates $P^{(i)}$ *simultaneously.*

It turns out that average case alphabet reduction is significantly more difficult to achieve than alphabet reduction. In general, one cannot find a constant size average case alphabet reduction with $\delta < \frac{1}{2} - \tilde{O}(\frac{1}{q})$.

In particular, when $\vec{P}$ is a good 3-*source extractor* $\vec{P} : [q] \times [q] \times [m] \to \{0,1\}$, no such alphabet reduction can be done. Our impossibility result for alphabet reduction boils down to a (slightly modified) folklore construction of 3-source extractors, which we include for completeness.

Theorem 3.2. *Let $\vec{P} = (P_{ij}^{(k)})$ be a uniformly random ± 1-entry $q \times q \times m$ 3-tensor subject to the contraint that $P^{(k)}$ is balanced for every k, and suppose that $q \leq m$. Then, for any constant C, we have that with overwhelming probability, every $\frac{q}{C} \times \frac{q}{C} \times \frac{m}{C}$ subtensor $\vec{P}|_{S \times T \times U}$ of $\vec{P}$ has discrepancy $\left|\sum_{(i,j,k) \in S \times T \times U} \vec{P}_{ij}^{(k)}\right| = O(\frac{\log(mq)}{q})$.*

Corollary 3.3. *If $\vec{P}$ is a uniformly random collection of m balanaced predicates $P^{(i)} : [q]^2 \to \{0,1\}$, then for any constant C, there is no $\left(C, \frac{1}{2} - O(\frac{\log(mq)}{q})\right)$-average case alphabet reduction for $\vec{P}$ with overwhelming probability.*

Proof. First, consider any fixed subtensor $\vec{P}|_{S \times T \times U}$ of size $\frac{q}{C} \times \frac{q}{C} \times \frac{m}{C}$, and suppose that $(P_{ij}^{(k)})$ is a uniformly random tensor (not constrained to be balanced). Then, $\vec{P}|_{S \times T \times U}$ is a uniformly random ± 1-tensor whose discrepancy is governed by the Chernoff bound:

$$\Pr\left[\left|\sum_{i \in S, j \in T, k \in U} P_{ij}^{(k)}\right| \geq \epsilon\right] \leq 2 \cdot 2^{-\Omega(\frac{mq^2}{C^3}\epsilon^2)}.$$

The number of subtensors we are considering is $\binom{m}{\frac{m}{C}}\binom{q}{\frac{q}{C}}\binom{q}{\frac{q}{C}}$, and the probability that a random tensor $\vec{P}$ has the property that $P^{(k)}$ is balanced for all k is bounded by $(\Omega(\frac{1}{q}))^m$ (as the discrepancy of each $P^{(k)}$ follows the distribution Binomial$(q^2, \frac{1}{2})$). Thus, the probability that a random $\vec{P}$ satisfies this ϵ-discrepancy property after conditioning on balanced is bounded by

$$O(q)^m \binom{m}{\frac{m}{C}} \binom{q}{\frac{q}{C}}^2 \cdot 2^{-\Omega(\frac{mq^2}{C^3}\epsilon^2)}.$$

Choosing $\epsilon = O(\frac{C^{1.5}\log(mq)}{q})$ suffices to make this probability negligible, so we are done. □ □

As a result of Theorem 3.2, it is unlikely for alphabet reduction to be sufficient for breaking $G_{H,\vec{P}}$ with $m = \tilde{\Omega}(q \cdot n)$ output length, because the refutation algorithms with which we combine predicate reduction have a $\frac{1}{\epsilon^2}$ dependence in the required output length for ϵ-refutation (and this dependence is typical). Therefore, it is unlikely for average case alphabet reduction to lead to a distinguisher for $G_{H,\vec{P}}$ when the output length $m = |E(H)|$ is less than $q^2 n$.

However, we note for completeness' sake that Theorem 3.2 is tight up to log factors; that is, $\left(\frac{1}{2} - \Omega(\frac{1}{q})\right)$-average case alphabet reduction is possible. The construction is as follows: pick sets S, T uniformly at random (of size $\frac{|\Sigma|}{2}$), choose $f, g : \Sigma \to \{0,1\}$ to be indicator functions for S and T, as before, and for each $\ell \in [m]$ define $Q^{(\ell)}(i,j)$ to be 1 if and only if the average value $E_{i,j}^{(\ell)}$ is in the top two out of four $E_{\cdot,\cdot}^{(\ell)}$, as before. Using average-case alphabet reduction, one can distinguish multiple-predicate Goldreich PRGs $G_{H,\vec{P}}$ when $m \geq \tilde{\Omega}(q^2 \cdot n)$; we will elaborate on this in Section 4.2.

4 From Small Alphabet Refutation to Large Alphabet Distinguishing

We now describe how alphabet reduction is used to obtain distinguishing algorithms for the (single predicate) Goldreich PRG $G_{H,P}$; combining this section with Theorem 3.1 yields Theorem 1.1. The cleanest interpretation of our application of alphabet reduction uses the notion of an "image refutation algorithm" for a function $G : [q]^n \to \{0,1\}^m$, which was formally defined in [BBKK17]. Interpreted in this language, our result says that *any* image refutation algorithm for Goldreich's PRG can be converted into a distinguishing algorithm for Goldreich's PRG with a significantly improved dependence on the alphabet size. The new distinguishing threshold is a simple function of the quality of the alphabet reduction that was used and the refutation threshold for the image refutation algorithm.

Definition 4 (Image Refutation)**.** Let $G : [q]^n \to \{0,1\}^m$ be any function. An *image refutation algorithm* for G is an algorithm $\mathcal{A}$ which receives G and a string $z \in \{0,1\}^m$ as input, with the following properties:

1. (Soundness) If $z \in G([q]^n)$, then $\mathcal{A}(G,z) =$"fail".
2. (Completeness) If $z \leftarrow U_m$, then $\mathcal{A}(G,z) =$"z is not in the image of G" with probability $1 - o(1)$.

Furthermore, $\mathcal{A}$ is an $(\frac{1}{2} - \delta)$*-image refutation algorithm* for G if it has the following properties:

1. (Strong Soundness) If z has Hamming distance less than or equal to $(\frac{1}{2} - \delta)m$ from $G([q]^n)$, then $\mathcal{A}(G,z) =$"fail".
2. (Strong Completeness) If $z \leftarrow U_m$, then $\mathcal{A}(G,z) =$"z is $(\frac{1}{2} - \delta)$-far from the image of G" with probability $1 - o(1)$.

Given this definition, we are ready to state our reduction theorem.

Theorem 4.1. *Let $P : [q]^2 \to \{0,1\}$ be a predicate. Assume the existence of the following two ingredients:*

- *An efficiently computable $(q', \frac{1}{2} - \epsilon)$-alphabet reduction for P that produces a tuple (Q, f, g) where $Q : [q']^2 \to \{0,1\}$; and*
- *An image refutation algorithm $\mathcal{A}$ that runs in $\mathsf{poly}(n, m, q', \frac{1}{\delta})$ time and does $(\frac{1}{2} - \delta)$-image refutation for the function $G_{H,Q}$ for any predicate $Q : [q']^2 \to \{0,1\}$, any $\delta > 0$ and any graph H satisfying $m = |E(H)| \geq T(n, q', \delta)$ for some threshold function $T(\cdot)$.*

Then, there is a distingusher $\mathcal{D}$ that, for any graph H with $m = |E(H)| \geq T(2n, q', \epsilon - O(\sqrt{\frac{n}{m}}))$, runs in poly$(n, q, \frac{1}{\epsilon})$ time and distinguishes a random string $z \leftarrow U_m$ from a random output $z \leftarrow G_{H,P}(U_{n,q})$ of $G_{H,P}$.

In particular, since Theorem 3.1 efficiently produces a $(2, \frac{1}{2} - \Omega(\frac{1}{\sqrt{q}}))$-alphabet reduction for any balanced predicate P, Theorem 4.1 implies that any strong image refutation algorithm for Goldreich's PRG over the *binary* alphabet immediately yields a distinguishing algorithm for Goldreich's PRG over larger alphabets.

Image Refutation Algorithms for Goldreich's PRG. We originally combined an alphabet reduction (with $q' = O(1)$) with the random CSP refutation algorithm of [AOW15] in place of a PRG image refutation algorithm, which turned out to be sufficient to obtain a distinguisher for $G_{H,P}$ over random graphs for all $m = \tilde{\Omega}(q \cdot n)$.

However, with an alphabet reduction using $q' = 2$, the state of affairs is much simpler; indeed, the Charikar-Wirth algorithm [CW04] directly gives us a PRG image refutation algorithm which can then be used to obtain a distinguisher for worst case graphs and worst case single predicates for $m = \tilde{\Omega}(q \cdot n)$ (for a sufficiently large logarithmic factor). This is because Charikar-Wirth $(\frac{1}{2} - \epsilon)$-refutes random 2-XOR instances with $m = \tilde{\Omega}(\frac{n}{\epsilon^2})$ constraints, and strongly refuting Goldreich's PRG instantiated with a balanced predicate $Q : \{0,1\}^2 \to \{0,1\}$ is exactly the same as strongly refuting a random 2-XOR instance (or a random 1-XOR instance, which is even easier). In particular, a balanced predicate $Q : \{0,1\}^2 \to \{0,1\}$ is either $Q(x,y) = x$, $Q(x,y) = y$, $Q(x,y) = x \oplus y$, or a negation of the previous three examples. Thus, any Goldreich PRG $G_{H,Q}$ defines a random 2-XOR instance or a random 1-XOR instance, either of which can be efficiently (strongly) refuted.

In the *multiple predicate* case, a Goldreich PRG $G_{H,\vec{Q}}$ (still over the binary alphabet) defines both a random 2-XOR instance and a random 1-XOR instance. It is not hard to see that if m is sufficiently large, at least one of these two CSP instances will be above its refutation threshold, yielding the necessary strong image refutation algorithm for $G_{\tilde{H},\vec{Q}}$. We will use this stronger fact for Theorem 4.3.

Furthermore, we note that this theorem can still be useful in regimes where general alphabet reduction is impossible; it says that if a predicate $P : [q]^2 \to \{0,1\}$ happens to have an alphabet reduction, then $G_{H,P}$ may be less secure than one would expect for the given alphabet size q.

We now prove Theorem 4.1. The intuition is quite simple: given an alphabet reduction (Q, f, g) for P and an image $z = G_{H,P}(x)$ for a random x, one would expect that z is noticeably closer to the point $G_{H,P'}(x)$ for $P'(x,y) = Q(f(x), g(y))$. Indeed, this is true in expectation over x, and holds with high probability by a concentration argument. Therefore, a strong refutation algorithm for the predicate Q should be able to distinguish $G_{H,P}(x)$ from a random string.

4.1 Proof of Theorem 4.1

Fix any predicate $P : [q]^2 \to \{0,1\}$, efficiently computable (q', δ)-alphabet reduction (Q, f, g),[3] and graph H with n vertices and m edges. Let $G_{H,P} : [q]^n \to \{0,1\}^m$ be Goldreich's PRG instantiated with P and H. We want to construct a distinguisher $\mathcal{D}(H, P, z)$ which, given P, H, and a string $z \in \{0,1\}^m$ (where m is the number of edges in H), outputs a bit $b \in \{0,1\}$ such that $\mathop{\mathbf{E}}_{z \leftarrow U_m}[\mathcal{D}(P, H, z)]$ is noticeably different from $\mathop{\mathbf{E}}_{z \leftarrow G_{H,P}(U_n)}[\mathcal{D}(P, H, z)]$. Our distinguisher $\mathcal{D}$ is defined as follows.

1. Compute (Q, f, g) given P.
2. Let $\tilde{H}$ be the bipartite double-cover of H, i.e. a graph with vertex set $[n] \times \{0,1\}$ and edges from $(i,0)$ to $(j,1)$ for every $(i,j) \in E(H)$.
3. Call $\mathcal{A}(\tilde{H}, Q, \epsilon - 5\sqrt{\frac{n}{m}}, z)$.
4. Return 1 if and only if the call to $\mathcal{A}$ returns "z is $(\frac{1}{2} - \epsilon + 5\sqrt{\frac{n}{m}})$-far from the image of $G_{\tilde{H},Q}$".

[3] This alphabet reduction may be randomized; this does not affect the proof.

Note that by assumption on $\mathcal{A}$, for $z \leftarrow U_m$, $\mathcal{D}(P, H, z)$ will output 1 with probability $1 - o(1)$ as long as $m \geq T(2n, q', \epsilon - 5\sqrt{\frac{n}{m}})$. What remains is to analyze the "pseudorandom" case.

Lemma 4.2. *With constant probability over $\mathbf{x} \leftarrow U_{n,q}$, $z = G_{H,P}(\mathbf{x})$ will have Hamming distance at most $(\frac{1}{2} - \epsilon + 5\sqrt{\frac{n}{m}})m$ from $G_{\tilde{H},Q}(\tilde{\mathbf{x}})$, where $\tilde{\mathbf{x}} \in (\mathbb{Z}_2^n)^2$ is defined by $\tilde{x}_{i,0} = f(x_i)$ and $\tilde{x}_{i,1} = g(x_i)$.*

Since the call to $\mathcal{A}(\tilde{H}, Q, \epsilon - 5\sqrt{\frac{n}{m}}, z)$ must return "fail" whenever z has Hamming distance at most $(\frac{1}{2} - \epsilon + 5\sqrt{\frac{n}{m}})m$ from $G_{\tilde{H},Q}(\mathbb{Z}_{q'}^n)$ (again for $m \geq T(2n, q', \epsilon - 5\sqrt{\frac{n}{m}})$), Lemma 4.2 suffices to prove that

$$\mathop{\mathbf{E}}_{\mathbf{x} \leftarrow U_{n,q}} [\mathcal{D}(H, P, G_{H,P}(\mathbf{x}))] = 1 - \Omega(1).$$

Proof. Let $P'(x, y) = Q(f(x), g(y))$ so that $\Pr_{(x,y) \leftarrow U_{2,q}}[P(x, y) = P'(x, y)] = \alpha \geq \frac{1}{2} + \epsilon$, as guaranteed by the fact that (Q, f, g) is a $(\frac{1}{2} - \epsilon)$-alphabet reduction for P. We are interested in the quantity $d_H(z, G_{\tilde{H},Q}(\tilde{\mathbf{x}})) = d_H(z, G_{H,P'}(\mathbf{x}))$, where d_H denotes fractional Hamming distance. First, we note that in expectation over $\mathbf{x} \leftarrow U_{n,q}$,

$$\begin{aligned}
E &:= 1 - \mathop{\mathbf{E}}_{\mathbf{x} \leftarrow U_{n,q}} [d_H(G_{H,P}(\mathbf{x}), G_{\tilde{H},Q}(\tilde{\mathbf{x}}))] \\
&= \mathop{\mathbf{E}}_{\mathbf{x} \leftarrow U_{n,q}} \left[\Pr_{(i,j) \xleftarrow{\$} E(H)} \left[P'(x_i, x_j) = P(x_i, x_j) \right] \right] \\
&\geq \alpha - \frac{n}{m} \\
&\geq \frac{1}{2} + \epsilon - \frac{n}{m},
\end{aligned}$$

where the $\frac{n}{m}$ term comes from the fraction of edges in H which are self loops (we cannot say that $P(x_i, x_i)$ is necessarily correlated to $P'(x_i, x_i)$). Now, we compute the variance (over $\mathbf{x}$) of this quantity to be

$$\begin{aligned}
&\mathop{\mathrm{Var}}_{\mathbf{x} \leftarrow U_{n,q}} [1 - d_H(G_{H,P}(\mathbf{x}), G_{\tilde{H},Q}(\tilde{\mathbf{x}}))] \\
&= \mathop{\mathbf{E}}_{\mathbf{x} \leftarrow U_{n,q}} \left[\left(\Pr_{(i,j) \xleftarrow{\$} E(H)} \left[P'(x_i, x_j) = P(x_i, x_j) \right] \right)^2 \right] - E^2 \\
&= \mathop{\mathbf{E}}_{\mathbf{x} \leftarrow U_{n,q}} \left[\frac{1}{m^2} \sum_{\substack{(i,j) \in E(H) \\ (k,l) \in E(H)}} \chi\left(P'(x_i, x_j) = P(x_i, x_j)\right) \chi\left(P'(x_k, x_l) = P(x_k, x_l)\right) \right] - E^2 \\
&= \frac{1}{m^2} \sum_{\substack{(i,j) \in E(H) \\ (k,l) \in E(H)}} \Pr_{\mathbf{x} \leftarrow U_{n,q}} \left[P'(x_i, x_j) = P(x_i, x_j) \text{ and } P'(x_k, x_l) = P(x_k, x_l) \right] - E^2.
\end{aligned}$$

Note that if the edges $(i, j), (k, l) \in E(H)$ have no vertices in common, the events "$P'(x_i, x_j) = P(x_i, x_j)$" and "$P'(x_k, x_l) = P(x_k, x_l)$" are independent. This means that our variance is upper bounded by

$$\frac{1}{m^2} \sum_{\substack{(i,j)\in E(H)\\(k,l)\in E(H)}} \Pr_{\mathbf{x}}\left[P'(x_i,x_j) = P(x_i,x_j)\right] \Pr_{\mathbf{x}}\left[P'(x_k,x_l) = P(x_k,x_l)\right] + \frac{m_{\text{bad}}}{m^2} - E^2$$

$$= \frac{m_{\text{bad}}}{m^2},$$

where m_{bad} is defined to be the number of pairs of edges $((i,j),(k,l))$ which have a vertex in common. This is bounded by the quantity

$$m_{\text{bad}} \leq \sum_{i\in[n]} \deg_H(i)^2 \leq 2n \cdot \sum_{i\in[n]} \deg_H(i) = 4mn.$$

Therefore, we conclude that

$$\operatorname*{Var}_{\mathbf{x}\leftarrow U_{n,q}}\left[1 - d_H(G_{H,P}(\mathbf{x}), G_{\tilde{H},Q}(\tilde{\mathbf{x}}))\right] \leq \frac{4n}{m}.$$

By Chebyshev's inequality, this means that with constant probability over $\mathbf{x} \leftarrow U_{n,q}$, we have that

$$1 - d_H(G_{H,P}(\mathbf{x}), G_{H,\tilde{Q}}(\tilde{\mathbf{x}})) \geq \alpha - \frac{n}{m} - 4\sqrt{\frac{n}{m}} \geq \frac{1}{2} + \epsilon - 5\sqrt{\frac{n}{m}},$$

so that $d_H(G_{H,P}(\mathbf{x}), G_{H,\tilde{Q}}(\tilde{\mathbf{x}})) \leq \frac{1}{2} - \epsilon + 5\sqrt{\frac{n}{m}}$, completing the proof of the lemma. □ □

Lemma 4.2 tells us that for $m \geq T(2n, q', \epsilon - 5\sqrt{\frac{n}{m}})$, with constant probability over $\mathbf{x} \leftarrow G_{n,q}$ the call made to $\mathcal{A}$ will return "fail", and so

$$\mathop{\mathbf{E}}_{\mathbf{x}\leftarrow G_{n,q}}\left[\mathcal{D}(H,P,G_{H,P}(\mathbf{x}))\right] = 1 - \Omega(1).$$

Thus, we conclude that $\mathcal{D}$ achieves a constant distinguishing advantage between the "truly random z" case and the "pseudorandom z" case, completing the proof of Theorem 4.1.

4.2 Generalization of Theorem 4.1 to Multiple Predicates

We note that the proof of Theorem 4.1 does not fundamentally use the fact that the predicates used in Goldreich's PRG $G_{H,P}$ are identical. Indeed, the following result holds by the same argument.

Theorem 4.3. *Let $P^{(1)}, P^{(2)}, \ldots, P^{(m)} : [q]^2 \to \{0,1\}$ be a collection of m predicates. Assume the existence of the following two ingredients:*

- *An efficiently computable $(q', \frac{1}{2} - \epsilon)$-average case alphabet reduction for $\vec{P}$ that produces a tuple $(\vec{Q}, f, g)$ where each $Q^{(\ell)} : [q']^2 \to \{0,1\}$; and*
- *An image refutation algorithm $\mathcal{A}$ that runs in $\mathsf{poly}(n, m, q', \frac{1}{\delta})$ time and does $(\frac{1}{2} - \delta)$-image refutation for the function $G_{H,\vec{Q}}$ for any predicate collection $Q^{(\ell)} : [q']^2 \to \{0,1\}$, any $\delta > 0$ and any graph H satisfying $m = |E(H)| \geq T(n, q', \delta)$ for some threshold function $T(\cdot)$.*

Then, there is a distingusher $\mathcal{D}$ that, for any graph H with $m = |E(H)| \geq T(2n, q', \epsilon - O(\sqrt{\frac{n}{m}}))$, runs in poly$(n, q, \frac{1}{\epsilon})$ time and distinguishes a random string $z \leftarrow U_m$ from a random output $z \leftarrow G_{H,\vec{P}}(U_{n,q})$ of $G_{H,\vec{P}}$.

Theorem 4.3, combined with the Charikar-Wirth algorithm and an average-case alphabet reduction with correlation $\Omega(\frac{1}{q})$, gives us a distinguisher for multiple predicate Goldreich PRGs $G_{H,\vec{P}} : [q]^n \to \{0,1\}^m$ for all $m \geq \tilde{\Omega}(q^2 n)$.

Acknowledgements. We thank Gil Cohen, Dana Moshkovitz and Prasad Raghavendra for their quick responses to our oracle calls about two-source extractors and CSPs. We also thank our anonymous TCC reviewers for their helpful comments and suggestions.

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.

[AJ15] Prabhanjan Ananth and Abhishek Jain. Indistinguishability obfuscation from compact functional encryption. In Rosario Gennaro and Matthew Robshaw, editors, *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, volume 9215 of *Lecture Notes in Computer Science*, pages 308–326. Springer, 2015.

[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.

[AOW15] Sarah R. Allen, Ryan O'Donnell, and David Witmer. How to refute a random csp. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 689–708. IEEE, 2015.

[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.

[BBKK17] Boaz Barak, Zvika Brakerski, Ilan Komargodski, and Pravesh K. Kothari. Limits on low-degree pseudorandom generators (or: Sum-of-squares meets program obfuscation). 2017. http://eprint.iacr.org/2017/312, version 20170411:133059. Submitted 9 April, 2017.

[BF03] Dan Boneh and Matthew Franklin. Identity-based encryption from the weil pairing. *SIAM journal on computing*, 32(3):586–615, 2003.

[BGI+01] Boaz Barak, Oded Goldreich, Rusell Impagliazzo, Steven Rudich, Amit Sahai, Salil Vadhan, and Ke Yang. On the (im) possibility of obfuscating programs. In *Annual International Cryptology Conference*, pages 1–18. Springer, 2001.

[BV15] Nir Bitansky and Vinod Vaikuntanathan. Indistinguishability obfuscation from functional encryption. In Venkatesan Guruswami, editor, *IEEE 56th Annual Symposium*

on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015, pages 171–190. IEEE Computer Society, 2015.

[CG88] Benny Chor and Oded Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM Journal on Computing*, 17(2):230–261, 1988.

[CW04] Moses Charikar and Anthony Wirth. Maximizing quadratic programs: Extending grothendieck's inequality. In *Foundations of Computer Science, 2004. Proceedings. 45th Annual IEEE Symposium on*, pages 54–60. IEEE, 2004.

[GGH+16] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. *SIAM Journal on Computing*, 45(3):882–929, 2016.

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

[GR07] Shafi Goldwasser and Guy Rothblum. On best-possible obfuscation. In *Theory of Cryptography Conference*, pages 194–213. Springer, 2007.

[GVW12] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Functional encryption with bounded collusions via multi-party computation. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, volume 7417 of *Lecture Notes in Computer Science*, pages 162–179. Springer, 2012.

[Jou00] Antoine Joux. A one round protocol for tripartite diffie–hellman. In *International algorithmic number theory symposium*, pages 385–393. Springer, 2000.

[Jou02] Antoine Joux. The weil and tate pairings as building blocks for public key cryptosystems. In *ANTS*, volume 2369, pages 20–32. Springer, 2002.

[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.

[LPST16] Huijia Lin, Rafael Pass, Karn Seth, and Sidharth Telang. Indistinguishability obfuscation with non-trivial efficiency. In Chen-Mou Cheng, Kai-Min Chung, Giuseppe Persiano, and Bo-Yin Yang, editors, *Public-Key Cryptography - PKC 2016 - 19th IACR International Conference on Practice and Theory in Public-Key Cryptography, Taipei, Taiwan, March 6-9, 2016, Proceedings, Part II*, volume 9615 of *Lecture Notes in Computer Science*, pages 447–462. Springer, 2016.

[LT17] Huijia Lin and Stefano Tessaro. Indistinguishability obfuscation from bilinear maps and block-wise local prgs. *IACR Cryptology ePrint Archive*, 2017:250, 2017. Version 20170320:142653.

[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.

[LV17] Alex Lombardi and Vinod Vaikuntanathan. On the non-existence of blockwise 2-local prgs with applications to indistinguishability obfuscation. Cryptology ePrint Archive, Report 2017/301, 2017. http://eprint.iacr.org/2017/301, version 20170409:183008. Submitted 6 April, 2017.

[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.

[SS10] Amit Sahai and Hakan Seyalioglu. Worry-free encryption: functional encryption with public keys. In Ehab Al-Shaer, Angelos D. Keromytis, and Vitaly Shmatikov, editors, *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS 2010, Chicago, Illinois, USA, October 4-8, 2010*, pages 463–472. ACM, 2010.

[Wit17] David Witmer. *Refutation of random constraint satisfaction problems using the sum of squares proof system.* PhD thesis, Carnegie Mellon University, 2017.