The GGM PRF Construction is Optimal

Bar Alon* alonbar08@gmail.com

Itai Dinur† dinuri@bgu.ac.il

Muthuramakrishnan Venkitasubramaniam‡ mv783@georgetown.edu

November 6, 2026

Abstract

We study the complexity of fully black-box constructions of pseudorandom functions (PRFs) from pseudorandom generators (PRGs). The seminal work of Goldreich, Goldwasser, and Micali (JACM 1986), combined with Levin's domain-extension technique, yields PRFs that make a superlogarithmic number of adaptive calls to a length-doubling PRG. A long-standing open problem is whether fewer calls suffice for fully black-box constructions. Despite progress for restricted classes of constructions and reductions, previous results have not been able to rule out even a single-call construction in full generality. A related question asks how many calls are needed to amplify a PRG's stretch from r bits to $\rho > r$ bits.

Our main result establishes the asymptotic optimality of the GGM construction with domain compression, both in the number of calls and in adaptivity, for length-doubling PRGs. More generally, for a PRG with seed length n and stretch $r(n)$, we rule out fully black-box PRF constructions with $d(n)$ adaptive rounds whenever $d(n)\log(2 + r(n)/n) = O(\log n)$, even allowing polynomially many calls per round. Our bounds hold for one-bit weak PRFs on domains of superpolynomial size. In particular, they exclude all polynomial-query nonadaptive constructions from PRGs of arbitrary polynomial stretch and require superlogarithmically many rounds when the underlying PRG doubles its input length.

We further show impossibility results for fully black-box non-adaptive PRG stretch amplification. First, we show that a non-adaptive construction of a ρ-bit stretch PRG from an r-bit stretch PRG requires $\Omega(\frac{\rho}{r+\log n})$ calls. Second, we show that it is impossible to construct a length-doubling PRG from an $o(n)$-bit stretch PRG with non-adaptive calls.

Our techniques combine and extend two prior lines of work. First, we develop a sensitivity framework that measures the effect of changing a uniformly chosen oracle entry and transfers this stability through black-box reductions, extending the sensitivity arguments of Bitansky et al. (CRYPTO 2017) and Bitansky and Degwekar (TCC 2019). Second, we extend the scoring mechanism of Alon et al. (CRYPTO 2026) and integrate it with our framework. Finally, we show that the sensitivity framework applies to constructions from one-way permutations. Specifically, we show another proof of Simon's impossibility of basing collision-resistant hash functions on one-way permutations. Our new proof can be viewed as an abstraction of the argument by Bitansky and Degwekar.

*School of Electrical Engineering, Tel-Aviv University. Supported by Israel Science Foundation (ISF) grant no. 2805/21 and by the European Union (ERC-2022-ADG) under grant agreement no.101097959 NFITSC

†Faculty of Computer and Information Science, Ben-Gurion University

‡Department of Computer Science, Georgetown University

Contents

1 Introduction **1**
- 1.1 Our Results . . . 1
- 1.2 Related Works . . . 3
- 1.3 Organization . . . 4

2 Our Techniques **5**
- 2.1 Our New Sensitivity Framework . . . 5
- 2.2 Optimality of the GGM Construction . . . 6
- 2.3 Technical Comparison to Prior Works . . . 10
- 2.4 AI Statement . . . 10

3 Preliminaries **11**
- 3.1 Notations . . . 11
- 3.2 Black-Box Constructions . . . 12

4 A Sensitivity Framework for Fully Black-Box Lower Bounds **15**

5 Impossibility of Non-Adaptive Constructions of PRFs from PRGs **20**

6 Lower Bounds for Adaptive Constructions of PRFs from PRGs **26**

7 Lower Bounds on Non-Adaptive Stretch Amplification **35**
- 7.1 Corollaries . . . 41

8 Impossibility of Constructions of CRH from OWP **42**
- 8.1 Sensitivity for Multi-Valued Algorithms . . . 42
- 8.2 An Inversion Planting Lemma . . . 44
- 8.3 Proof of Theorem 8.1 . . . 46
- 8.4 Comparison with Bitansky and Degwekar . . . 51

Bibliography **51**

1 Introduction

A pseudorandom function (PRF) family is a collection of functions that appear random to any efficient algorithm. PRFs are fundamental to cryptography, both in theory and in practice, and have found many applications in areas outside of cryptography, such as in complexity and learning theory.

The seminal work of Goldreich, Goldwasser, and Micali [10] (henceforth the GGM construction) shows that a PRF can be constructed from *any* pseudorandom generator (PRG) – an efficient algorithm that, on a random input, outputs a longer string that appears random to any efficient algorithm. Specifically, they constructed a PRF $\mathcal{F} = \left\{f_k : \{0,1\}^{\ell(n)} \to \{0,1\}^n\right\}_{k\in\{0,1\}^n}$ for any polynomially bounded $\ell(n)$, when given black-box access to a length-doubling PRG $G : \{0,1\}^n \to \{0,1\}^{2n}$. The number of calls to G in the GGM construction is $|x|$ for any input x. A variant of the GGM construction that relies on Levin's domain extension technique [14] improves the number of calls to $\omega(\log n)$.[1] However, a fundamental question has remained open since it was first discovered—is the construction optimal in the number of PRG invocations? More precisely, the following remains open:

What is the minimal number of invocations of a (length-doubling) PRG required to build a PRF?

Recently, Beimel et al. [2] made progress in answering this question. For a limited class of constructions that they refer to as *tree constructions*, which generalizes the GGM construction, they prove that the GGM construction is optimal. In fact, their result holds even when considering PRFs with one-bit output. However, the state of affairs for more general (i.e., not necessarily tree) constructions still remains open. More recently, for a natural class of fully black-box constructions, referred to as *query-bounded* reductions,[2] Alon et al. [1] showed that no such construction can make $o(n/\log n)$ and $o(\ell/\log \ell)$ *non-adaptive* calls to the PRG, where ℓ is the input length of the PRF.

In fact, previous results do not rule out the possibility of building a PRF with a *single call* to the PRG.

A related question is understanding the number of calls required to amplify the stretch of a PRG, leaving the following question open:

What is the minimal number of invocations of a one-bit stretch PRG required to build a length-doubling PRG?

1.1 Our Results

Our main result completely resolves the question of black-box constructions of PRF from a PRG whose stretch is $\Omega(n)$, showing that the GGM construction is optimal. Furthermore, we show that the calls have to be *adaptive*. We further show lower bounds for fully black-box PRG stretch amplification in non-adaptive constructions, and prove that there is no non-adaptive fully black-box construction of a length-doubling PRG from any $o(n)$-stretch PRG. All of our results follow from

[1] The final construction adds to the key a hash $h : \{0,1\}^{\ell(n)} \to \{0,1\}^{\omega(\log n)}$ from a pairwise independent hash function family, and applies the GGM construction to $h(x)$.

[2] Roughly, a reduction is query-bounded if the number of interactions it makes with the adversary is upper-bounded by a polynomial that is independent of the adversary's query complexity (for all sufficiently successful adversaries).

a new framework for proving fully black-box lower bounds, which we believe to be of independent interest. We next give details of our results.

Optimality of the GGM construction. Our main result is a lower bound on the number of rounds of interaction with the underlying PRG that is required to construct a PRF. We show that the GGM construction with Levin's trick is optimal whenever the stretch of the underlying PRG is $r(n) = \Omega(n)$.

Theorem 1.1 (Informal, optimality of GGM)**.** *There is no fully black-box construction of a PRF from an r-bit stretch PRG using $O(\frac{\log n}{\log(2+r/n)})$ rounds.*

If $r = \Theta(n)$, then the bound matches GGM with Levin's trick: $O(\log n)$ rounds are ruled out, while $\gamma(n)$ rounds suffice up to a constant factor for any efficiently computable, polynomially bounded $\gamma(n) = \omega(\log n)$. If $r = \omega(n)$, use a b-ary GGM tree, where b is the largest power of two not exceeding $\lfloor (n+r)/n \rfloor$. After hashing to $\gamma(n)$ bits, this uses $\lceil \gamma(n)/\log b \rceil$ rounds. Since $\log b = \Theta(\log(2+r/n))$, this matches the lower-bound threshold up to an arbitrarily slowly growing factor. Whenever $r = o(n)$, Theorem 1.1 does not show that the GGM construction is optimal. We leave this as an interesting open problem.

Lower bounds for PRG stretch amplification. Our second result shows that constructing a ρ-bit stretch PRG from an r-bit stretch PRG using non-adaptive black-box constructions, requires $\Omega(\frac{\rho}{r+\log n})$ calls.

Theorem 1.2 (Informal, lower bounds for stretch amplification)**.** *There is no fully black-box construction of ρ-bit stretch PRG from an r-bit stretch PRG using $o(\frac{\rho}{r+\log n})$ non-adaptive calls.*

For $\rho \geq r$, this is optimal if $r = \omega(\log n)$: partition an input of length $n\lceil \rho/r \rceil$ into $\lceil \rho/r \rceil$ blocks of size n each, apply the underlying PRG on each block independently, and truncate the excess stretch. Note, however, that if the goal is to construct a length-doubling PRG, then the construction fails. This is due to the fact that ρ now depends on the new input length. For $r = o(n)$, we show that there is no non-adaptive construction of a length-doubling PRG from an r-bit stretch PRG.

Theorem 1.3 (Informal, impossibility of non-adaptive construction of length-doubling PRG)**.** *For all $r = o(n)$, there is no fully black-box non-adaptive construction of a length-doubling PRG from an r-bit stretch PRG.*

In Theorem 7.1, we state a general result that implies both Theorems 1.2 and 1.3.

A new framework for proving lower bounds. Towards proving Theorems 1.1 to 1.3, we develop a new framework for proving lower bounds on fully black-box constructions. In essence, our framework reduces the task of proving a lower bound, to constructing an adversary for the construction (e.g., the PRF) that is *insensitive* to a random change in the underlying primitive (e.g., the PRG). In more detail, we consider the notion of one-replacement sensitivity where we compare the acceptance probability of the adversary when interacting with the primitive G and primitive $G^{s \leftarrow v}$, where $G^{s \leftarrow v}$ is defined to be the same as G everywhere except on input s where the output is set to v. Slightly more formally:

Definition 1.4 (Informal, sensitivity of adversaries)**.** *The sensitivity of an adversary $\mathcal{A}$ for a black-box construction $C^{(\cdot)}$, denoted $\sigma(\mathcal{A})$, is defined as*[3]

$$\sigma(\mathcal{A}) = \max_{G,z,v} \mathrm{E}_s \left[\left| \Pr\left[\mathcal{A}^G(z) = 1\right] - \Pr\left[\mathcal{A}^{G^{s \leftarrow v}}(z) = 1\right] \right| \right].$$

The main result of our new framework states that if there is an adversary for the construction whose sensitivity is negligible, then there exists a function G^*, such that the reduction cannot distinguish $G^*(U)$ from a random string, where U is a random input to G^*. Note that G^* is not necessarily a PRG, but rather some function which fools the specific reduction.

Theorem 1.5 (Informal, sensitivity framework)**.** *Let $(C^{(\cdot)}, R)$ be a fully black-box construction of a primitive C from a PRG $G : \{0,1\}^n \to \{0,1\}^{n+r}$, where R is the reduction and makes polynomially many oracle calls in n. Assume there exists an adversary $\mathcal{A}$ for C with negligible sensitivity. Then there exists a function $G^* : \{0,1\}^n \to \{0,1\}^{n+r}$ such that*

$$\left| \Pr_{s \leftarrow \{0,1\}^n} \left[R^{\mathcal{A}, G^*}\left(G^*(s)\right) = 1 \right] - \Pr_{y \leftarrow \{0,1\}^{n+r}} \left[R^{\mathcal{A}, G^*}\left(y\right) = 1 \right] \right| = \mathrm{neg}(n).$$

The theorem implies that R fails on G^* with the adversary $\mathcal{A}$. Thus, to obtain an impossibility result, it suffices to construct an adversary with negligible sensitivity that successfully breaks the construction, and then invoke Theorem 1.5. Importantly, we allow the adversary to be inefficient and depend on the entire truth-table of G.

Our sensitivity framework abstracts the insensitivity-to-local-changes approach underlying earlier black-box separation results [4, 3], which use insensitivity to local changes to preserve the security of the underlying primitive. A closely related argument appears in [12], where insensitivity of a suitably restricted inversion oracle to local changes is used to preserve one-wayness. Theorem 1.5 isolates the approach as a quantitative condition on the adversary's output distribution. This separates the general security argument from the construction-specific task of designing a successful low-sensitivity adversary.

A new proof of Simon's result. Finally, we demonstrate the applicability of our framework beyond PRG-based constructions by giving a new proof of Simon's black-box separation of collision-resistant hash functions from one-way permutations [20]. We do so by extending the framework to adversaries that output strings, measuring sensitivity using statistical distance, and to reductions that attempt to invert a permutation.

Theorem 1.6 (Informal, Simon's black-box impossibility)**.** *There is no fully black-box construction of collision-resistant hash functions from one-way permutations.*

The formal statement and proof appear in Section 8.

1.2 Related Works

Separation of cryptographic primitives is a fundamental question in cryptography that qualitatively measures the relative hardness of different cryptographic primitives. The seminal work of Impagliazzo and Rudich [13] provided the first separations ruling out constructions of key exchange from one-way functions. The class of constructions ruled out in this work is referred to as a fully

[3] Formally, we allow v to depend on s, thus the maximum is taken over all vectors $(v_s)_{s \in \{0,1\}^n}$.

black-box construction (see [19] for a taxonomy on reductions) where the construction uses the underlying primitive as a black-box and the reduction is black-box.

Within the framework of black-box separations, the *meta-reduction* paradigm introduced in the work of Boneh and Venkatesan [5], considers designing a (possibly inefficient) attacker $\mathcal{A}$ that when invoked with the reduction breaks the challenge in the underlying security reduction, then emulating $\mathcal{A}$ in a manner that is efficient and does not affect the interaction of the reduction and the challenger (see also [6, 11, 8, 17, 18]).

Miles and Viola [15, 16] were the first to consider lower bounds on black-box constructions of PRFs from PRGs.[4] They ruled out single-call constructions where the output is one of the PRG's output bits. More recently, Beimel et al. [2] generalized the result of [15, 16], and proved a lower bound on the number of calls in tree constructions, which generalize the GGM construction. Additionally, they showed a lower bound on the key length required for black-box constructions.

The previous work [1] considered query-bounded reductions, whose number of adversary invocations is independent of the adversary's function-query count. Here the reduction's oracle-query bounds may depend polynomially on that count. Bitansky and Degwekar [3] use smoothing, puncturing, and coupling to show that adversarial executions with access to a collision-finding oracle are largely insensitive to suitable local changes in the underlying permutation. This stability allows them to preserve one-wayness in the presence of the collision finder, yielding an alternative proof of Simon's fully black-box separation. They also showed that hard problems in the class Statistical Zero Knowledge cannot be based on collision-resistant hash functions. Holenstein and Sinha [12] similarly show that, for one-way functions, the oracle they design for inverting the candidate construction is unlikely to change its answer when the underlying function's value at a uniformly random input is replaced. They use this stability to show that the inversion oracle does not help invert the underlying functions.

On stretch amplification of PRG, Gennaro et al. [9] consider black-box constructions of PRGs from OWPs. Their results imply that k calls to a $s(n)$-stretch PRG for a constant $k \in \mathbb{N}$ cannot achieve stretch $k \cdot s(n) + \omega(\log n)$, even when the queries are adaptive. Bronson, Juma and Papakonstantinou [7] show that $k \in \mathbb{N}$ non-adaptive calls to a $s(n)$-stretch PRG cannot achieve stretch $s(n) + \omega(\log n)$, where the oracle queries depends only the last $n + O(\log n)$ bits of the input. They also show that with constant non-adaptive calls to a $s(n)$-bit stretch PRG, one cannot construct a PRG with seed length at most $n + O(\log n)$ and with stretch even $s(n) + 1$.

A related question is that of constructing PRFs from non-adaptive PRFs (naPRFs), where security is only guaranteed against non-adaptive adversaries. Soni and Tessaro [21] showed that certain fully black-box non-adaptive constructions of PRFs from naPRFs with a constant number of calls are impossible.

1.3 Organization

We give an overview of our techniques in Section 2. The preliminaries are given in Section 3. We develop the sensitivity framework in Section 4, prove the non-adaptive PRF lower bound in Section 5, and extend it to adaptive rounds in Section 6. Section 7 gives the stretch-amplification lower bounds. Finally, in Section 8 we prove Simon's impossibility result using the sensitivity framework.

[4] The paper considers the question of extending the stretch of a PRG; however, they noted in [16] that their result applies to constructing PRFs from PRGs as well.

2 Our Techniques

We now present our techniques. We first explain our new framework in Section 2.1, where we demonstrate that it suffices to construct a good adversary with negligible sensitivity. Then, in Section 2.2.1, we describe a warm-up, where we show that non-adaptive constructions of a PRF from a PRG are impossible. Finally, in Section 2.2.2, we extend the ideas to adaptive construction. The proof of Theorems 1.2 and 1.3 follows similar ideas to the lower bound on PRF constructions.

2.1 Our New Sensitivity Framework

For ruling out black-box constructions of a primitive Q based on some primitive P, a standard approach is to design an oracle relative to which P exists, but the target primitive does not exist. For the case of PRFs based on PRGs, prior works [2, 1] have relied on a PSPACE oracle to break the underlying PRF while only making polynomially many queries to the underlying PRG.

In this work, we pursue an alternative approach that proceeds in three steps: First, we show that one-replacement sensitivity of any reduction is tied to the one-replacement sensitivity of the underlying adversary. In other words, the (in)sensitivity of the adversary translates to the (in)sensitivity of the reduction. Next, we show that the advantage of the reduction can be bounded by its one-replacement sensitivity. We argue this by computing the expected squared advantage of the reduction for a random G. In the third step, we show how to construct an adversary with negligible one-replacement sensitivity. Finally, we conclude that there exists one particular function G^* that minimizes the advantage, and on this G^* the reduction with the adversary does not distinguish $G^*(U)$ from a uniformly random string.

In more detail:

First step: bounding the sensitivity of the reduction. Consider coupling the execution $R^{\mathcal{A}^G,G}$ with all executions $R^{\mathcal{A}^{G^{s\leftarrow v}},G^{s\leftarrow v}}$ using the same coins for R. Until they diverge, R sees exactly the same transcript. There are only two possible sources of divergence:

1. R queries G at the modified location s, or
2. during one of its calls to $\mathcal{A}$, the two algorithms return different bits.

Let q_G denote the number of calls R makes to G, and let $q_{\mathcal{A}}$ denote the number of calls R makes to $\mathcal{A}^G$. Averaged over all s, the first event contributes at most $q_G/2^n$ to the probability of divergence, and the second contributes at most $q_{\mathcal{A}} \cdot \sigma(\mathcal{A})$. Since q_G is polynomially bounded and the modified location is chosen uniformly, the first event occurs with probability at most $q_G/2^n = \mathrm{neg}(n)$. Therefore, the sensitivity of the reduction is bounded by $\mathrm{neg}(n) + q_{\mathcal{A}} \cdot \sigma(\mathcal{A})$.

Second step: bounding the expected squared advantage of the reduction. We now bound the expected squared advantage. Let

$$\Delta_G = \Pr_{s\leftarrow\{0,1\}^n}\left[R^{\mathcal{A},G}\left(G(s)\right)=1\right] - \Pr_{y\leftarrow\{0,1\}^{n+r}}\left[R^{\mathcal{A},G}\left(y\right)=1\right]$$

denote the reduction's (signed) advantage when the (purported) PRG is G. Our goal is to bound $\mathrm{E}_G\left[\Delta_G^2\right]$. Let $h^G(y) = \Pr\left[R^{\mathcal{A},G}(y) = 1\right]$. Then expanding yields

$$\begin{aligned}\Delta_G^2 &= \mathrm{E}_{s,s' \leftarrow \{0,1\}^n}\left[\left(h^G(G(s)) - \mathrm{E}_{y \leftarrow \{0,1\}^{n+r}}[h^G(y)]\right)\left(h^G(G(s')) - \mathrm{E}_{y' \leftarrow \{0,1\}^{n+r}}[h^G(y')]\right)\right] \\ &= \mathrm{E}_{s,s',y,y'}\left[\left(h^G(G(s)) - h^G(y)\right)\left(h^G(G(s')) - h^G(y')\right)\right]\end{aligned}$$

for all G. Thus, Δ_G^2 measures the correlation between the deviations at two randomly chosen locations. Since $s = s'$ occurs with probability 2^{-n}, we may focus only on $s \neq s'$. In this case, we bound $\mathrm{E}_G\left[\Delta_G^2\right]$ using the sensitivity of R using a symmetry argument detailed below.

Given a uniformly chosen oracle G, consider the following experiment for generating a random function $G_{s,s'} : \{0,1\}^n \to \{0,1\}^{n+r}$:

1. For all $t \neq s, s'$, let $G_{s,s'}(t) = G(t)$.
2. Sample $v \leftarrow \{0,1\}^{n+r}$ and set $G_{s,s'}(s) = v$.
3. Sample $v' \leftarrow \{0,1\}^{n+r}$ and set $G_{s,s'}(s') = v'$.

Clearly $G_{s,s'}$ is distributed uniformly at random. Therefore (ignoring the case $s = s'$)

$$\mathrm{E}_G\left[\Delta_G^2\right] \approx \mathrm{E}_{G,s,s',v,v',y,y'}\left[\left(h^{G_{s,s'}}(v) - h^{G_{s,s'}}(y)\right)\left(h^{G_{s,s'}}(v') - h^{G_{s,s'}}(y')\right)\right].$$

Intuitively, if the reduction were completely independent of G (i.e., had sensitivity 0), the off-diagonal expectation would vanish: v and y are independent uniform values, so their centered difference has zero mean. The diagonal contribution is at most 2^{-n}. We show that because the sensitivity of the reduction is small, we can still argue that the (squared) advantage of the reduction is small.

Consider another experiment: let $G'_{s,s'} = G_{s,s'}^{s \leftarrow y}$. That is, we swap the output v with y. Once again, $G'_{s,s'}$ is distributed uniformly at random. However, now the challenge is either $G'_{s,s'}(s) = y$ or v, whereas before the swap it was either $G_{s,s'}(s) = v$ or y. This means that the difference has its two arguments swapped. Thus the sign has changed, i.e.,

$$\mathrm{E}_G\left[\Delta_G^2\right] \approx -\mathrm{E}_{G,s,s',v,v',y,y'}\left[\left(h^{G'_{s,s'}}(v) - h^{G'_{s,s'}}(y)\right)\left(h^{G'_{s,s'}}(v') - h^{G'_{s,s'}}(y')\right)\right].$$

We can now average the two expressions with opposite signs, which involve closely related executions of R. Rearranging allows to bound $\mathrm{E}_G\left[\Delta_G^2\right]$ using the sensitivity of R.

2.2 Optimality of the GGM Construction

We now demonstrate our lower bound on the number of rounds of interaction with a PRG that are required to construct a PRF. We start with a warm-up, showing that a single round does not suffice.

2.2.1 Impossibility of Non-Adaptive Constructions of PRFs from PRGs

Assume towards contradiction that $(\mathcal{F}, R)$ is a fully black-box construction of a PRF from an r-bit stretch PRG, where each $f_k^{(\cdot)}$ makes c non-adaptive oracle calls. Fix an arbitrary function

$G : \{0,1\}^n \to \{0,1\}^{n+r}$, and let f_k^G denote the resulting instantiation of the construction. By Theorem 1.5, it suffices to design an adversary $\mathcal{A}^{f,G}$ with negligible sensitivity for a non-adaptive construction of a PRF. The simplest adversary queries G on every seed, sends sufficiently many distinct inputs $\mathbf{x} = (x_1, \ldots, x_m)$ to f, and outputs "PRF" if and only if there exists a key for which the outputs $\mathbf{z} = (z_1, \ldots, z_m)$ of f are consistent with the current PRG oracle. Although $\mathcal{A}$ successfully distinguishes f_k from a random function, a rejected output may become consistent with a different key for each of many edited locations of G. This can make the average sensitivity large.

We propose a different strategy. Instead of searching for the "best" key, the adversary computes a *smooth approximation*. This is done by assigning weights to every key, based on how many changes to G are needed for the PRF construction to output $z_1, \ldots, z_m$ (when queried on the fixed $x_1, \ldots, x_m$). Specifically, for a key k, let $\mathcal{T}_k$ denote the set of all seeds to G which the construction needs to query in order to compute $f_k(x_1), \ldots, f_k(x_m)$. Then $|\mathcal{T}_k| \leq cm$. The adversary computes the minimal distance between $G|_{\mathcal{T}_k}$[5] and the set of assignments $Y : \mathcal{T}_k \to \{0,1\}^{n+r}$ that are consistent with the outputs $\mathbf{z}$. Let $\mathsf{dist}_k^G(\mathbf{z})$ be this distance, setting it to ∞ if no such Y exists. The adversary then outputs "PRF" with probability $p^G(\mathbf{z})$, where

$$W^G(\mathbf{z}) = \sum_k 2^{-\mathsf{dist}_k^G(\mathbf{z})}, \qquad p^G(\mathbf{z}) = \min\left\{1, W^G(\mathbf{z})\right\}.$$

Before explaining why the adversary works, we first compare it with the naive adversary. With the new notations, the naive adversary can be described as one that searches for a key that minimizes $\mathsf{dist}_k^G(\mathbf{z})$. Consider instead the expression

$$-\log\left(W^G(\mathbf{z})\right) = -\log\left(\sum_k 2^{-\mathsf{dist}_k^G(\mathbf{z})}\right).$$

Intuitively, this expression approximates $\min_k \mathsf{dist}_k^G(\mathbf{z})$ since small distances dominate the sum. In particular, small distances translate to a high probability of outputting "PRF".

This also explains why the adversary distinguishes f_k from a random function. Indeed, when interacting with f_k, for at least one of the keys, $\mathsf{dist}_k^G(\mathbf{z}) = 0$, hence the adversary outputs "PRF". On the other hand, for a positive integer threshold τ, the number of output vectors $\mathbf{z}$ that are explainable by some key using fewer than τ edits is at most $2^\lambda \mathsf{Vol}_{\{0,1\}^{n+r}}(cm, \tau - 1)$, where $\mathsf{Vol}_\Sigma(d,t)$ denotes the volume of the d-dimensional Hamming ball of radius t over the alphabet Σ, and where λ is the key length. We choose m so that this is small compared with the 2^m possible output vectors, while the total weight of all other explanations is at most $2^{\lambda-\tau}$. This means that the adversary will identify f as being a random function.

To obtain a contradiction, all that is left is to show that $\mathcal{A}$ has negligible sensitivity. Suppose we modify G at one seed s. Let G_s denote the modified function. Consider a fixed key k. If no feasible assignment exists, its weight is zero for both oracles. Otherwise, there are two possibilities:

- The construction f_k (on inputs $x_1, \ldots, x_m$) never queries G on s. In this case, nothing changes for this key, and so
$$\mathsf{dist}_k^{G_s}(\mathbf{z}) = \mathsf{dist}_k^G(\mathbf{z}).$$

[5] We let $G|_{\mathcal{T}_k}$ denote the restriction of G to $\mathcal{T}_k$.

- The construction f_k queries G on s. Then, recalling that $\mathcal{T}_k$ is a set, changing $G(s)$ can alter the minimum number of required corrections by at most one, i.e.,

$$\left|\mathsf{dist}_k^{G_s}(\mathbf{z}) - \mathsf{dist}_k^{G}(\mathbf{z})\right| \leq 1.$$

  This implies that in the sum

$$W^G(\mathbf{z}) = \sum_k 2^{-\mathsf{dist}_k^G(\mathbf{z})},$$

  only entries for keys with $s \in \mathcal{T}_k$ can change, and each changes by at most a factor of 2.

Now sum the affected weights over seeds: $\sum_s \sum_{k:s\in\mathcal{T}_k} 2^{-\mathsf{dist}_k^G(\mathbf{z})} \leq cm\, W^G(\mathbf{z})$. The average over $s \leftarrow \{0,1\}^n$ bounds the sensitivity of $\mathcal{A}$ by $cm/2^n$.

2.2.2 Generalizing the Impossibility to More Rounds

We next explain the idea for proving the impossibility of two rounds of calls to the PRG. The adversary $\mathcal{A}^{f,G}$ makes m distinct queries to f, where m is a sufficiently large polynomial. The adversary receives an answer vector $\mathbf{z} \in \{0,1\}^m$ and asks whether $\mathbf{z}$ has a plausible explanation as the outputs of the construction under some key. Since the construction is adaptive, an explanation must also specify hypothetical answers to the primitive oracle in every round.

Naive weight function. A natural (but naive) approach would be to copy the non-adaptive approach, and take the "best" answers for any key before summing over all the keys. However, the non-adaptive sensitivity argument does not directly extend, since different hypothetical first-round assignments can lead to different second-round query sets. Specifically, if the best first-round assignment changes from Y_1 to Y_1' after modifying $G(s)$, the selected explanation might involve a completely different set of second-round queries. So although each fixed branch is locally stable, we cannot bound the overall sensitivity simply by counting the seeds queried along one fixed branch.

Improved weight function. A better approach would be to take a sum over all possible Y_1 for any fixed key. Now no branch is selected. Instead, every branch has a contribution, so changing G just reweights existing terms and changes their scores. This allows us to bound changes using a weighted sum of the changes in the individual branches. Unfortunately, it raises a different problem, namely, the fact that there are too many branches Y_1. As we next explain, this prevents the adversary from distinguishing the PRF from a random function. Let c_1 denote the number of calls to G in the first round of each evaluation, let $\mathcal{T}_1$ be the set of distinct first-round seeds across the m evaluations for a fixed key, and write $t_1 = |\mathcal{T}_1| \leq c_1 m$. For every seed $s \in \mathcal{T}_1$ there are 2^{n+r} options for $Y_1(s)$. Thus, the first round has $2^{t_1(n+r)}$ possible assignments Y_1. A naive sum can therefore incur a multiplicative factor as large as $2^{t_1(n+r)}$. Thus, transcripts can acquire a huge score because there are so many branches, potentially preventing the adversary from distinguishing the PRF from a random function.

We fix this by *penalizing* each hypothetical first-round assignment according to the number of queried PRG seeds on which it disagrees with the actual oracle G. For a fixed key k and first-round assignment Y_1, denote the second-round non-adaptive weight by $w_{2,k}^G(Y_1, \mathbf{z})$. As previously, this weight is 2^{-a}, where a is the minimum number of distinct second-round seeds whose answers must differ from G to produce $\mathbf{z}$, keeping Y_1 fixed; the weight is zero if no second-round assignment produces $\mathbf{z}$. Specifically, when summing over the possible first-round assignments $Y_1 : \mathcal{T}_1 \to \{0,1\}^{n+r}$,

we multiply $w_{2,k}^G(Y_1, \mathbf{z})$ by $\eta^{\mathsf{dist}(Y_1, G|_{\mathcal{T}_1})}$, where $0 < \eta < 1$ is sufficiently small and $\mathsf{dist}(Y_1, G|_{\mathcal{T}_1})$ is the number of seeds $s \in \mathcal{T}_1$ for which $Y_1(s) \neq G(s)$. Thus, assignments that disagree with G on many queried seeds receive exponentially smaller penalty factors. Writing $\mathcal{T}_{1,k}$ for the first-round query set associated with key k, the total weight at this stage is

$$W^G(\mathbf{z}) = \sum_{k \in \{0,1\}^\lambda} \sum_{Y_1 : \mathcal{T}_{1,k} \to \{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_1, G|_{\mathcal{T}_{1,k}})} w_{2,k}^G(Y_1, \mathbf{z}).$$

The adversary accepts with probability $p^G(\mathbf{z}) = \min\left\{1, W^G(\mathbf{z})\right\}$.

For each $j = 0, \ldots, t_1$, there are $\binom{t_1}{j}(2^{n+r}-1)^j$ assignments Y_1 with $\mathsf{dist}(Y_1, G|_{\mathcal{T}_1}) = j$: choose the j seeds on which Y_1 disagrees with G, and choose one of the $2^{n+r}-1$ different answers at each of these seeds. Every such assignment has penalty factor η^j. Therefore, summing the penalty factors over all assignments gives

$$\sum_{Y_1 : \mathcal{T}_1 \to \{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_1, G|_{\mathcal{T}_1})} = \sum_{j=0}^{t_1} \binom{t_1}{j} (2^{n+r}-1)^j \eta^j = \left(1 + (2^{n+r}-1)\eta\right)^{t_1}.$$

Choosing $\eta = 2^{-(n+r+\gamma)}$, where $\gamma = \omega(\log n)$, makes this sum $1 + \mathrm{neg}(n)$ for polynomial t_1. For every fixed key and first-round assignment Y_1, the non-adaptive analysis of the second round gives some upper bound $\varepsilon = \varepsilon(m)$ on the expected weight $w_{2,k}^G(Y_1, \mathbf{z})$ when $\mathbf{z}$ is uniformly random. Since the first-round penalty factors do not depend on $\mathbf{z}$, linearity of expectation bounds the expected weighted sum for each key by $\varepsilon\left(1 + (2^{n+r}-1)\eta\right)^{t_1} = \varepsilon(1 + \mathrm{neg}(n))$. Choosing m sufficiently large, but still polynomial, makes $2^\lambda \varepsilon$ negligible, where λ is the key length. Thus, after summing over all keys and capping the total weight at 1, the adversary accepts a random-function transcript with negligible probability. In contrast, for a PRF transcript, the actual key and actual oracle answers explain $\mathbf{z}$ exactly in both rounds. They therefore give a first-round penalty factor $\eta^0 = 1$ and a second-round weight $2^{-0} = 1$, so the total weight is at least 1 and the adversary always accepts.

Now, while this choice of η controls the loss from summing over the branches, it creates a second problem for the sensitivity bound. Specifically, η being too small can create large multiplicative changes in the penalty factors. To see this, suppose we change $G(s)$ for a seed s that is queried in the first round. For a fixed branch Y_1, its Hamming distance can change by one and its penalty factor then increases from η^t to $\eta^{t-1} = \eta^{-1}\eta^t$. Therefore its current-round penalty factor can increase by a factor of $\eta^{-1} = 2^{n+r+\gamma}$. Averaging over the 2^n possible seeds and accounting for the (possible) $c_1 m$ first-round seeds gives a contribution of order $c_1 m\, 2^{r+\gamma}$ to the sensitivity, which is too large.

Final weight function. To compensate for this, for each fixed key we raise the sum of the weighted contributions of all possible first-round assignments to the power α, for a sufficiently small parameter $0 < \alpha < 1/2$. Specifically, we define

$$w_k^G(\mathbf{z}) = \left(\sum_{Y_1 : \mathcal{T}_{1,k} \to \{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_1, G|_{\mathcal{T}_{1,k}})} w_{2,k}^G(Y_1, \mathbf{z}) \right)^\alpha,$$

and replace the preceding total weight and acceptance probability by

$$W^G(\mathbf{z}) = \sum_{k \in \{0,1\}^\lambda} w_k^G(\mathbf{z}), \qquad p^G(\mathbf{z}) = \min\left\{1, W^G(\mathbf{z})\right\}.$$

Thus, the power is applied separately for each key, before summing over keys. This means that the first-round penalty incurs a factor of $\eta^{-\alpha}$ instead of η^{-1}. Accounting also for the factor of at most two by which a fixed second-round weight can change, the first-round contribution to the sensitivity bound is of order $c_1 m\, 2^{\alpha(n+r+\gamma)-n}$ instead of the previous $c_1 m\, 2^{r+\gamma}$. Taking α sufficiently small allows us to conclude that the sensitivity is negligible.

However, a small α increases the likelihood that a random function would be identified as a PRF. Specifically, raising a per-key sum in $(0, 1)$ to a very small power α can result in a value very close to 1, even when interacting with a random function. This can be compensated with a sufficiently large number of inputs m, which limits how small α can be while keeping m polynomial.

Finally, our analysis shows that these ideas generalize to a larger number of rounds d. The logarithmic bound on d will come from balancing the sensitivity bound with the requirement that m remains polynomial. Specifically, every additional round introduces another power of α in the soundness exponent. Since the last round uses the non-adaptive weight, the resulting exponent contains α^{d-1}. Thus, our choice of m incurs a factor of $\alpha^{-(d-1)}$, in addition to the polynomial factors already present in the non-adaptive case.

2.3 Technical Comparison to Prior Works

Our techniques combine two prior lines of work. First, our sensitivity framework is related to earlier black-box separations based on stability under local oracle changes. Bitansky et al. [4] use randomized decision thresholds and smoothness arguments to construct an oracle that solves statistical zero-knowledge problems while preserving the security of underlying one-way permutations and indistinguishability obfuscation. Bitansky and Degwekar [3] develop related smoothing, puncturing, and coupling arguments for collision-finding oracles, obtaining new proofs of black-box separations for collision-resistant hashing. Our framework follows the same approach, but isolates a quantitative condition on the adversary's acceptance probabilities: we transfer one-replacement sensitivity to the reduction and then bound its expected squared PRG distinguishing advantage. This separates the general stability-to-security argument from the construction-specific task of designing a successful low-sensitivity adversary.

Second, our scoring mechanism extends the one developed in [1]. However, the scoring used in [1] was too rigid (i.e., sensitive), resulting in the query-bounded restriction. Concretely, although [1] already smooths its decision using a randomized threshold, its maximum-over-keys scoring rule can still have inverse-polynomial one-replacement sensitivity. Our non-adaptive scoring replaces this with a sum of exponentially decaying distance weights. Thus, our results crucially combine and extend these two lines of work.

2.4 AI Statement

The authors designed the black-box separation framework based on sensitivity and the non-adaptive PRF adversary with the assistance of ChatGPT after independently identifying the rigidity (sensitivity) limitation of the prior scoring mechanism of [1]. Subsequently, through several interactions, ChatGPT suggested the technical choices of weights for the adaptive PRF adversary and provided an initial draft of the black-box separation proof.

The proof provided by ChatGPT was first the impossibility of non-adaptive PRF construction. This was followed by the impossibility of constant number of adaptive rounds and then it proved

optimality of the GGM construction. We then prompted it on the PRG stretch amplification for ρ that is independent of ℓ.

The authors significantly revised these contributions and determined the final content and structure of the paper. In particular, we abstracted the sensitivity framework. We then prompted ChatGPT to see if the framework can be applied to other questions, such as the impossibility of non-adaptively constructing length-doubling PRGs from $o(n)$-bit stretch PRGs, and to obtain a new proof of Simon's result. The authors take full responsibility for this paper and all its contents.

3 Preliminaries

3.1 Notations

For $n, m \in \mathbb{N}$ we let $[n] = \{1, 2 \dots n\}$. All logarithms are taken in base 2. For a set $\mathcal{S}$, we write $s \leftarrow \mathcal{S}$ to indicate that s is selected uniformly at random from $\mathcal{S}$. Given a random variable (or a distribution) X, we write $x \leftarrow X$ to indicate that x is selected according to X. For every nonempty finite set S, we write $\mathsf{U}(S)$ for the uniform distribution over S. A function $\mu : \mathbb{N} \to [0,1]$ is called negligible, if for every positive polynomial $p(\cdot)$ and all sufficiently large n, $\mu(n) < 1/p(n)$. For $n, m \in \mathbb{N}$, we let $\mathcal{F}_{n,m} = \{f : \{0,1\}^n \to \{0,1\}^m\}$ denote the set of all functions from $\{0,1\}^n$ to $\{0,1\}^m$, and let $\mathcal{F}_n = \mathcal{F}_{n,1}$. We let $\Pi_n = \{\pi : \{0,1\}^n \to \{0,1\}^n : \pi \text{ is a permutation}\}$ denote the set of all permutations over $\{0,1\}^n$. For two functions $f, g : X \to Y$, we let their distance $\mathsf{dist}(f,g)$ be the number of inputs on which they disagree, i.e.,

$$\mathsf{dist}(f,g) = |\{x \in X : f(x) \neq g(x)\}| .$$

For $X' \subseteq X$ we let $f|_{X'} : X' \to Y$ denote the restriction of f to the domain X'.

For a function $G : \{0,1\}^n \to \{0,1\}^{n'}$, a seed $s \in \{0,1\}^n$, and a value $v_s \in \{0,1\}^{n'}$, define $G^{s \leftarrow v_s} : \{0,1\}^n \to \{0,1\}^{n'}$ to agree with G everywhere except on s, where its output is v_s. That is,

$$G^{s \leftarrow v_s}(s') = \begin{cases} G(s') & \text{if } s' \neq s \\ v_s & \text{otherwise.} \end{cases}$$

We use the following elementary bound

Fact 3.1. *Define $\phi : \mathbb{R} \to \mathbb{R}$ as $\phi(x) = \min\{1, x\}$. Then for all $a, b \geq 0$,*

$$|\phi(a) - \phi(b)| \leq \frac{|a-b|}{\max\{1, b\}}.$$

Proof. If $b \leq 1$, then the right-hand side becomes $|a-b|$, while the left-hand side is equal to $|\phi(a) - \phi(b)| = |\phi(a) - b|$, which is clearly upper bounded by $|a-b|$. If $b > 1$, the left-hand side is zero when $a \geq 1$, whereas for $a < 1$,

$$|\phi(a) - \phi(b)| = 1 - a \leq \frac{b-a}{b}.$$

□

Definition 3.2 (Volume of Hamming balls and weighted Hamming balls)**.** *Let $d, r \geq 0$ be integers and let Σ be a finite alphabet. The volume of the d-dimensional Hamming ball of radius r over the alphabet Σ is*

$$\mathsf{Vol}_{\Sigma}(d, r) = \sum_{j=0}^{\min\{d,r\}} \binom{d}{j} \cdot (|\Sigma| - 1)^j .$$

Let $\eta > 0$. A weighted Hamming ball assigns a factor η^j to points at distance j from the center. These weights decrease exponentially with distance when $0 < \eta < 1$. Thus, the volume of the η-weighted d-dimensional Hamming ball of radius r over the alphabet Σ is

$$\mathsf{Vol}_{\Sigma}(d, r, \eta) = \sum_{j=0}^{\min\{d,r\}} \binom{d}{j} \cdot (|\Sigma| - 1)^j \cdot \eta^j .$$

When $r \geq d$,

$$\mathsf{Vol}_{\Sigma}(d, r, \eta) = \mathsf{Vol}_{\Sigma}(d, d, \eta) = (1 + (|\Sigma| - 1)\,\eta)^d .$$

We write $\mathsf{Vol}_{\Sigma}(d, \eta)$ to alleviate notation.

The following are two simple upper bounds on the volume.

Fact 3.3. *Let $d, r \geq 0$ be integers, let Σ be a finite alphabet, and let $\eta > 0$. Then*

$$\mathsf{Vol}_{\Sigma}(d, r, \eta) \leq \min\left\{(1 + |\Sigma| \cdot \eta)^d,\ (r + 1) \cdot \max\{1, d \cdot |\Sigma| \cdot \eta\}^r\right\}.$$

Proof. The first inequality follows from the binomial formula. For the second inequality, write

$$\mathsf{Vol}_{\Sigma}(d, r, \eta) \leq \sum_{j=0}^{r} d^j \cdot |\Sigma|^j \cdot \eta^j \leq (r + 1) \cdot \max\{1, d \cdot |\Sigma| \cdot \eta\}^r .$$

□

3.2 Black-Box Constructions

In this section, we define fully black-box constructions of one primitive from another. We start with constructions of pseudorandom function (PRF) from pseudorandom generators (PRGs) and then define black-box stretch amplification of a PRG.

3.2.1 Black-Box Constructions of Pseudorandom Functions

We next define black-box constructions of a PRF from a PRG. We state the definition for PRFs with a single-bit output. This will suffice since our impossibility result holds for this case.

Definition 3.4 (Fully black-box PRF constructions)**.** *A* fully black-box construction *of a PRF from an $r(n)$-bit stretch PRG is a pair $(\mathcal{F}, R)$ where*

$$\mathcal{F} = \left\{f_k^{(\cdot)} : \{0,1\}^{\ell(n)} \to \{0,1\}\right\}_{n \in \mathbb{N}, k \in \{0,1\}^{\lambda(n)}}$$

is an efficiently computable oracle-aided function family, where $\ell(n) = \omega(\log n)$,[6] *and* $R^{(\cdot),(\cdot)}$ *is a probabilistic oracle-aided algorithm such that the following holds. There exists a polynomial* $q : \mathbb{N} \to \mathbb{N}$ *such that for every* $\mathcal{G} = \{G_n : \{0,1\}^n \to \{0,1\}^{n+r(n)}\}_{n\in\mathbb{N}}$ *and every oracle-aided algorithm* $\mathcal{A}^{(\cdot),(\cdot)}$ *for which there exists a constant* $c \in \mathbb{N}$ *such that*

$$\left|\Pr_{k\leftarrow\{0,1\}^{\lambda(n)}}\left[\mathcal{A}^{f_k^{G_n},G_n}(1^n) = 1\right] - \Pr_{f\leftarrow\mathcal{F}_{\ell(n)}}\left[\mathcal{A}^{f,G_n}(1^n) = 1\right]\right| \geq \frac{1}{n^c}$$

for infinitely many n*'s,*

$$\left|\Pr_{s\leftarrow\{0,1\}^n}\left[R^{\mathcal{A}^{G_n},G_n}(1^n, G_n(s)) = 1\right] - \Pr_{y\leftarrow\{0,1\}^{n+r(n)}}\left[R^{\mathcal{A}^{G_n},G_n}(1^n, y) = 1\right]\right| \geq \frac{1}{q(n^c)}$$

for infinitely many n*'s. The probabilities include all private randomness of* $\mathcal{A}$ *and* R*.*

$(\mathcal{F}, R)$ *is a fully black-box construction of a* weak PRF*, if the above holds for all algorithms* $\mathcal{A}$ *that query i.i.d. uniform queries to* f*.*

Remark 3.5. *We require the number of calls* R *makes to each of its oracles to be bounded by a fixed polynomial in* n *and* m*, where* m *bounds the number of function queries in each invocation of the adversary.*

Remark 3.6. *Note that every time the reduction* R *calls the adversary* $\mathcal{A}$*, the adversary responds with inputs* $x_1, \ldots, x_m$ *(perhaps chosen adaptively) and expects answers that are supposedly provided by its oracle (which is either a PRF or a random function). Now, the reduction has to provide it with answers* $z_1, \ldots, z_m$*. We bound* R*'s own running time by a fixed polynomial in* n *and* m*, where* m *bounds the number of function queries in each adversary invocation. The adversary's internal computation is not charged to this bound;* R*'s processing of the visible interaction is charged. Each invocation starts afresh with independent private randomness. The reduction sees the function-query interaction and final output, but not the adversary's private coins, internal state, or internal queries to* G*. In particular, we allow the reduction to call its oracles a number of times that depends on* m*. This was* not *the case in [1], which assumed the reduction is query-bounded, namely, the number of times* $q_{\mathcal{A}}$ *it calls* $\mathcal{A}$ *is independent of* m[7] *(*R *could still call* G *a number of times that depends on* m*). Our results in Sections 5 and 6 apply to the more general case, where* $q_{\mathcal{A}}$ *depends on* m*.*

We next define the round complexity of a black-box construction.

Definition 3.7 (d-round constructions)**.** *Fix functions* $\ell(n)$*,* $r(n)$*, and* $\lambda(n)$*, and let*

$$\mathcal{F} = \left\{f_k^{(\cdot)} : \{0,1\}^{\ell(n)} \to \{0,1\}\right\}_{n\in\mathbb{N},k\in\{0,1\}^{\lambda(n)}}$$

be a black-box PRF construction from an $r(n)$*-bit stretch PRG. For a function* $d : \mathbb{N} \to \mathbb{N}$*, we say that* $\mathcal{F}$ *is a* $d(n)$-round construction *if the following holds for all* $n \in \mathbb{N}$*. There exist* $d(n)$ *functions* $c_1, \ldots, c_d : \mathbb{N} \to \mathbb{N}$*, such that for every* $i \in [d(n)]$ *there exist* $c_i(n)$ *functions*

$$S_{i,1}, \ldots, S_{i,c_i(n)} : \{0,1\}^{\ell(n)} \times \{0,1\}^{\lambda(n)} \times \left(\{0,1\}^{n+r(n)}\right)^{c_1(n)+c_2(n)+\ldots+c_{i-1}(n)} \to \{0,1\}^n,$$

[6] Otherwise, when $\ell(n) = O(\log n)$ the PRF can be constructed unconditionally by simply letting the key be the truth table of the function (in particular, there is no need for a reduction).

[7] Technically, query-bounded only requires this to be the case assuming the adversary is "sufficiently good".

and there exists a function $P : \{0,1\}^{\ell(n)} \times \{0,1\}^{\lambda(n)} \times (\{0,1\}^{n+r(n)})^{c_1(n)+\ldots+c_d(n)} \to \{0,1\}$ *such that*

$$f_k^G(x) = P\left(x, k, (G(s_{i,j}))_{i\in[d(n)],j\in[c_i(n)]}\right)$$

for all $G : \{0,1\}^n \to \{0,1\}^{n+r(n)}$, $x \in \{0,1\}^\ell$ *and* $k \in \{0,1\}^\lambda$, *where*

$$s_{i,j} = S_{i,j}\left(x, k, (G(s_{i',j}))_{i'\in[i-1],j\in[c_{i'}(n)]}\right)$$

for all $i \in [d(n)]$ *and* $j \in [c_i(n)]$. *The functions* $S_{i,j}$ *and* P *are uniformly efficiently computable, and* $\sum_{i=1}^{d(n)} c_i(n)$ *is polynomially bounded.*

3.2.2 Black-Box Stretch Amplification of PRGs

We now define black-box stretch amplification of a PRG.

Definition 3.8 (Fully black-box PRG stretch amplification)**.** *Fix* $\ell : \mathbb{N} \to \mathbb{N}$ *and* $\rho : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$. *A* fully black-box construction *of a* $\rho(n,\ell)$*-bit stretch PRG from an* $r(n)$*-bit stretch PRG is a pair* $(\mathcal{C}, R)$ *where*

$$\mathcal{C} = \left\{C_n^{(\cdot)} : \{0,1\}^{\ell(n)} \to \{0,1\}^{\ell(n)+\rho(n,\ell)}\right\}_{n\in\mathbb{N}}$$

is an efficiently computable oracle-aided function family and $R^{(\cdot),(\cdot)}$ *is a* PPT *oracle-aided algorithm such that the following holds. There exists a polynomial* $q : \mathbb{N} \to \mathbb{N}$ *such that for every* $\mathcal{G} = \{G_n : \{0,1\}^n \to \{0,1\}^{n+r(n)}\}_{n\in\mathbb{N}}$ *and every oracle-aided algorithm* $\mathcal{A}^{(\cdot)}$ *for which there exists a constant* $c \in \mathbb{N}$ *such that*

$$\left|\Pr_{x\leftarrow\{0,1\}^{\ell(n)}}\left[\mathcal{A}^{G_n}(1^n, C_n^{G_n}(x)) = 1\right] - \Pr_{z\leftarrow\{0,1\}^{\ell(n)+\rho(n,\ell)}}\left[\mathcal{A}^{G_n}(1^n, z) = 1\right]\right| \geq \frac{1}{n^c}$$

for infinitely many n*'s,*

$$\left|\Pr_{s\leftarrow\{0,1\}^n}\left[R^{\mathcal{A}^{G_n},G_n}(1^n, G_n(s)) = 1\right] - \Pr_{y\leftarrow\{0,1\}^{n+r(n)}}\left[R^{\mathcal{A}^{G_n},G_n}(1^n, y) = 1\right]\right| \geq \frac{1}{q(n^c)}$$

for infinitely many n*'s. The probabilities include all private randomness of* $\mathcal{A}$ *and* R.

We next define non-adaptive black-box constructions of PRG stretch amplification.

Definition 3.9 (c-call non-adaptive constructions)**.** *Let* $\ell(n)$, $r(n)$, *and* $\rho(n,\ell)$ *be polynomially bounded, and let*

$$\mathcal{C} = \left\{C_n^{(\cdot)} : \{0,1\}^{\ell(n)} \to \{0,1\}^{\ell(n)+\rho(n,\ell)}\right\}_{n\in\mathbb{N}}$$

be a black-box ρ*-bit stretch PRG construction from an* $r(n)$*-bit stretch PRG. For a function* $c : \mathbb{N} \to \mathbb{N}$, *we say that* $\mathcal{C}$ *is a* $c(\cdot)$-call non-adaptive construction *if for every* $i \in [c(n)]$ *there exist* $S_i : \{0,1\}^{\ell(n)} \to \{0,1\}^n$ *and* $P : \{0,1\}^{\ell(n)} \times (\{0,1\}^{n+r(n)})^{c(n)} \to \{0,1\}^{\ell(n)+\rho(n,\ell)}$ *such that*

$$C_n^G(x) = P\left(x, G\left(S_1\left(x\right)\right), \ldots, G\left(S_c\left(x\right)\right)\right)$$

for all $G : \{0,1\}^n \to \{0,1\}^{n+r(n)}$ *and* $x \in \{0,1\}^\ell$. *The functions* S_i *and* P *are uniformly efficiently computable, and* $c(n)$ *is polynomially bounded.*

3.2.3 Black-Box Constructions of Collision-Resistant Hash Functions

Finally, we define black-box constructions of a CRH from a OWP.

Definition 3.10 (Fully black-box CRHF constructions from one-way permutations)**.** *A* fully black-box construction *of a collision-resistant hash function family from a one-way permutation is a pair* $(\mathcal{H}, R)$*, where*

$$\mathcal{H} = \left\{h_k^{(\cdot)} : \{0,1\}^{\ell(n)} \to \{0,1\}^{\ell_{\mathsf{out}}(n)}\right\}_{n\in\mathbb{N}, k\in\{0,1\}^{\lambda(n)}}$$

is an efficiently computable deterministic oracle-aided function family satisfying $\ell(n) > \ell_{\mathsf{out}}(n)$*, and* $R^{(\cdot),(\cdot)}$ *is a* PPT *oracle-aided algorithm, such that the following holds. There exists a polynomial* $q : \mathbb{N} \to \mathbb{N}$ *such that for every sequence of permutations* $\Pi = \{\pi_n \in \Pi_n\}_{n\in\mathbb{N}}$ *and every oracle-aided algorithm* $\mathcal{A}^{(\cdot)}$*, if there exists* $\alpha \in \mathbb{N}$ *for which*

$$\Pr_{k\leftarrow\{0,1\}^{\lambda(n)}}\left[\mathcal{A}^{\pi_n}(1^n, k) = (x, x'),\ x \neq x',\ h_k^{\pi_n}(x) = h_k^{\pi_n}(x')\right] \geq \frac{1}{n^\alpha}$$

for infinitely many n*, then*

$$\Pr_{s\leftarrow\{0,1\}^n}\left[R^{\mathcal{A}^{\pi_n},\pi_n}(1^n, \pi_n(s)) = s\right] \geq \frac{1}{q(n^\alpha)}$$

for infinitely many n*. The probabilities include all private randomness of* $\mathcal{A}$ *and* R*.*

4 A Sensitivity Framework for Fully Black-Box Lower Bounds

In this section, we develop our main tools for obtaining lower bounds on fully black-box constructions. Our goal is to show that if a reduction R has oracle access to an adversary whose sensitivity with respect to a function G is small, then the reduction fails for some function G^*. The proof is largely done in two steps: Lemma 4.2 transfers low sensitivity from the adversary to the reduction; Lemma 4.3 converts sensitivity into a second-moment bound; Theorem 4.4 combines the two. We first formally define the notion of sensitivity. Throughout the section, we fix $n, n', m \in \mathbb{N}$.

Definition 4.1 (Sensitivity of algorithms)**.** *Let* $\mathcal{A}^{(\cdot)}$ *be an oracle-aided algorithm that on input* $\mathbf{z} \in \{0,1\}^m$ *and oracle access to* $G : \{0,1\}^n \to \{0,1\}^{n'}$ *outputs a Boolean value, and let* $p^G(\mathbf{z}) = \Pr\left[\mathcal{A}^G(\mathbf{z}) = 1\right]$*. For a vector of values* $\mathbf{v} = (v_s)_{s\in\{0,1\}^n}$*, where* $v_s \in \{0,1\}^{n'}$ *for every* $s \in \{0,1\}^n$*, and a vector* $\mathbf{z} \in \{0,1\}^m$*, define*

$$\sigma^G(\mathcal{A}, \mathbf{v}, \mathbf{z}) = \mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\left|p^G(\mathbf{z}) - p^{G^{s\leftarrow v_s}}(\mathbf{z})\right|\right].$$

For fixed G*, define* $\sigma^G(\mathcal{A}) = \max_{\mathbf{v},\mathbf{z}} \sigma^G(\mathcal{A}, \mathbf{v}, \mathbf{z})$*. We define the* sensitivity of $\mathcal{A}$ *as*

$$\sigma(\mathcal{A}) = \max_G \sigma^G(\mathcal{A}).$$

The definition naturally generalizes to algorithms $\mathcal{A}^{f,G}$ *that receive* $\mathbf{z} = (z_1, \ldots, z_m)$ *from* f *interactively.*

Lemma 4.2 (Sensitivity transfer)**.** *Let $\mathcal{A}^{(\cdot)}$ be an oracle-aided algorithm that on input $\mathbf{z} \in \{0,1\}^m$ and oracle access to a function $G : \{0,1\}^n \to \{0,1\}^{n'}$, outputs 1 with probability $p^G(\mathbf{z})$ and outputs 0 with probability $1 - p^G(\mathbf{z})$. Consider an oracle-aided algorithm $R^{(\cdot),(\cdot)}$ that on $y \in \{0,1\}^{n'}$ and 1^n outputs a Boolean value, and define $h^G : \{0,1\}^{n'} \to [0,1]$ as*

$$h^G(y) = \Pr\left[R^{\mathcal{A}^G,G}(1^n, y) = 1\right],$$

where the probability is taken over the reduction's coins and the private coins of $\mathcal{A}^G$ over all calls to it. We write $\sigma(R)$ for the sensitivity of the composed algorithm $R^{\mathcal{A}^G,G}$. Let q_G denote the number of calls R makes to G, and let $q_{\mathcal{A}}$ denote the number of calls R makes to $\mathcal{A}^G$. Then

$$\sigma(R) \leq \frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A}).$$

Moreover, this holds even for algorithms $\mathcal{A}^{f,G}$ that receive $\mathbf{z}$ from f on queries that are sampled independently of G.

Proof. Fix $y \in \{0,1\}^{n'}$, $G : \{0,1\}^n \to \{0,1\}^{n'}$, and $\mathbf{v} = (v_s)_{s\in\{0,1\}^n}$. We bound

$$\mathrm{E}_{s\in\{0,1\}^n}\left[\left|h^G(y) - h^{G^{s\leftarrow v_s}}(y)\right|\right].$$

The proof is done by coupling the execution $R^{\mathcal{A},G}(y)$ with all of the executions $R^{\mathcal{A}^{G^{s\leftarrow v_s}},G^{s\leftarrow v_s}}(y)$ over all $s \in \{0,1\}^n$. Denote the base execution by E and for every $s \in \{0,1\}^n$, denote the perturbed execution by E_s. We couple them using the same randomness for R in all executions. At the ith call to the adversary, all adversaries sample $p_i \leftarrow [0,1]$ uniformly at random (using the same randomness). As long as E and E_s have the same transcript, the message R sends to the adversary is the same string $\mathbf{z}_i \in \{0,1\}^m$. In this case, in E the adversary $\mathcal{A}$ responds with 1 if and only if $p_i \leq p^G(\mathbf{z}_i)$, and in E_s it responds with 1 if and only if $p_i \leq p^{G^{s\leftarrow v_s}}(\mathbf{z}_i)$. After the first time E_s diverges from E, we complete the execution with arbitrary fresh randomness having the correct marginal distribution.

Let F_s denote the event that E_s diverges from E. Observe that if E and E_s do not diverge, then R outputs the same bit in both executions. Therefore, for every $s \in \{0,1\}^n$,

$$\left|h^G(y) - h^{G^{s\leftarrow v_s}}(y)\right| \leq \Pr\left[\mathsf{F}_s\right].$$

There are two ways the executions may diverge. The first is if R queries G on s, and the second is if $\mathcal{A}$ responds in execution E differently from execution E_s. Let F'_s be the event that divergence first occurred because R queried G on s, and let $\mathsf{F}_{s,i}$ be the event that divergence first occurred at the ith call to the adversary. Since R makes at most q_G queries to G, it follows that

$$\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\Pr\left[\mathsf{F}'_s\right]\right] = \frac{1}{2^n} \cdot \sum_{s\in\{0,1\}^n} \Pr\left[\mathsf{F}'_s\right] \leq \frac{q_G}{2^n}. \tag{1}$$

Now, consider the ith call to the adversary in execution E. Let H_i denote the coupled history until p_i is sampled, and for every $s \in \{0,1\}^n$, let $I_{s,i}$ be the indicator random variable for the event that E and E_s are coupled until p_i is sampled. Then

$$\Pr\left[\mathsf{F}_{s,i} \mid H_i\right] = I_{s,i} \cdot \left|p^G(\mathbf{z}_i) - p^{G^{s\leftarrow v_s}}(\mathbf{z}_i)\right|.$$

Thus,

$$\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\Pr\left[\mathsf{F}_{s,i} \mid H_i\right]\right] \le \sigma(\mathcal{A}). \tag{2}$$

Since R calls the adversary at most $q_{\mathcal{A}}$ times, combining Equations (1) and (2) yields

$$\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\Pr\left[\mathsf{F}_s\right]\right] \le \frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A}).$$

The "moreover" part follows from coupling the queries made by $\mathcal{A}$ to f to be identical across all executions. □

We next state and prove the second lemma. Roughly, it bounds the expected squared gap between the average of h^G on input $G(s)$ and its average on a uniform y using the algorithm's sensitivity.

Lemma 4.3. *Let $R^{(\cdot)}$ be an oracle-aided algorithm that on input $y \in \{0,1\}^{n'}$ and oracle access to $G : \{0,1\}^n \to \{0,1\}^{n'}$ outputs a Boolean value, and let $h^G(y) = \Pr\left[R^G(y) = 1\right]$. For every $G : \{0,1\}^n \to \{0,1\}^{n'}$ define*

$$\Delta_G = \mathrm{E}_{s\leftarrow\{0,1\}^n}\left[h^G(G(s))\right] - \mathrm{E}_{y\leftarrow\{0,1\}^{n'}}[h^G(y)].$$

Then

$$\mathrm{E}_{G\leftarrow\mathcal{F}_{n,n'}}\left[\Delta_G^2\right] \le 4\sigma(R) + \frac{1}{2^n}.$$

Proof. For every $G : \{0,1\}^n \to \{0,1\}^{n'}$, denote $\bar{h}_G = \mathrm{E}_{y\leftarrow\{0,1\}^{n'}}[h^G(y)]$. Then

$$\begin{aligned}
\Delta_G^2 &= \left(\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[h^G(G(s))\right] - \bar{h}_G\right)^2 \\
&= \left(\frac{1}{2^n}\cdot\sum_{s\in\{0,1\}^n} h^G(G(s)) - \bar{h}_G\right)^2 \\
&= \frac{1}{2^{2n}}\cdot\sum_{s,s'\in\{0,1\}^n} h^G(G(s))\cdot h^G(G(s')) - \frac{2}{2^n}\cdot\bar{h}_G\cdot\sum_{s\in\{0,1\}^n} h^G(G(s)) + \bar{h}_G^2 \\
&= \frac{1}{2^{2n}}\cdot\sum_{s,s'\in\{0,1\}^n}\left(h^G(G(s)) - \bar{h}_G\right)\left(h^G(G(s')) - \bar{h}_G\right).
\end{aligned}$$

For $s, s' \in \{0,1\}^n$, let

$$\Theta_{s,s'} = \mathrm{E}_{G\leftarrow\mathcal{F}_{n,n'}}\left[\left(h^G(G(s)) - \bar{h}_G\right)\left(h^G(G(s')) - \bar{h}_G\right)\right].$$

For brevity, we will write E_G instead. Then by linearity of expectation,

$$\mathrm{E}_G\left[\Delta_G^2\right] = \frac{1}{2^{2n}}\cdot\sum_{s,s'\in\{0,1\}^n}\Theta_{s,s'}. \tag{3}$$

We now bound $\Theta_{s,s'}$ for all $s, s' \in \{0,1\}^n$. Observe that if $s = s'$, then

$$\Theta_{s,s} = \mathrm{E}_G\left[\left(h^G(G(s)) - \bar{h}_G\right)^2\right] \le 1,$$

where the inequality is because h^G outputs values in $[0,1]$. Thus,

$$\sum_{s\in\{0,1\}^n} \Theta_{s,s} \le 2^n. \tag{4}$$

Next, fix $s \neq s'$. Sample $G \leftarrow \mathcal{F}_{n,n'}$ and $v, v', y, y' \leftarrow \{0,1\}^{n'}$ independently, and define $G_{s,s'} := G^{s\leftarrow v, s'\leftarrow v'}$. That is, $G_{s,s'}$ is obtained by changing the value of G on both s and s'. Then $G_{s,s'}$ is distributed uniformly over $\mathcal{F}_{n,n'}$. Therefore,

$$\begin{aligned}
\Theta_{s,s'} &= \mathrm{E}_G\left[\left(h^G(G(s)) - \bar{h}_G\right)\left(h^G(G(s')) - \bar{h}_G\right)\right] \\
&= \mathrm{E}_{G,v,v',y,y'}\left[\left(h^{G_{s,s'}}(G_{s,s'}(s)) - \bar{h}_{G_{s,s'}}\right)\left(h^{G_{s,s'}}(G_{s,s'}(s')) - \bar{h}_{G_{s,s'}}\right)\right] \\
&= \mathrm{E}_{G,v,v',y,y'}\left[\left(h^{G_{s,s'}}(v) - \bar{h}_{G_{s,s'}}\right)\left(h^{G_{s,s'}}(v') - \bar{h}_{G_{s,s'}}\right)\right] \\
&= \mathrm{E}_{G,v,v',y,y'}\left[\left(h^{G_{s,s'}}(v) - h^{G_{s,s'}}(y)\right)\left(h^{G_{s,s'}}(v') - h^{G_{s,s'}}(y')\right)\right],
\end{aligned}$$

where the last equality is by independence of y and y'. Now, consider swapping y and v. Then the distributions remain the same; however, the sign of the last equality changes. In more detail, define $\widetilde{G}_{s,s'} := G^{s\leftarrow y, s'\leftarrow v'}$ using the same sampled variables. Then

$$\Theta_{s,s'} = -\mathrm{E}_{G,v,v',y,y'}\left[\left(h^{\widetilde{G}_{s,s'}}(v) - h^{\widetilde{G}_{s,s'}}(y)\right)\left(h^{\widetilde{G}_{s,s'}}(v') - h^{\widetilde{G}_{s,s'}}(y')\right)\right].$$

We abbreviate by letting

$$\begin{aligned}
X &= h^{G_{s,s'}}(v) - h^{G_{s,s'}}(y), & X' &= h^{G_{s,s'}}(v') - h^{G_{s,s'}}(y'), \\
\widetilde{X} &= h^{\widetilde{G}_{s,s'}}(v) - h^{\widetilde{G}_{s,s'}}(y), & \widetilde{X}' &= h^{\widetilde{G}_{s,s'}}(v') - h^{\widetilde{G}_{s,s'}}(y').
\end{aligned}$$

Then

$$\Theta_{s,s'} = \mathrm{E}_{G,v,v',y,y'}\left[X \cdot X'\right] = -\mathrm{E}_{G,v,v',y,y'}\left[\widetilde{X} \cdot \widetilde{X}'\right].$$

Averaging the two representations yields

$$\Theta_{s,s'} = \frac{1}{2} \cdot \mathrm{E}_{G,v,v',y,y'}\left[X \cdot X' - \widetilde{X} \cdot \widetilde{X}'\right].$$

Note that $X, X', \widetilde{X}, \widetilde{X}' \in [-1,1]$. Therefore, the triangle inequality yields

$$\begin{aligned}
|\Theta_{s,s'}| &\leq \frac{1}{2} \cdot \mathrm{E}_{G,v,v',y,y'} \left[\left| X \cdot X' - \widetilde{X} \cdot \widetilde{X}' \right| \right] \\
&\leq \frac{1}{2} \cdot \mathrm{E}_{G,v,v',y,y'} \left[\left| X \cdot \left(X' - \widetilde{X}' \right) \right| + \left| \widetilde{X}' \left(X - \widetilde{X} \right) \right| \right] \\
&\leq \frac{1}{2} \cdot \mathrm{E}_{G,v,v',y,y'} \left[\left| X' - \widetilde{X}' \right| + \left| X - \widetilde{X} \right| \right] \\
&\leq \frac{1}{2} \cdot \mathrm{E}_{G,v,v',y,y'} \Bigg[\left| h^{G_{s,s'}}(v') - h^{G_{s,s'}}(y') - h^{\widetilde{G}_{s,s'}}(v') + h^{\widetilde{G}_{s,s'}}(y') \right| \\
&\qquad + \left| h^{G_{s,s'}}(v) - h^{G_{s,s'}}(y) - h^{\widetilde{G}_{s,s'}}(v) + h^{\widetilde{G}_{s,s'}}(y) \right| \Bigg] \\
&\leq \frac{1}{2} \cdot \mathrm{E}_{G,v,v',y,y'} \Bigg[\left| h^{G_{s,s'}}(v') - h^{\widetilde{G}_{s,s'}}(v') \right| + \left| h^{G_{s,s'}}(y') - h^{\widetilde{G}_{s,s'}}(y') \right| \\
&\qquad + \left| h^{G_{s,s'}}(v) - h^{\widetilde{G}_{s,s'}}(v) \right| + \left| h^{G_{s,s'}}(y) - h^{\widetilde{G}_{s,s'}}(y) \right| \Bigg].
\end{aligned}$$

Note that $G_{s,s'}$ and $\widetilde{G}_{s,s'}$ differ only on s. Let $G_{s'} = G^{s' \leftarrow v'}$. Then $G_{s,s'} = G_{s'}^{s \leftarrow v}$ and $\widetilde{G}_{s,s'} = G_{s'}^{s \leftarrow y}$. By the triangle inequality,

$$\begin{aligned}
&\sum_{s \in \{0,1\}^n \setminus \{s'\}} \left| h^{G_{s,s'}}(w) - h^{\widetilde{G}_{s,s'}}(w) \right| \\
&\leq \sum_{s \in \{0,1\}^n} \left| h^{G_{s'}^{s \leftarrow v}}(w) - h^{G_{s'}}(w) \right| + \sum_{s \in \{0,1\}^n} \left| h^{G_{s'}^{s \leftarrow y}}(w) - h^{G_{s'}}(w) \right| \\
&\leq 2 \cdot 2^n \cdot \sigma(R) \\
&= 2^{n+1} \cdot \sigma(R),
\end{aligned}$$

for all $w \in \{v, v', y, y'\}$ and every $s' \in \{0,1\}^n$. Therefore,

$$\sum_{s \in \{0,1\}^n \setminus \{s'\}} |\Theta_{s,s'}| \leq \frac{1}{2} \cdot 4 \cdot 2^{n+1} \cdot \sigma(R) = 4 \cdot 2^n \cdot \sigma(R)$$

for all $s' \in \{0,1\}^n$. Summing over all s' yields

$$\sum_{s \neq s' \in \{0,1\}^n} \Theta_{s,s'} \leq \sum_{s' \in \{0,1\}^n} \sum_{s \in \{0,1\}^n \setminus \{s'\}} |\Theta_{s,s'}| \leq 4 \cdot 2^{2n} \cdot \sigma(R). \tag{5}$$

Combining Equations (3) to (5) gives

$$\mathrm{E}_{G \leftarrow \mathcal{F}_{n,n'}} \left[\Delta_G^2 \right] \leq \frac{2^n + 4 \cdot 2^{2n} \cdot \sigma(R)}{2^{2n}} = 4 \cdot \sigma(R) + \frac{1}{2^n}.$$

□

We are now ready to prove the main result of this section. We show that for any oracle-aided algorithm $R^{\mathcal{A}^G, G}$ making polynomially many oracle calls in n, if the sensitivity of $\mathcal{A}^G$ is negligible, then there is a function G^* such that the advantage of $R^{\mathcal{A}^{G^*}, G^*}$ is negligible.

Theorem 4.4. *Let $\mathcal{A}^{(\cdot)}$ be an oracle-aided algorithm that on input $\mathbf{z} \in \{0,1\}^m$ and oracle access to a function $G : \{0,1\}^n \to \{0,1\}^{n'}$, outputs 1 with probability $p^G(\mathbf{z})$ and outputs 0 with probability $1-p^G(\mathbf{z})$. Consider an oracle-aided algorithm $R^{(\cdot),(\cdot)}$ that on $y \in \{0,1\}^{n'}$ and 1^n outputs a Boolean value, and define $h^G : \{0,1\}^{n'} \to [0,1]$ as*

$$h^G(y) = \Pr\left[R^{\mathcal{A}^G,G}(1^n, y) = 1\right],$$

where the probability is taken over the reduction's coins and the private coins of $\mathcal{A}^G$ over all calls to it. Let q_G denote the number of calls R makes to G, and let $q_{\mathcal{A}}$ denote the number of calls R makes to $\mathcal{A}^G$.

For every $G : \{0,1\}^n \to \{0,1\}^{n'}$, let

$$\Delta_G = \Pr_{s \leftarrow \{0,1\}^n}\left[R^{\mathcal{A}^G,G}(1^n, G(s)) = 1\right] - \Pr_{y \leftarrow \{0,1\}^{n'}}\left[R^{\mathcal{A}^G,G}(1^n, y) = 1\right].$$

Then there exists $G^* : \{0,1\}^n \to \{0,1\}^{n'}$ such that

$$|\Delta_{G^*}| \leq \sqrt{4 \cdot \left(\frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}}.$$

Proof. First, observe that

$$\Delta_G = \mathrm{E}_{s \leftarrow \{0,1\}^n}\left[h^G(G(s))\right] - \mathrm{E}_{y \leftarrow \{0,1\}^{n'}}\left[h^G(y)\right].$$

Therefore, by Lemma 4.3,

$$\mathrm{E}_{G \leftarrow \mathcal{F}_{n,n'}}\left[\Delta_G^2\right] \leq 4\sigma(R) + \frac{1}{2^n},$$

Lemma 4.2 yields

$$\mathrm{E}_{G \leftarrow \mathcal{F}_{n,n'}}\left[\Delta_G^2\right] \leq 4 \cdot \left(\frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}.$$

Let G^* be the lexicographically first minimizer of Δ_G^2. Then

$$\Delta_{G^*}^2 = \min_{G \in \mathcal{F}_{n,n'}}\left\{\Delta_G^2\right\} \leq \mathrm{E}_{G \leftarrow \mathcal{F}_{n,n'}}\left[\Delta_G^2\right] \leq 4 \cdot \left(\frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}.$$

Thus

$$|\Delta_{G^*}| \leq \sqrt{4 \cdot \left(\frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}}.$$

□

5 Impossibility of Non-Adaptive Constructions of PRFs from PRGs

In this section, we give a warm-up to our main result, stating that the GGM construction is optimal. We use our framework to show that PRFs cannot be constructed non-adaptively from PRGs, using fully black-box constructions.

Theorem 5.1. *For all polynomially bounded $c(n)$ and $r(n)$, there is no $c(n)$-call fully black-box non-adaptive construction of a weak PRF from an $r(n)$-bit stretch PRG.*

Proof. Assume towards contradiction that there exist polynomially bounded $c = c(n)$ and $r = r(n)$ for which there is a fully black-box construction $(\mathcal{F}, R)$ of a weak PRF from an r-bit stretch PRG, using c non-adaptive calls. By adding an unused query when necessary, assume $c \geq 1$. We write

$$\mathcal{F} = \left\{ f_k^{(\cdot)} : \{0,1\}^\ell \to \{0,1\} \right\}_{n \in \mathbb{N}, k \in \{0,1\}^\lambda}$$

where $f_k^{(\cdot)}$ is given by

$$f_k^G(x) = P\left(x, k, G(S_1(x,k)), G(S_2(x,k)), \ldots, G(S_c(x,k))\right),$$

for efficiently computable functions $S_1, S_2, \ldots, S_c : \{0,1\}^\ell \times \{0,1\}^\lambda \to \{0,1\}^n$ and $P : \{0,1\}^\ell \times \{0,1\}^\lambda \times (\{0,1\}^{n+r})^c \to \{0,1\}$, where $\ell = \ell(n)$ and $\lambda = \lambda(n)$ are polynomially bounded.

To obtain a contradiction, we show that there exists an adversary $\mathcal{A}$ and a function $G^* : \{0,1\}^n \to \{0,1\}^{n+r}$ such that on the one hand

$$\left| \Pr_{k \leftarrow \{0,1\}^\lambda} \left[\mathcal{A}^{f_k^{G^*}, G^*}(1^n) = 1 \right] - \Pr_{f \leftarrow \mathcal{F}_\ell} \left[\mathcal{A}^{f, G^*}(1^n) = 1 \right] \right| \geq \frac{1}{n^\alpha}$$

for some constant α and all sufficiently large n's; on the other hand,

$$\left| \Pr_{s \leftarrow \{0,1\}^n} \left[R^{\mathcal{A}^{G^*}, G^*}(1^n, G^*(s)) = 1 \right] - \Pr_{y \leftarrow \{0,1\}^{n+r(n)}} \left[R^{\mathcal{A}^{G^*}, G^*}(1^n, y) = 1 \right] \right| = \mathrm{neg}(n).$$

The key idea is to construct a "soft" consistency test. Rather than asking whether a transcript is exactly consistent with some key, we measure how many values of G must be changed to make the transcript consistent. This makes the adversary stable when one value of G changes. As a result, the G-sensitivity of the adversary is small, allowing us to invoke Theorem 4.4.

Notations and setup. We first introduce some notations. Let $m = m(n) \leq 2^\ell$ be a polynomial to be chosen by the analysis below, and fix distinct inputs $x_1, \ldots, x_m \in \{0,1\}^\ell$. For every key $k \in \{0,1\}^\lambda$, let

$$\mathcal{T}_k = \{S_j(x_i, k) : i \in [m], j \in [c]\}$$

denote the set of all seeds required to compute the PRF on all inputs using the key k. Further denote $t_k = |\mathcal{T}_k| \leq cm$. Next, for every assignment of possible PRG outputs $Y : \mathcal{T}_k \to \{0,1\}^{n+r}$, let

$$\Phi_k(Y) = \left(P\left(x_i, k, Y\left(S_1(x_i,k)\right), \ldots, Y\left(S_c(x_i,k)\right)\right)\right)_{i \in [m]}$$

denote the corresponding m PRF outputs.

For m possible PRF outputs $\mathbf{z} \in \{0,1\}^m$, let

$$\Phi_k^{-1}(\mathbf{z}) = \left\{ Y : \mathcal{T}_k \to \{0,1\}^{n+r} \mid \Phi_k(Y) = \mathbf{z} \right\}.$$

For a function $G : \{0,1\}^n \to \{0,1\}^{n+r}$, define

$$\mathsf{dist}_k^G(\mathbf{z}) = \min_{Y \in \Phi_k^{-1}(\mathbf{z})} \mathsf{dist}\left(Y, G|_{\mathcal{T}_k}\right)$$

to be the minimum number of changes to G needed for the PRF to output the results in $\mathbf{z}$. Stated differently, $\mathsf{dist}_k^G(\mathbf{z})$ is the Hamming distance between $G|_{\mathcal{T}_k}$ and the set of functions $G' : \mathcal{T}_k \to$

$\{0,1\}^{n+r}$ that are consistent with the output $\mathbf{z}$, where the alphabet is the set $\{0,1\}^{n+r}$. Note that it could be the case where no changes to G result in the output $\mathbf{z}$. In this case we let the distance be ∞. Next, we define the weight of $\mathbf{z}$ as

$$w_k^G(\mathbf{z}) = 2^{-\mathsf{dist}_k^G(\mathbf{z})},$$

where we let $2^{-\infty} = 0$. Intuitively, the weight $w_k^G(\mathbf{z})$ gives a "soft" measure of how plausible a transcript $\mathbf{z}$ is under the key k. If $\mathbf{z}$ is generated by the PRF construction with key k, then $\mathsf{dist}_k^G(\mathbf{z}) = 0$ and $w_k^G(\mathbf{z}) = 1$. If obtaining $\mathbf{z}$ requires changing many PRG values, its weight is exponentially small. Finally, denote

$$W^G(\mathbf{z}) = \sum_{k\in\{0,1\}^\lambda} w_k^G(\mathbf{z}), \qquad p^G(\mathbf{z}) = \min\left\{1, W^G(\mathbf{z})\right\}.$$

We next define the adversary $\mathcal{A}^{f,G}$. It simply queries the function f at $x_1,\ldots,x_m$,[8] and outputs 1 with probability $p^G(\mathbf{z})$ and outputs 0 with probability $1-p^G(\mathbf{z})$, where $\mathbf{z}\in\{0,1\}^m$ is the vector of answers.

Algorithm 5.2 ($\mathcal{A}^{(\cdot),(\cdot)}$)**.**
Input: The security parameter 1^n.
Oracle access: A function $f:\{0,1\}^\ell\to\{0,1\}$ *and a (purported) PRG* $G:\{0,1\}^n\to\{0,1\}^{n+r}$.

1. *Query* f *on* $x_1,\ldots,x_m$. *Let* $\mathbf{z}\in\{0,1\}^m$ *denote the vector of corresponding answers.*
2. *Output 1 with probability* $p^G(\mathbf{z})$.

We first show that $\mathcal{A}^{(\cdot),(\cdot)}$ successfully distinguishes the PRF $\mathcal{F}$ from a random function. Intuitively, in the genuine PRF experiment, the observed transcript is perfectly explained by the sampled key and the actual PRG G. Hence the corresponding weight is exactly 1, so the adversary always accepts. For a random function, however, the transcript is a uniform m-bit string. For a fixed key k, only a small number of transcripts can be obtained after changing fewer than a fixed amount, say τ, of PRG values. This number of entries is bounded by the volume of the cm-dimensional Hamming ball of radius $\tau-1$, i.e., $\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(cm,\tau-1)$. Every other transcript receives an exponentially small weight. Since there are only 2^λ keys, the total acceptance probability is negligible.

Claim 5.3. *For all* $G:\{0,1\}^n\to\{0,1\}^{n+r}$ *and* $\tau\in[m]$,

$$\left|\Pr_{k\leftarrow\{0,1\}^\lambda}\left[\mathcal{A}^{f_k^G,G}(1^n)=1\right]-\Pr_{f\leftarrow\mathcal{F}_\ell}\left[\mathcal{A}^{f,G}(1^n)=1\right]\right| \geq 1-2^\lambda\cdot\left(\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(cm,\tau-1)}{2^m}+2^{-\tau}\right).$$

Consequently, if $\tau\geq\lambda+n$ *and* $m\geq 3(2n+r)\cdot\tau+2\lambda$ *are polynomials, then the advantage is at least* $1-2^{-n+1}$ *for all sufficiently large* n*'s.*

[8] This is the fixed-input test. For weak PRFs, at each invocation independently sample $x_1,\ldots,x_m$ uniformly from $\{0,1\}^\ell$, issue all m queries, and output 0 if an input repeats; otherwise use this test with the sampled list. The probability of a repetition is at most $\binom{m}{2}2^{-\ell}$, which is negligible for polynomial m for $\ell=\omega(\log n)$.

Proof. We first show that

$$\Pr_{k\leftarrow\{0,1\}^\lambda}\left[\mathcal{A}^{f_k^G,G}(1^n)=1\right]=1.$$

Indeed, if the key that was sampled is k, then the function $Y:\mathcal{T}_k\to\{0,1\}^{n+r}$ defined as $Y(s)=G(s)$ results in $\Phi_k(Y)=\mathbf{z}$. Therefore, $\mathsf{dist}_k^G=0$, hence $w_k^G(\mathbf{z})=1$, implying that $p^G(\mathbf{z})=1$.

We next analyze the probability that the adversary answers 1 when interacting with a random function. Let $\mathbf{Z}$ be the random variable that is the vector of outputs the adversary receives from f. Since the inputs $x_1,\ldots,x_m$ are distinct, $\mathbf{Z}$ is distributed uniformly over $\{0,1\}^m$. Then by linearity of expectation,

$$\begin{aligned}\Pr_{f\leftarrow\mathcal{F}_\ell}\left[\mathcal{A}^{f,G}(1^n)=1\right]&=\mathrm{E}_{\mathbf{Z}}\left[p^G(\mathbf{Z})\right]\\&=\mathrm{E}_{\mathbf{Z}}\left[\min\left\{1,W^G(\mathbf{Z})\right\}\right]\\&\le\mathrm{E}_{\mathbf{Z}}\left[\sum_{k\in\{0,1\}^\lambda}w_k^G(\mathbf{Z})\right]\\&=\sum_{k\in\{0,1\}^\lambda}\mathrm{E}_{\mathbf{Z}}\left[w_k^G(\mathbf{Z})\right].\end{aligned}$$

We next bound the expected weight for every key k. The idea is to split according to whether $\mathsf{dist}_k^G<\tau$ or not.

For a key $k\in\{0,1\}^\lambda$, consider the set

$$\mathcal{N}_k^G=\left\{\mathbf{z}\in\{0,1\}^m:\mathsf{dist}_k^G(\mathbf{z})<\tau\right\}$$

of all possible PRF outputs that are consistent with changing G in less than τ positions. By the definition of $\mathsf{dist}_k^G(\mathbf{z})$, every $\mathbf{z}\in\mathcal{N}_k^G$ is induced by at least one assignment $Y:\mathcal{T}_k\to\{0,1\}^{n+r}$ that is of distance less than τ away from $G|_{\mathcal{T}_k}$.

The number of such assignments is $\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(t_k,\tau-1)$. Therefore,

$$|\mathcal{N}_k^G|\le\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(t_k,\tau-1)\le\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(cm,\tau-1).$$

Now, for every $\mathbf{z}\in\mathcal{N}_k^G$, its weight is at most 1, and for every other $\mathbf{z}$, its weight is at most $2^{-\tau}$. Thus,

$$\begin{aligned}\mathrm{E}_{\mathbf{Z}}\left[w_k^G(\mathbf{Z})\right]&=\sum_{\mathbf{z}\in\mathcal{N}_k^G}\frac{1}{2^m}\cdot w_k^G(\mathbf{z})+\sum_{\mathbf{z}\in\{0,1\}^m\setminus\mathcal{N}_k^G}\frac{1}{2^m}\cdot w_k^G(\mathbf{z})\\&\le\frac{|\mathcal{N}_k^G|}{2^m}+2^{-\tau}\\&\le\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(cm,\tau-1)}{2^m}+2^{-\tau}.\end{aligned}\tag{6}$$

Therefore,

$$\Pr_{f\leftarrow\mathcal{F}_\ell}\left[\mathcal{A}^{f,G}(1^n)=1\right]\le\sum_{k\in\{0,1\}^\lambda}\mathrm{E}_{\mathbf{Z}}\left[w_k^G(\mathbf{Z})\right]\le2^\lambda\cdot\left(\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(cm,\tau-1)}{2^m}+2^{-\tau}\right).$$

Finally, note that if $\tau \geq \lambda + n$ and $m \geq 3(2n + r) \cdot \tau + 2\lambda$ are polynomials, then by Fact 3.3,

$$2^{\lambda} \cdot \left(\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(cm, \tau - 1)}{2^m} + 2^{-\tau} \right) \leq 2^{\lambda - m + (n+r+\log(cm))(\tau-1) + \log \tau} + 2^{\lambda - \tau}.$$

Since $\log(cm), \log \tau < n$ for all sufficiently large n's,

$$(n + r + \log(cm))(\tau - 1) + \log \tau \leq m/2$$

for all sufficiently large n's. Therefore,

$$2^{\lambda - m + (n+r+\log(cm))(\tau-1) + \log \tau} + 2^{\lambda - \tau} \leq 2^{\lambda - m/2} + 2^{\lambda - \tau} \leq 2^{-n+1}.$$

□

The next step of the proof is to bound the G-sensitivity of the adversary. Specifically, we show that it is bounded by $cm/2^n$. Intuitively, this follows from the following observation. Change the value of G at one seed s. For a fixed key k, nothing changes if $s \notin \mathcal{T}_k$. If $s \in \mathcal{T}_k$, then the minimum Hamming distance can change by at most 1, so the corresponding weight $w_k^G(\mathbf{z}) = 2^{-\mathsf{dist}_k^G(\mathbf{z})}$ changes by at most a factor of 2. The total effect of changing one seed is therefore proportional to the total weight carried by keys whose computation actually uses that seed. Summing over all seeds counts each key-weight once for every seed in $\mathcal{T}_k$, which is of size at most cm. Thus the total sensitivity is at most $cm/2^n$.

Lemma 5.4. $\sigma^G(\mathcal{A}) \leq cm/2^n$.

Proof. Fix a function $G : \{0,1\}^n \to \{0,1\}^{n+r}$, a vector of values $\mathbf{v} = (v_s)_{s \in \{0,1\}^n}$, where $v_s \in \{0,1\}^{n+r}$ for every $s \in \{0,1\}^n$, and a vector of possible PRF outputs $\mathbf{z} \in \{0,1\}^m$. We show that $\sigma^G(\mathcal{A}, \mathbf{v}, \mathbf{z}) \leq cm/2^n$. First, if $\Phi_k^{-1}(\mathbf{z}) = \emptyset$ then $\mathsf{dist}_k^G(\mathbf{z}) = \infty$. Thus, $w_k^G(\mathbf{z}) = 0$, implying that the key does not contribute to $\sigma^G(\mathcal{A}, \mathbf{v}, \mathbf{z})$.

We now restrict ourselves to keys k such that $\Phi_k^{-1}(\mathbf{z}) \neq \emptyset$. We first bound

$$\left| \mathsf{dist}_k^G(\mathbf{z}) - \mathsf{dist}_k^{G^{s \leftarrow v_s}}(\mathbf{z}) \right|$$

for every $s \in \{0,1\}^n$. Note that if $s \in \mathcal{T}_k$, then replacing $G(s)$ with v_s changes $\mathsf{dist}_k^G(\mathbf{z})$ by at most 1. Therefore,

$$\left| \mathsf{dist}_k^G(\mathbf{z}) - \mathsf{dist}_k^{G^{s \leftarrow v_s}}(\mathbf{z}) \right| \leq 1.$$

On the other hand, if $s \notin \mathcal{T}_k$, then replacing $G(s)$ with another value does not change $\mathsf{dist}_k^G(\mathbf{z})$. Therefore, the difference is exactly 0 in this case. As a result, for every $s \in \mathcal{T}_k$,

$$\frac{1}{2} \cdot w_k^G(\mathbf{z}) \leq w_k^{G^{s \leftarrow v_s}}(\mathbf{z}) \leq 2 \cdot w_k^G(\mathbf{z}),$$

and for every $s \notin \mathcal{T}_k$, $w_k^G(\mathbf{z}) = w_k^{G^{s \leftarrow v_s}}(\mathbf{z})$. It follows that

$$\left| w_k^{G^{s \leftarrow v_s}}(\mathbf{z}) - w_k^G(\mathbf{z}) \right| \leq w_k^G(\mathbf{z}) \tag{7}$$

for all $s \in \mathcal{T}_k$.

Let

$$G_s = G^{s \leftarrow v_s}, \qquad W = W^G(\mathbf{z}), \qquad W_s = W^{G_s}(\mathbf{z}), \qquad A_s = \sum_{\substack{k \in \{0,1\}^\lambda: \\ s \in \mathcal{T}_k}} w_k^G(\mathbf{z}).$$

By the triangle inequality, summing Equation (7) over all affected keys k gives

$$|W_s - W| \leq A_s. \tag{8}$$

Define $\phi(u) = \min\{1, u\}$. Then $p^G(\mathbf{z}) = \phi(W)$ and $p^{G^{s \leftarrow v_s}}(\mathbf{z}) = \phi(W_s)$. By Fact 3.1 with $a = W_s$ and $b = W$,

$$\left| p^{G_s}(\mathbf{z}) - p^G(\mathbf{z}) \right| = |\phi(W_s) - \phi(W)| \leq \frac{|W_s - W|}{\max\{1, W\}} \tag{9}$$

for all $s \in \{0,1\}^n$. Applying Equations (8) and (9) yields

$$\left| p^{G_s}(\mathbf{z}) - p^G(\mathbf{z}) \right| \leq \frac{A_s}{\max\{1, W\}}.$$

Moreover,

$$\sum_{s \in \{0,1\}^n} A_s = \sum_{s \in \{0,1\}^n} \sum_{\substack{k \in \{0,1\}^\lambda: \\ s \in \mathcal{T}_k}} w_k^G(\mathbf{z}) = \sum_{k \in \{0,1\}^\lambda} \sum_{s \in \mathcal{T}_k} w_k^G(\mathbf{z}) = \sum_{k \in \{0,1\}^\lambda} t_k \cdot w_k^G(\mathbf{z}) \leq cm \cdot W.$$

Therefore,

$$\sum_{s \in \{0,1\}^n} |\phi(W) - \phi(W_s)| \leq \frac{\sum_{s \in \{0,1\}^n} A_s}{\max\{1, W\}} \leq \frac{cm \cdot W}{\max\{1, W\}} \leq cm.$$

Averaging over $s \in \{0,1\}^n$ and maximizing over $\mathbf{v}$ and $\mathbf{z}$ yields $\sigma^G(\mathcal{A}) \leq cm/2^n$, as claimed. □

Finally, for every $G : \{0,1\}^n \to \{0,1\}^{n+r}$, let

$$\Delta_G = \Pr_{s \leftarrow \{0,1\}^n}\left[R^{\mathcal{A}^G, G}(1^n, G(s)) = 1\right] - \Pr_{y \leftarrow \{0,1\}^{n+r(n)}}\left[R^{\mathcal{A}^G, G}(1^n, y) = 1\right]$$

be the reduction's advantage. Additionally, let $q_G = q_G(n, m)$ denote the number of calls R makes to G, and let $q_{\mathcal{A}} = q_{\mathcal{A}}(n, m)$ denote the number of calls R makes to $\mathcal{A}^G$. Then by Theorem 4.4, there exists $G^* : \{0,1\}^n \to \{0,1\}^{n+r}$ such that

$$|\Delta_{G^*}| \leq \sqrt{\frac{4 \cdot (q_G + q_{\mathcal{A}} \cdot cm) + 1}{2^n}}.$$

Set $m = 3(2n + r) \cdot \tau + 2\lambda$, where $\tau = \lambda + n$, as in Claim 5.3. Then on the one hand, $\mathcal{A}$ has advantage at least $1 - 2^{-n+1}$. On the other hand, the reduction's advantage is $|\Delta_{G^*}| = \mathrm{neg}(n)$. Finally, recall that $\ell = \omega(\log n)$, hence, $m \leq 2^\ell$ for all sufficiently large n's, implying there exist m distinct inputs $x_1, \ldots, x_m$. This completes the proof of Theorem 5.1. □

6 Lower Bounds for Adaptive Constructions of PRFs from PRGs

In this section, we prove our main result. We show that to construct a PRF from an r-bit stretch PRG using a fully black-box construction, the GGM construction combined with Levin's technique is optimal whenever $r = \Omega(n)$.

Theorem 6.1. *For all polynomially bounded positive integer-valued functions $d(n)$ and $r(n)$, such that*

$$d(n) \log\left(2 + \frac{r(n)}{n}\right) = O(\log n),$$

there is no fully black-box d-round construction of a weak PRF from an $r(n)$-bit stretch PRG.

Proof. Assume towards contradiction that there exist polynomially bounded positive integer-valued functions $d = d(n)$ and $r = r(n)$ for which there is a fully black-box d-round construction $(\mathcal{F}, R)$ of a weak PRF from an r-bit stretch PRG. We write

$$\mathcal{F} = \left\{ f_k^{(\cdot)} : \{0,1\}^\ell \to \{0,1\} \right\}_{n \in \mathbb{N}, k \in \{0,1\}^\lambda},$$

where $\ell = \ell(n)$ and $\lambda = \lambda(n)$ are polynomially bounded, and where an evaluation of $f_k^G(x)$ is arranged in $d = d(n)$ adaptive rounds. Empty rounds may be padded with one unused query, so we assume $c_i \geq 1$. In round $i \in [d]$, the construction makes $c_i = c_i(n)$ non-adaptive calls (given the key, the input and the previous answers), and we let

$$C = C(n) = \sum_{i=1}^{d} c_i = \mathrm{poly}(n).$$

We denote the functions for generating the seeds in round i by $S_{i,1}, \ldots, S_{i,c_i}$. These functions receive as input the PRF input $x \in \{0,1\}^\ell$, the key $k \in \{0,1\}^\lambda$, and all previous PRG answers. Their output is a seed in $\{0,1\}^n$.

To obtain a contradiction, we show that there exists an adversary $\mathcal{A}$ and a function $G^* : \{0,1\}^n \to \{0,1\}^{n+r}$ such that on the one hand

$$\left| \Pr_{k \leftarrow \{0,1\}^\lambda} \left[\mathcal{A}^{f_k^{G^*}, G^*}(1^n) = 1 \right] - \Pr_{f \leftarrow \mathcal{F}_\ell} \left[\mathcal{A}^{f, G^*}(1^n) = 1 \right] \right| \geq \frac{1}{n^\kappa}$$

for some constant κ and all sufficiently large n's; on the other hand,

$$\left| \Pr_{s \leftarrow \{0,1\}^n} \left[R^{\mathcal{A}^{G^*}, G^*}(1^n, G^*(s)) = 1 \right] - \Pr_{y \leftarrow \{0,1\}^{n+r(n)}} \left[R^{\mathcal{A}^{G^*}, G^*}(1^n, y) = 1 \right] \right| = \mathrm{neg}(n).$$

For m distinct inputs $x_1, \ldots, x_m$, it will be convenient to consider the tree of *all possible computations* of the PRF on the m inputs and a key k: For every key $k \in \{0,1\}^\lambda$, label every node u by the set of seeds $\mathcal{T}_{u,k}$ the computation sends to G at that stage of the computation, and every outgoing edge from u to a child v is labeled by a possible assignment $Y_{v,k} : \mathcal{T}_{u,k} \to \{0,1\}^{n+r}$.[9] The leaves correspond to the last set of queries made by the construction. We then give each node a weight. We start with the leaves of the tree and go backwards; the weight of the root is defined to be the weight of the key.

[9] Note that it could be the case that a seed appears in two different nodes on the same path, yet the two assignments given by the outgoing edges do not agree. We show that this does not hurt the argument.

Notations and tree setup. We first introduce some notations. Fix $\gamma = \gamma(n) = \omega(\log n)$ with $\gamma = o(n)$, and let $\eta = 2^{-(n+r+\gamma)}$, $0 < \alpha = \alpha(n) < 1/2$, and $m = m(n) \leq 2^{\ell}$ be polynomially bounded, with the parameters chosen by the analysis below. Further fix m distinct inputs $\mathbf{x} = (x_1, \ldots, x_m) \in (\{0,1\}^{\ell})^m$. For every key $k \in \{0,1\}^{\lambda}$, we define the following. Let

$$\mathcal{T}_{1,k} = \{S_{1,j}(x_i, k) : i \in [m], j \in [c_1]\}$$

denote the set of all seeds required to compute the first round of the PRF computation on all inputs using the key k. Thus, $\mathcal{T}_{1,k}$ corresponds to the root of the tree of all possible PRF evaluations. Further denote $t_{1,k} = |\mathcal{T}_{1,k}| \leq c_1 m$. A possible first-round assignment is a function $Y_{1,k} : \mathcal{T}_{1,k} \to \{0,1\}^{n+r}$ that assigns possible PRG outputs for all seeds for the first round of computation. Each $Y_{1,k}$ corresponds to an outgoing edge from the root.

We define these values for subsequent rounds recursively. For every $i \in [d]$ and fixed possible assignments $Y_{1,k}, \ldots, Y_{i-1,k}$ of prior rounds, let:

- $\mathbf{Y}_{i-1,k} = (Y_{1,k}, \ldots, Y_{i-1,k})$ be the path from the root,
- $\mathcal{T}_{i,k}(\mathbf{Y}_{i-1,k}) = \{S_{i,j}(x_{i'}, k) : i' \in [m], j \in [c_i]\}$ be the set of all seeds required to compute the ith round of the PRF computation on all inputs using the key k, i.e., the next node in the tree,
- $t_{i,k}(\mathbf{Y}_{i-1,k}) = |\mathcal{T}_{i,k}(\mathbf{Y}_{i-1,k})| \leq c_i m$,
- $Y_{i,k}(\mathbf{Y}_{i-1,k}) : \mathcal{T}_{i,k}(\mathbf{Y}_{i-1,k}) \to \{0,1\}^{n+r}$ denote possible assignments for round i, i.e., an outgoing edge from $\mathcal{T}_{i,k}(\mathbf{Y}_{i-1,k})$.[10]

When $\mathbf{Y}_{i-1,k}$ is clear from context, we remove it to alleviate notations. In the context of the tree of all possible computations, $\mathcal{T}_{i,k}$ corresponds to nodes in level i, $Y_{i,k}$ corresponds to edges from level i to $i+1$, and $\mathbf{Y}_{i,k}$ corresponds to a path with i edges. As such, we will sometimes refer to these values using their corresponding tree terminology.

Assigning weights. Our next goal is to define the weight of the key k for a given vector of outputs $\mathbf{z} \in \{0,1\}^m$. This is done recursively, starting from the leaves and going up the tree towards its root.

Consider the last round. For every vector of assignments $\mathbf{Y}_{d,k}$, we let $\Phi_k(\mathbf{Y}_{d,k}) \in \{0,1\}^m$ be the vector of outputs of the PRF on all m inputs. Define its inverse as

$$\Phi_k^{-1}(\mathbf{Y}_{d-1,k}, \mathbf{z}) = \left\{ Y_{d,k} : \mathcal{T}_{d,k} \to \{0,1\}^{n+r} \mid \Phi_k(\mathbf{Y}_{d,k}) = \mathbf{z} \right\}.$$

For a fixed prefix $\mathbf{Y}_{d-1,k}$ and output $\mathbf{z}$, define the last-round distance as the minimum number of changes to $G|_{\mathcal{T}_{d,k}}$ needed to obtain $\mathbf{z}$ while keeping the earlier hypothetical answers fixed, i.e.,

$$\mathsf{dist}_k^G(\mathbf{Y}_{d-1,k};\ \mathbf{z}) = \min_{Y_{d,k} \in \Phi_k^{-1}(\mathbf{Y}_{d-1,k}, \mathbf{z})} \mathsf{dist}\left(Y_{d,k}, G|_{\mathcal{T}_{d,k}}\right).$$

This is a distance between last-round assignments, not necessarily an edit distance to a single globally consistent oracle: a prefix may assign different answers to a repeated seed in different rounds. If no last-round assignment in $\Phi_k^{-1}(\mathbf{Y}_{d-1,k}, \mathbf{z})$ exists, we let the distance be ∞.

[10] When $i = d$, $\mathcal{T}_{d,k}(\mathbf{Y}_{d-1,k})$ corresponds to a leaf, and so technically this is not an edge of the tree.

We can now recursively define the weights. Starting with the leaves, let

$$w^G_{d,k}\left(\mathbf{Y}_{d-1,k}, \mathbf{z}\right) = 2^{-\mathsf{dist}^G_k\left(\mathbf{Y}_{d-1,k};\ \mathbf{z}\right)},$$

where we let $2^{-\infty} = 0$. For $i = d-1, d-2, \ldots, 1$ and $\mathbf{Y}_{i-1,k}$, define

$$w^G_{i,k}\left(\mathbf{Y}_{i-1,k}, \mathbf{z}\right) = \left(\sum_{Y_{i,k}:\mathcal{T}_{i,k}\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}\left(Y_{i,k},G|_{\mathcal{T}_{i,k}}\right)} \cdot w^G_{i+1,k}\left(\mathbf{Y}_{i,k}, \mathbf{z}\right)\right)^{\alpha}.$$

The weight of the key k is

$$w^G_k(\mathbf{z}) = w^G_{1,k}(\mathbf{z}).$$

Finally, we define

$$W^G(\mathbf{z}) = \sum_{k\in\{0,1\}^\lambda} w^G_k(\mathbf{z}), \qquad p^G(\mathbf{z}) = \min\left\{1, W^G(\mathbf{z})\right\}.$$

Whenever k is clear from context, we suppress it from notations for clarity.

We next define the adversary $\mathcal{A}^{f,G}$. It simply queries the function f at $x_1, \ldots, x_m$,[11] and outputs 1 with probability $p^G(\mathbf{z})$ and outputs 0 with probability $1 - p^G(\mathbf{z})$, where $\mathbf{z} \in \{0,1\}^m$ is the vector of answers.

Algorithm 6.2 ($\mathcal{A}^{(\cdot),(\cdot)}$)**.**
Input: The security parameter 1^n.
Oracle access: A function $f : \{0,1\}^\ell \to \{0,1\}$ *and a (purported) PRG* $G : \{0,1\}^n \to \{0,1\}^{n+r}$.

1. *Query* f *on* $x_1, \ldots, x_m$. *Let* $\mathbf{z} \in \{0,1\}^m$ *denote the vector of corresponding answers.*
2. *Output 1 with probability* $p^G(\mathbf{z})$.

Advantage analysis of $\mathcal{A}$. We first show that $\mathcal{A}^{(\cdot),(\cdot)}$ successfully distinguishes the PRF $\mathcal{F}$ from a random function. That is, we prove the following lemma.

Lemma 6.3. *For all* $G : \{0,1\}^n \to \{0,1\}^{n+r}$ *and* $\tau \in [m]$,

$$\begin{aligned}&\left|\Pr_{k\leftarrow\{0,1\}^\lambda}\left[\mathcal{A}^{f^G_k,G}(1^n) = 1\right] - \Pr_{f\leftarrow\mathcal{F}_\ell}\left[\mathcal{A}^{f,G}(1^n) = 1\right]\right| \\ &\qquad \geq 1 - \left(2^{\lambda+1}\cdot\left(\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c_d m, \tau - 1)}{2^m} + 2^{-\tau}\right)^{\alpha^{d-1}}\right),\end{aligned}$$

for all sufficiently large n's. Consequently, if $\tau \geq (\lambda + n + 1)/\alpha^{d-1}$ *and* $m \geq 3(2n + r)\cdot\tau$ *are polynomials, then the advantage is at least* $1 - 2^{-n+1}$ *for all sufficiently large n's.*

Towards proving the lemma, we first prove two claims about the weight aggregations in the tree. The first claim bounds the total penalty being given for every internal node.

[11] This is the fixed-input adversary that is easily extended to weak PRFs as in the non-adaptive case.

Claim 6.4. *For every key $k \in \{0,1\}^\lambda$, every level $i \in [d]$, and every path $\mathbf{Y}_{i-1,k}$,*

$$\sum_{Y_{i,k}:\mathcal{T}_{i,k}\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}\left(Y_{i,k},G|_{\mathcal{T}_{i,k}}\right)} \leq 2$$

for all sufficiently large n's.

Proof. For every seed $s \in \mathcal{T}_{i,k}$, the assignment $Y_{i,k}$ can either keep $G(s)$ or change its values. The former has weight 1, while the latter contributes weight η for each of the $2^{n+r}-1$ possible other values. Therefore,

$$\begin{aligned}&\sum_{Y_{i,k}:\mathcal{T}_{i,k}\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}\left(Y_{i,k},G|_{\mathcal{T}_{i,k}}\right)} = \left(1+\left(2^{n+r}-1\right)\cdot\eta\right)^{t_{i,k}}\\ &\leq \left(1+\left(2^{n+r}-1\right)\cdot\eta\right)^{c_i m} \leq \left(1+2^{-\gamma}\right)^{c_i m} \leq 2,\end{aligned}$$

where the last inequality holds for all sufficiently large n's since $\gamma = \omega(\log n)$ and $c_i m = \mathrm{poly}(n)$. □

We next bound the maximum expected weight a node can be given when the oracle output is chosen uniformly at random.

Claim 6.5. *For every key $k \in \{0,1\}^\lambda$ and every level $i \in [d]$, let*

$$\varepsilon_{i,k} = \max_{\mathbf{Y}_{i-1,k}} \left\{ \mathrm{E}_{\mathbf{Z}} \left[w^G_{i,k}\left(\mathbf{Y}_{i-1,k}, \mathbf{Z}\right)\right]\right\}$$

denote the largest expected value at level i for the fixed key k, where $\mathbf{Z} \leftarrow \{0,1\}^m$ is uniform. Then for every $k \in \{0,1\}^\lambda$ and $i \in [d-1]$,

$$\varepsilon_{i,k} \leq \left(2\varepsilon_{i+1,k}\right)^\alpha,$$

for all sufficiently large n's. Consequently,

$$\varepsilon_{1,k} \leq 2\cdot\varepsilon_{d,k}^{\alpha^{d-1}},$$

for all sufficiently large n's.

Proof. By Jensen's inequality and the concavity of the mapping $u \mapsto u^\alpha$ (recall that $\alpha < 1$),

$$\begin{aligned}\varepsilon_{i,k} &= \max_{\mathbf{Y}_{i-1,k}} \left\{ \mathrm{E}_{\mathbf{Z}} \left[w^G_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{Z})\right]\right\}\\ &= \max_{\mathbf{Y}_{i-1,k}} \left\{ \mathrm{E}_{\mathbf{Z}} \left[\left(\sum_{Y_{i,k}:\mathcal{T}_{i,k}\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}\left(Y_{i,k},G|_{\mathcal{T}_{i,k}}\right)} \cdot w^G_{i+1,k}\left(\mathbf{Y}_{i,k}, \mathbf{Z}\right)\right)^\alpha\right]\right\}\\ &\leq \max_{\mathbf{Y}_{i-1,k}} \left\{ \left(\sum_{Y_{i,k}:\mathcal{T}_{i,k}\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}\left(Y_{i,k},G|_{\mathcal{T}_{i,k}}\right)} \cdot \mathrm{E}_{\mathbf{Z}} \left[w^G_{i+1,k}\left(\mathbf{Y}_{i,k}, \mathbf{Z}\right)\right]\right)^\alpha\right\}\\ &\leq \max_{\mathbf{Y}_{i-1,k}} \left\{ \left(\sum_{Y_{i,k}:\mathcal{T}_{i,k}\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}\left(Y_{i,k},G|_{\mathcal{T}_{i,k}}\right)} \right)^\alpha \cdot \varepsilon^\alpha_{i+1,k}\right\}\\ &\leq \left(2\varepsilon_{i+1,k}\right)^\alpha,\end{aligned}$$

where the last inequality follows from Claim 6.4 for all sufficiently large n's.

Applying the relation inductively yields

$$\varepsilon_{1,k} \le 2^{\sum_{i=1}^{d-1} \alpha^i} \cdot \varepsilon_{d,k}^{\alpha^{d-1}} = 2^{\frac{\alpha\left(1-\alpha^{d-1}\right)}{1-\alpha}} \cdot \varepsilon_{d,k}^{\alpha^{d-1}}.$$

Recall that $\alpha < 1/2$. Therefore, $\varepsilon_{1,k} \le 2 \cdot \varepsilon_{d,k}^{\alpha^{d-1}}$. □

We are now ready to prove the adversary successfully distinguishes the construction from a random function.

Proof of Lemma 6.3. We first show that

$$\Pr_{k \leftarrow \{0,1\}^\lambda}\left[\mathcal{A}^{f_k^G, G}(1^n) = 1\right] = 1.$$

Indeed, for every key $k \in \{0,1\}^\lambda$, at every round $i \in [d]$ choose the function $Y_{i,k} : \mathcal{T}_{i,k} \to \{0,1\}^{n+r}$ as $Y_{i,k}(s) = G(s)$ for all $s \in \mathcal{T}_{i,k}$. This results in $\Phi_k(\mathbf{Y}_{d,k}) = \mathbf{z}$. Therefore, $\mathsf{dist}_k^G(\mathbf{Y}_{d-1,k};\ \mathbf{z}) = 0$, hence $w_{d,k}^G(\mathbf{z}) = 1$. Moreover, $\mathsf{dist}(Y_{i,k}, G|_{\mathcal{T}_{i,k}}) = 0$ for all $i \in [d]$, hence $w_{i,k}^G(\mathbf{z}) \ge 1$ for all $i \in [d-1]$, implying that $p^G(\mathbf{z}) = 1$.

We next analyze the probability that the adversary answers 1 when interacting with a random function. Let $\mathbf{Z}$ be the random variable that is the vector of outputs the adversary receives from f. Since the inputs $x_1, \ldots, x_m$ are distinct, $\mathbf{Z}$ is distributed uniformly over $\{0,1\}^m$. Then by linearity of expectation,

$$\begin{aligned}
\Pr_{f \leftarrow \mathcal{F}_\ell}\left[\mathcal{A}^{f,G}(1^n) = 1\right] &= \mathrm{E}_{\mathbf{Z}}\left[p^G(\mathbf{Z})\right] \\
&= \mathrm{E}_{\mathbf{Z}}\left[\min\left\{1, W^G(\mathbf{Z})\right\}\right] \\
&\le \mathrm{E}_{\mathbf{Z}}\left[\sum_{k \in \{0,1\}^\lambda} w_k^G(\mathbf{Z})\right] \\
&= \sum_{k \in \{0,1\}^\lambda} \mathrm{E}_{\mathbf{Z}}\left[w_k^G(\mathbf{Z})\right].
\end{aligned}$$

We next bound the expected weight for every key k. Let $\varepsilon_{i,k}$ be as in Claim 6.5. Then it yields,

$$\varepsilon_{1,k} \le 2 \cdot \varepsilon_{d,k}^{\alpha^{d-1}} \tag{10}$$

for every key $k \in \{0,1\}^\lambda$ and all sufficiently large n's.

We now bound $\varepsilon_{d,k}$ as in the non-adaptive case. Fix a key $k \in \{0,1\}^\lambda$ and a path $\mathbf{Y}_{d-1,k}$. The last-round query set $\mathcal{T}_{d,k}$ is then fixed, with $t_{d,k} \le c_d m$, and $w_{d,k}^G(\mathbf{Y}_{d-1,k}, \mathbf{z})$ is the corresponding non-adaptive weight. The per-key bound Equation (6) in the proof of Claim 5.3 therefore gives

$$\mathrm{E}_{\mathbf{Z}}\left[w_{d,k}^G(\mathbf{Y}_{d-1,k}, \mathbf{Z})\right] \le \frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(t_{d,k}, \tau - 1)}{2^m} + 2^{-\tau} \le \frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c_d m, \tau - 1)}{2^m} + 2^{-\tau},$$

where $\mathbf{Z}$ is uniform over $\{0,1\}^m$. Since the final bound holds for all keys and all paths,

$$\varepsilon_{d,k} \le \frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c_d m, \tau - 1)}{2^m} + 2^{-\tau}.$$

Plugging into Equation (10) yields

$$\varepsilon_{1,k} \leq 2 \cdot \left(\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c_d m, \tau - 1)}{2^m} + 2^{-\tau} \right)^{\alpha^{d-1}}.$$

Therefore,

$$\begin{aligned} \Pr_{f \leftarrow \mathcal{F}_\ell}\left[\mathcal{A}^{f,G}(1^n) = 1\right] &\leq \sum_{k \in \{0,1\}^\lambda} \mathrm{E}_{\mathbf{Z}}\left[w_k^G(\mathbf{Z})\right] \leq \sum_{k \in \{0,1\}^\lambda} \varepsilon_{1,k} \\ &\leq \left(2^{\lambda+1} \cdot \left(\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c_d m, \tau - 1)}{2^m} + 2^{-\tau} \right)^{\alpha^{d-1}} \right). \end{aligned}$$

Finally, suppose that $\tau \geq (\lambda + n + 1)/\alpha^{d-1}$ and $m \geq 3(2n + r) \cdot \tau$ are polynomials. By Fact 3.3,

$$\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c_d m, \tau - 1)}{2^m} + 2^{-\tau} \leq 2^{-m + (n+r+\log(c_d m))(\tau-1) + \log \tau} + 2^{-\tau}.$$

Since $\log(c_d m), \log \tau < n$ for all sufficiently large n's,

$$(n + r + \log(c_d m))(\tau - 1) + \log \tau \leq m/2$$

for all sufficiently large n's. Therefore,

$$2^{-m + (n+r+\log(c_d m))(\tau-1) + \log \tau} + 2^{-\tau} \leq 2^{-m/2} + 2^{-\tau}.$$

Since $m \geq 2\tau \geq 2(\lambda + n + 1)/\alpha^{d-1}$, we conclude that

$$\left(2^{\lambda+1} \cdot \left(\frac{\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c_d m, \tau - 1)}{2^m} + 2^{-\tau} \right)^{\alpha^{d-1}} \right) \leq \left(2^{\frac{\lambda+1}{\alpha^{d-1}} - \frac{m}{2}} + 2^{\frac{\lambda+1}{\alpha^{d-1}} - \tau} \right)^{\alpha^{d-1}} \leq 2^{-n + \alpha^{d-1}} \leq 2^{-n+1}.$$

□

Sensitivity analysis of $\mathcal{A}$. The next step of the proof is to bound the G-sensitivity of the adversary. Specifically, we show that it is bounded by $Cm \cdot 2^{-n+1+\frac{\alpha(n+r+\gamma)}{1-\alpha}}$.

Lemma 6.6. $\sigma^G(\mathcal{A}) \leq Cm \cdot 2^{-n+1+\frac{\alpha(n+r+\gamma)}{1-\alpha}}$.

Fix a function $G : \{0,1\}^n \to \{0,1\}^{n+r}$, a vector of values $\mathbf{v} = (v_s)_{s \in \{0,1\}^n}$, where $v_s \in \{0,1\}^{n+r}$ for every $s \in \{0,1\}^n$, and a vector of possible PRF outputs $\mathbf{z} \in \{0,1\}^m$. We abbreviate $G_s = G^{s \leftarrow v_s}$. Towards bounding the sensitivity, we first bound

$$\sum_{s \in \{0,1\}^n} \left| w_k^{G_s}(\mathbf{z}) - w_k^G(\mathbf{z}) \right|.$$

This will be done by recursively bounding the sum of differences on every path $\mathbf{Y}_{i,k}$. Intuitively, at a fixed node, changing a single entry s in G can increase a current-round penalty by a factor of at most η^{-1}. Raising the sum to the power α changes this current-round factor to $\eta^{-\alpha}$. We

can translate this ratio into a bound on the sum of differences: if $s \in \mathcal{T}_{i,k}$ then the ratio induces a contribution, and otherwise, there is no additional current-round contribution.

We next give the formal details. For every $i \in [d-1]$ let

$$e_i = \alpha \cdot (e_{i+1} + 1),$$

where $e_d = 1/\log \eta^{-1} = 1/(n + r + \gamma)$, and let

$$B_i = c_i m \cdot (\eta^{-e_i} - 1) + B_{i+1},$$

where $B_d = c_d m$. We prove the following recursive relation between the weights of every round.

Claim 6.7. *For every $i \in [d]$, every key $k \in \{0,1\}^\lambda$, and every prefix $\mathbf{Y}_{i-1,k}$,*

$$\begin{aligned}
&w^G_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z}) = 0 \Rightarrow w^{G_s}_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z}) = 0,\\
&w^G_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z}) > 0 \Rightarrow \eta^{e_i} \leq \frac{w^{G_s}_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z})}{w^G_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z})} \leq \eta^{-e_i},\\
&\sum_{s \in \{0,1\}^n} \left| w^{G_s}_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z}) - w^G_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z}) \right| \leq B_i \cdot w^G_{i,k}(\mathbf{Y}_{i-1,k}, \mathbf{z}).
\end{aligned}$$

Proof. For brevity, we remove $k \in \{0,1\}^\lambda$, $\mathbf{Y}_{i,k}$, and $\mathbf{z}$ from the notations when they are clear from context. The proof is by induction on i, starting from the leaves at $i = d$. If $w^G_d = 0$, then $\Phi^{-1}_k(\mathbf{Y}_{d-1,k}, \mathbf{z}) = \emptyset$. Since this is independent of G, $w^{G_s}_d = 0$. Assume now that $w^G_d > 0$. Changing one location in G can change the distance by at most 1. Recall that the weight in the leaves is $2^{-\mathsf{dist}^G_k(\mathbf{Y}_{d-1,k};\ \mathbf{z})}$, hence,

$$\eta^{e_d} = \frac{1}{2} \leq \frac{w^{G_s}_d}{w^G_d} \leq 2 = \eta^{-e_d}.$$

Since this implies $\left| w^{G_s}_d - w^G_d \right| \leq w^G_d$, we conclude that

$$\sum_{s \in \{0,1\}^n} \left| w^{G_s}_d - w^G_d \right| = \sum_{s \in \mathcal{T}_d} \left| w^{G_s}_d - w^G_d \right| \leq t_d \, w^G_d \leq c_d m \, w^G_d.$$

We now show that if the claim holds for $i+1$, it holds for i. If $w^G_i = 0$, then all of its children must be 0, i.e., $w^G_{i+1}((\mathbf{Y}_{i-1}, Y_i), \mathbf{z}) = 0$ for all $Y_i : \mathcal{T}_i \to \{0,1\}^{n+r}$. Therefore, all children of $w^{G_s}_i$ have weight 0, thus $w^{G_s}_i = 0$.

Assume now that $w^G_i > 0$. By the induction hypothesis, for every $Y_i : \mathcal{T}_i \to \{0,1\}^{n+r}$ such that $w^G_{i+1}(\mathbf{Y}_i, \mathbf{z}) > 0$,

$$\eta^{e_{i+1}} \leq \frac{w^{G_s}_{i+1}(\mathbf{Y}_i, \mathbf{z})}{w^G_{i+1}(\mathbf{Y}_i, \mathbf{z})} \leq \eta^{-e_{i+1}}. \tag{11}$$

Recall that

$$w^{G_s}_i(\mathbf{Y}_{i-1}, \mathbf{z}) = \left(\sum_{\substack{Y_i : \mathcal{T}_i \to \{0,1\}^{n+r}: \\ w^{G_s}_{i+1}(\mathbf{Y}_i, \mathbf{z}) > 0}} \eta^{\mathsf{dist}\left(Y_i, G_s|_{\mathcal{T}_i}\right)} \cdot w^{G_s}_{i+1}(\mathbf{Y}_i, \mathbf{z}) \right)^\alpha.$$

If $s \in \mathcal{T}_i$ then $|\mathsf{dist}\left(Y_i, G_s|_{\mathcal{T}_i}\right) - \mathsf{dist}\left(Y_{i,k}, G|_{\mathcal{T}_i}\right)| \leq 1$ for all Y_i, hence

$$\eta^{e_i} = \eta^{(e_{i+1}+1)\cdot\alpha} \leq \frac{w_i^{G_s}}{w_i^G} \leq \eta^{-(e_{i+1}+1)\cdot\alpha} = \eta^{-e_i}. \tag{12}$$

Otherwise, the difference is 0, hence

$$\eta^{e_i} \leq \eta^{e_{i+1}\cdot\alpha} \leq \frac{w_i^{G_s}}{w_i^G} \leq \eta^{-e_{i+1}\cdot\alpha} \leq \eta^{-e_i}.$$

For every $s \in \mathcal{T}_i$, Equation (12) yields

$$\left|w_i^{G_s} - w_i^G\right| \leq (\eta^{-e_i} - 1) w_i^G.$$

Therefore,

$$\sum_{s\in\mathcal{T}_i} \left|w_i^{G_s} - w_i^G\right| \leq t_i(\eta^{-e_i} - 1) w_i^G \leq c_i m(\eta^{-e_i} - 1) w_i^G.$$

We next bound the sum over all $s \notin \mathcal{T}_i$. Intuitively, changing $G(s)$ does not affect the current round at all. It can only change what happens below the current node, which is handled by the induction hypothesis. First, observe that for every $b \geq 0$, $|b^\alpha - 1| \leq |b - 1|$. Thus, for every $b \geq 0$ and $c > 0$,

$$|b^\alpha - c^\alpha| \leq \left|\frac{b}{c} - 1\right| \cdot c^\alpha = \frac{|b-c|}{c} \cdot c^\alpha.$$

Second, $\mathsf{dist}\left(Y_i, G_s|_{\mathcal{T}_i}\right) = \mathsf{dist}\left(Y_i, G|_{\mathcal{T}_i}\right)$ for all Y_i. Write this value as $\mathsf{dist}(Y_i)$. Therefore, for every $s \notin \mathcal{T}_i$,

$$\begin{aligned}
\left|w_i^{G_s} - w_i^G\right| &\leq \frac{\left|\sum_{Y_i:\mathcal{T}_i\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_i)} \left(w_{i+1}^{G_s}\left(\mathbf{Y}_i, \mathbf{z}\right) - w_{i+1}^{G}\left(\mathbf{Y}_i, \mathbf{z}\right)\right)\right|}{\sum_{Y_i:\mathcal{T}_i\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_i)} w_{i+1}^{G}\left(\mathbf{Y}_i, \mathbf{z}\right)} \cdot w_i^G \\
&\leq \frac{\sum_{Y_i:\mathcal{T}_i\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_i)} \cdot \left|w_{i+1}^{G_s}\left(\mathbf{Y}_i, \mathbf{z}\right) - w_{i+1}^{G}\left(\mathbf{Y}_i, \mathbf{z}\right)\right|}{\sum_{Y_i:\mathcal{T}_i\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_i)} w_{i+1}^{G}\left(\mathbf{Y}_i, \mathbf{z}\right)} \cdot w_i^G.
\end{aligned}$$

Summing over all $s \notin \mathcal{T}_i$, the induction hypothesis yields an upper bound of

$$\sum_{s\notin\mathcal{T}_i} \left|w_i^{G_s} - w_i^G\right| \leq \frac{\sum_{Y_i:\mathcal{T}_i\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_i)} \cdot B_{i+1} \cdot w_{i+1}^G}{\sum_{Y_i:\mathcal{T}_i\to\{0,1\}^{n+r}} \eta^{\mathsf{dist}(Y_i)} w_{i+1}^{G}\left(\mathbf{Y}_i, \mathbf{z}\right)} \cdot w_i^G = B_{i+1} \cdot w_i^G.$$

We conclude that

$$\sum_{s\in\{0,1\}^n} \left|w_i^{G_s} - w_i^G\right| \leq c_i m(\eta^{-e_i} - 1) w_i^G + B_{i+1} \cdot w_i^G = B_i \cdot w_i^G.$$

□

We now solve the recurrence.

Claim 6.8. *For every key $k \in \{0,1\}^\lambda$,*

$$\sum_{s\in\{0,1\}^n} \left| w_k^{G_s}(\mathbf{z}) - w_k^G(\mathbf{z}) \right| \le Cm \left(2^{1+\frac{\alpha}{1-\alpha}\cdot(n+r+\gamma)} - 1 \right) \cdot w_k^G(\mathbf{z})$$

Proof. For every $i \in [d]$, since $\alpha < 1/2$,

$$e_i \le \frac{\alpha^{d-i}}{\log \eta^{-1}} + \sum_{j=1}^{d-i} \alpha^j \le \frac{1}{n+r+\gamma} + \alpha \cdot \frac{1-\alpha^{d-i}}{1-\alpha} \le \frac{1}{n+r+\gamma} + \frac{\alpha}{1-\alpha}.$$

Therefore,

$$B_1 \le \sum_{i=1}^{d} c_i m \left(\eta^{-\left(\frac{1}{n+r+\gamma}+\frac{\alpha}{1-\alpha}\right)} - 1 \right) = Cm \left(2^{1+\frac{\alpha}{1-\alpha}\cdot(n+r+\gamma)} - 1 \right).$$

Thus, by Claim 6.7,

$$\sum_{s\in\{0,1\}^n} \left| w_k^{G_s}(\mathbf{z}) - w_k^G(\mathbf{z}) \right| \le Cm \left(2^{1+\frac{\alpha}{1-\alpha}\cdot(n+r+\gamma)} - 1 \right) \cdot w_k^G(\mathbf{z})$$

for every key $k \in \{0,1\}^\lambda$. □

We are now ready to show that the sensitivity of the adversary is negligible.

Proof of Lemma 6.6. We show that

$$\sigma^G(\mathcal{A}, \mathbf{v}, \mathbf{z}) \le Cm \cdot 2^{-n+1+\frac{\alpha(n+r+\gamma)}{1-\alpha}}.$$

Let

$$W = W^G(\mathbf{z}), \qquad W_s = W^{G_s}(\mathbf{z}).$$

By Claim 6.8,

$$\sum_{s\in\{0,1\}^n} \left| w_k^{G_s}(\mathbf{z}) - w_k^G(\mathbf{z}) \right| \le Cm \left(2^{1+\frac{\alpha}{1-\alpha}\cdot(n+r+\gamma)} - 1 \right) \cdot w_k^G(\mathbf{z})$$

for every key $k \in \{0,1\}^\lambda$. Summing over all keys $k \in \{0,1\}^\lambda$, the triangle inequality gives

$$\sum_{s\in\{0,1\}^n} |W_s - W| \le Cm \left(2^{1+\frac{\alpha}{1-\alpha}\cdot(n+r+\gamma)} - 1 \right) \cdot W. \tag{13}$$

Define $\phi(u) = \min\{1, u\}$. Then $p^G(\mathbf{z}) = \phi(W)$ and $p^{G^{s\leftarrow v_s}}(\mathbf{z}) = \phi(W_s)$. By Fact 3.1 with $a = W_s$ and $b = W$,

$$\sum_{s\in\{0,1\}^n} \left| p^{G_s}(\mathbf{z}) - p^G(\mathbf{z}) \right| = \sum_{s\in\{0,1\}^n} |\phi(W_s) - \phi(W)| \le \sum_{s\in\{0,1\}^n} \frac{|W_s - W|}{\max\{1, W\}}. \tag{14}$$

Applying Equations (13) and (14),

$$\sum_{s\in\{0,1\}^n} \left| p^{G_s}(\mathbf{z}) - p^G(\mathbf{z}) \right| \le \frac{Cm \left(2^{1+\frac{\alpha}{1-\alpha}\cdot(n+r+\gamma)} - 1 \right) \cdot W}{\max\{1, W\}} \le Cm \left(2^{1+\frac{\alpha}{1-\alpha}\cdot(n+r+\gamma)} - 1 \right).$$

Averaging over $s \in \{0,1\}^n$ and maximizing over $\mathbf{v}$, $\mathbf{z}$, and G yields

$$\sigma(\mathcal{A}) \le Cm \cdot 2^{-n+1+\frac{\alpha(n+r+\gamma)}{1-\alpha}},$$

as claimed. □

Obtaining a contradiction. For every $G : \{0,1\}^n \to \{0,1\}^{n+r}$, let

$$\Delta_G = \Pr_{s \leftarrow \{0,1\}^n}\left[R^{\mathcal{A}^G, G}(1^n, G(s)) = 1\right] - \Pr_{y \leftarrow \{0,1\}^{n+r}}\left[R^{\mathcal{A}^G, G}(1^n, y) = 1\right]$$

be the reduction's advantage. Let $q_G = q_G(n,m)$ denote the number of direct calls R makes to G, and let $q_{\mathcal{A}} = q_{\mathcal{A}}(n,m)$ denote the number of calls R makes to $\mathcal{A}^G$. By Theorem 4.4 and Lemma 6.6, there exists $G^* : \{0,1\}^n \to \{0,1\}^{n+r}$ such that

$$\begin{aligned}|\Delta_{G^*}| &\le \sqrt{4\left(\frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}} \\ &\le \sqrt{4\left(\frac{q_G}{2^n} + q_{\mathcal{A}} \cdot \left(Cm \cdot 2^{-n+1+\frac{\alpha(n+r+\gamma)}{1-\alpha}}\right)\right) + \frac{1}{2^n}}.\end{aligned}$$

We now choose the parameters to obtain a contradiction. To make $|\Delta_{G^*}|$ negligible, we need to ensure

$$\alpha \le \frac{n-\delta}{2n+r+\gamma-\delta},$$

for some $\delta = \delta(n) = \omega(\log n)$. On the other hand, by Lemma 6.3, taking $m \ge 3(2n+r)\cdot\tau$, where $\tau \ge (\lambda+n+1)/\alpha^{d-1}$, makes the adversary successfully break the PRF construction. Thus, for m to be polynomial, α must satisfy

$$\alpha^{d-1} \ge \frac{1}{\mathrm{poly}(n)}.$$

Recall that $d \le \frac{\beta \log n}{\log\left(2+\frac{r}{n}\right)}$ for some constant $\beta > 0$ and all sufficiently large n. Choose

$$\gamma = \log n^2, \qquad \delta = \gamma, \qquad \alpha = \frac{n-\gamma}{2n+r}.$$

For all sufficiently large n, we have $0 < \alpha < 1/2$ and

$$-n+1+\frac{\alpha(n+r+\gamma)}{1-\alpha} = 1-\gamma.$$

Moreover,

$$\log(1/\alpha) = \log(2+r/n) - \log(1-\gamma/n),$$

so $d\log(1/\alpha) = O(\log n)$ and $\alpha^{-(d-1)}$ is polynomially bounded. Thus we may take

$$\tau = (\lambda+n+1)/\alpha^{d-1}, \qquad m = 3(2n+r)\tau,$$

which are polynomially bounded. Since $\ell = \omega(\log n)$, eventually $m \le 2^\ell$, so we can select distinct inputs as required. The PRF adversary has advantage at least $1 - 2^{-n+1}$, whereas the reduction's advantage is negligible. This concludes the proof. □

7 Lower Bounds on Non-Adaptive Stretch Amplification

In this section, we prove a lower bound on the number of calls to a PRG with a stretch r required in order to construct a PRG with a larger stretch ρ.

Theorem 7.1. *Let $\ell(n)$, $r(n)$, $\rho(n,\ell)$, and $c(n)$ be polynomially bounded, and let $\gamma(n) = \omega(\log n)$ such that $0 \leq \gamma(n) < n$ for all $n \in \mathbb{N}$. Define*

$$\tau(n) = \min\left\{c(n), \left\lceil \frac{\ell(n)}{n-\gamma(n)} \right\rceil - 1\right\}.$$

If

$$\mathsf{Vol}_{\{0,1\}^{n+r(n)}}\left(c(n), \tau(n), 2^{-(n-\gamma(n))}\right) \leq 2^{\rho(n,\ell)-1},$$

then there is no $c(n)$-call fully black-box non-adaptive construction of a $\rho(n,\ell)$-bit stretch PRG from an $r(n)$-bit stretch PRG.

Proof. Assume towards contradiction that the claim is false. Then there exist polynomially bounded $\ell = \ell(n)$, $\rho = \rho(n,\ell)$, $r = r(n)$, and $c = c(n)$, and there exists $\gamma = \gamma(n) = \omega(\log n)$ such that $0 \leq \gamma < n$,

$$\mathsf{Vol}_{\{0,1\}^{n+r}}\left(c, \tau, 2^{-(n-\gamma)}\right) \leq 2^{\rho-1},$$

where

$$\tau = \tau(n) = \min\left\{c, \left\lceil \frac{\ell}{n-\gamma} \right\rceil - 1\right\},$$

yet there is a fully black-box construction $(\mathcal{C}, R)$ of a ρ-bit stretch PRG from an r-bit stretch PRG, using c non-adaptive calls.

$$\mathcal{C} = \left\{C_n^{(\cdot)} : \{0,1\}^{\ell} \to \{0,1\}^{\ell+\rho}\right\}_{n\in\mathbb{N}}$$

where $C_n^{(\cdot)}$ is given by

$$C_n^G(x) = P\left(x, G(S_1(x)), G(S_2(x)), \ldots, G(S_c(x))\right),$$

for efficiently computable functions $S_1, S_2, \ldots, S_c : \{0,1\}^{\ell} \to \{0,1\}^n$ and $P : \{0,1\}^{\ell} \times (\{0,1\}^{n+r})^c \to \{0,1\}^{\ell+\rho}$, where $\ell = \ell(n)$ is polynomially bounded. In the following, we remove n to simplify the notations.

To obtain a contradiction, we show that there exists an adversary $\mathcal{A}$ and a function $G^* : \{0,1\}^n \to \{0,1\}^{n+r}$ such that on the one hand

$$\left|\Pr_{x\leftarrow\{0,1\}^{\ell}}\left[\mathcal{A}^{G^*}(1^n, C^{G^*}(x)) = 1\right] - \Pr_{z\leftarrow\{0,1\}^{\ell+\rho}}\left[\mathcal{A}^{G^*}(1^n, z) = 1\right]\right| \geq \frac{1}{n^{\alpha}}$$

for some constant α and all sufficiently large n's; on the other hand,

$$\left|\Pr_{s\leftarrow\{0,1\}^n}\left[R^{\mathcal{A}^{G^*},G^*}(1^n, G^*(s)) = 1\right] - \Pr_{y\leftarrow\{0,1\}^{n+r}}\left[R^{\mathcal{A}^{G^*},G^*}(1^n, y) = 1\right]\right| = \mathrm{neg}(n).$$

We construct an adversary $\mathcal{A}^G$ and show that, on the one hand, $\mathcal{A}^G$ distinguishes $\mathcal{C}^G$ from uniform with constant advantage for every G; on the other hand, the G-sensitivity of $\mathcal{A}^G$ is small. We can then invoke Theorem 4.4 to obtain a function G^* on which the reduction has negligible advantage.

Proof overview. The proof follows the same approach as that of Theorem 5.1, but we have less freedom in the choice of parameters. Specifically, in the PRF setting, we can increase the number of queries m to make the expected total weight of a random-function transcript small. Here, the adversary receives a single challenge string of length $\ell + \rho$, which is fixed by the construction. To compensate, we use weights that decrease more rapidly with the number of required edits to G and set them to zero when the distance exceeds the parameter τ.

Notations and setup. We may replace γ by $\lfloor\gamma\rfloor$: this weakly decreases both $\eta = 2^{-(n-\gamma)}$ and τ, so it preserves the volume condition and $\gamma = \omega(\log n)$. Hence assume γ is integer-valued and denote $\eta = 2^{-(n-\gamma)}$. For every $x \in \{0,1\}^\ell$, let

$$\mathcal{T}_x = \{S_j(x) : j \in [c]\}$$

be the set of distinct primitive seeds queried when evaluating $\mathcal{C}^G(x)$, and let $t_x = |\mathcal{T}_x| \leq c$. For every assignment $Y : \mathcal{T}_x \to \{0,1\}^{n+r}$, let

$$\Phi_x(Y) = P\left(x, Y(S_1(x)), \ldots, Y(S_c(x))\right)$$

denote the output of the construction on input x when the answers to its oracle calls are specified by Y. Repeated primitive queries are answered consistently because Y is defined on the set $\mathcal{T}_x$.

For an output $z \in \{0,1\}^{\ell+\rho}$, let

$$\Phi_x^{-1}(z) = \left\{Y : \mathcal{T}_x \to \{0,1\}^{n+r} \mid \Phi_x(Y) = z\right\}.$$

Define

$$\mathsf{dist}_x^G(z) = \min_{Y \in \Phi_x^{-1}(z)} \mathsf{dist}(Y, G|_{\mathcal{T}_x})$$

to be the minimum number of changes to G on the queried seeds that are needed in order to obtain the output z. If $\Phi_x^{-1}(z) = \emptyset$, we let $\mathsf{dist}_x^G(z) = \infty$. Define the weight

$$w_x^G(z) = \begin{cases} \eta^{\mathsf{dist}_x^G(z)} & \text{if } \mathsf{dist}_x^G(z) \leq \tau \\ 0 & \text{otherwise,} \end{cases}$$

where $\eta^\infty = 0$, and finally let

$$W^G(z) = \sum_{x \in \{0,1\}^\ell} w_x^G(z), \qquad p^G(z) = \min\left\{1, W^G(z)\right\}.$$

We next define the adversary $\mathcal{A}^{(\cdot)}$. It has oracle access to a (purported) PRG $G : \{0,1\}^n \to \{0,1\}^{n+r}$, and on input $(1^n, z)$, where $z \in \{0,1\}^{\ell+\rho}$, the adversary $\mathcal{A}^G$ simply outputs 1 with probability $p^G(z)$ and outputs 0 with probability $1 - p^G(z)$.

We first show that $\mathcal{A}^G$ distinguishes the output of the construction from a uniform string. Intuitively, a genuine output is perfectly explained by its actual seed and therefore receives weight one. For a uniform string, the total weight associated with any fixed seed is bounded by the weighted volume of all consistent assignments of answers to the at most c oracle calls.

Claim 7.2. *For every $G : \{0,1\}^n \to \{0,1\}^{n+r}$,*

$$\Pr_{x \leftarrow \{0,1\}^\ell}\left[\mathcal{A}^G\left(1^n, \mathcal{C}^G(x)\right) = 1\right] - \Pr_{z \leftarrow \{0,1\}^{\ell+\rho}}\left[\mathcal{A}^G\left(1^n, z\right) = 1\right] \geq \frac{1}{2}.$$

Proof. We first show that the adversary always accepts a genuine output of C^G. Fix $x \in \{0,1\}^\ell$ and let $z = C^G(x)$. The assignment $Y = G|_{\mathcal{T}_x}$ belongs to $\Phi_x^{-1}(z)$, and hence $\mathsf{dist}_x^G(z) = 0$. Therefore $w_x^G(z) = 1$, implying that $W^G(z) \geq 1$ and $p^G(z) = 1$. Thus,

$$\Pr_{x \leftarrow \{0,1\}^\ell}\left[\mathcal{A}^G\left(1^n, C^G(x)\right) = 1\right] = 1.$$

We next analyze the probability that the adversary outputs 1 on a uniform string. Since $p^G(z) \leq W^G(z)$,

$$\begin{aligned}
\Pr_{z \leftarrow \{0,1\}^{\ell+\rho}}\left[\mathcal{A}^G(1^n, z) = 1\right] &= \mathrm{E}_Z\left[p^G(Z)\right] \\
&= \mathrm{E}_Z\left[\min\left\{1, W^G(Z)\right\}\right] \\
&\leq \mathrm{E}_Z\left[\sum_{x \in \{0,1\}^\ell} w_x^G(Z)\right] \\
&= \sum_{x \in \{0,1\}^\ell} \mathrm{E}_Z\left[w_x^G(Z)\right] \\
&= \frac{1}{2^{\ell+\rho}} \cdot \sum_{x \in \{0,1\}^\ell} \sum_{z \in \{0,1\}^{\ell+\rho}} w_x^G(z).
\end{aligned}$$

Fix $x \in \{0,1\}^\ell$. Every $z \in \{0,1\}^{\ell+\rho}$ such that $w_x^G(z) > 0$ is induced by some assignment $Y : \mathcal{T}_x \to \{0,1\}^{n+r}$ of distance at most τ from $G|_{\mathcal{T}_x}$. Thus,

$$\sum_{z \in \{0,1\}^{\ell+\rho}} w_x^G(z) = \sum_{z \in \{0,1\}^{\ell+\rho}} \max_{\substack{Y \in \Phi_x^{-1}(z): \\ \mathsf{dist}(Y, G|_{\mathcal{T}_x}) \leq \tau}} \eta^{\mathsf{dist}(Y, G|_{\mathcal{T}_x})} \leq \sum_{\substack{Y : \mathcal{T}_x \to \{0,1\}^{n+r}, \\ \mathsf{dist}(Y, G|_{\mathcal{T}_x}) \leq \tau}} \eta^{\mathsf{dist}(Y, G|_{\mathcal{T}_x})}.$$

Here an empty maximum is defined to be 0. For each queried seed, one possible assignment agrees with G, which contributes 1, while the other $2^{n+r} - 1$ have weight η. Every assignment Y can change at most $\min\{c, \tau\}$ locations, hence

$$\sum_{\substack{Y : \mathcal{T}_x \to \{0,1\}^{n+r}, \\ \mathsf{dist}(Y, G|_{\mathcal{T}_x}) \leq \tau}} \eta^{\mathsf{dist}(Y, G|_{\mathcal{T}_x})} \leq \mathsf{Vol}_{\{0,1\}^{n+r}}(c, \tau, \eta).$$

Therefore,

$$\Pr_{z \leftarrow \{0,1\}^{\ell+\rho}}\left[\mathcal{A}^G(1^n, z) = 1\right] \leq 2^{-\rho} \cdot \mathsf{Vol}_{\{0,1\}^{n+r}}(c, \tau, \eta) \leq \frac{1}{2}.$$

Combining the two bounds proves the claim. □

The next step is to bound the G-sensitivity of the adversary by $2c/2^\gamma$.

Lemma 7.3. $\sigma^G(\mathcal{A}) \leq 2c/2^\gamma$.

Proof. Fix a function $G : \{0,1\}^n \to \{0,1\}^{n+r}$, a vector $\mathbf{v} = (v_s)_{s \in \{0,1\}^n}$, where $v_s \in \{0,1\}^{n+r}$ for every $s \in \{0,1\}^n$, and an output $z \in \{0,1\}^{\ell+\rho}$. For every $s \in \{0,1\}^n$, let $G_s = G^{s \leftarrow v_s}$, and let

$$W = W^G(z), \qquad W_s = W^{G_s}(z).$$

Define $\phi(u) = \min\{1, u\}$. Then $p^G(z) = \phi(W)$ and $p^{G^{s \leftarrow v_s}}(z) = \phi(W_s)$. Thus, it suffices to show

$$\sum_{s \in \{0,1\}^n} |\phi(W_s) - \phi(W)| \leq 2c \cdot 2^{n-\gamma}.$$

By Fact 3.1 with $a = W_s$ and $b = W$,

$$|\phi(W_s) - \phi(W)| \le \frac{|W_s - W|}{\max\{1, W\}}. \tag{15}$$

By the triangle inequality,

$$\sum_{s\in\{0,1\}^n} |W_s - W| \le \sum_{s\in\{0,1\}^n} \sum_{x\in\{0,1\}^\ell} \left| w_x^{G_s}(z) - w_x^G(z) \right|.$$

If $s \notin \mathcal{T}_x$, then changing $G(s)$ does not affect $\mathsf{dist}_x^G(z)$, hence $w_x^{G_s}(z) = w_x^G(z)$. For every $s \in \mathcal{T}_x$ we prove the following bound.

Claim 7.4. *For every $x \in \{0,1\}^\ell$ and every $s \in \mathcal{T}_x$,*

$$\left| w_x^{G_s}(z) - w_x^G(z) \right| \le \eta^{-1} w_x^G(z) + \eta^\tau \cdot \mathbf{1}_{\left\{\mathsf{dist}_x^G(z) = \tau+1\right\}}.$$

Proof. If $\Phi_x^{-1}(z) = \emptyset$, both weights are zero and the claim is immediate. Otherwise, replacing $G(s)$ by v_s changes the Hamming distance to every feasible assignment by at most 1. Therefore,

$$\left| \mathsf{dist}_x^{G_s}(z) - \mathsf{dist}_x^G(z) \right| \le 1. \tag{16}$$

We now consider three cases. First, if $\mathsf{dist}_x^G(z) > \tau + 1$, then Equation (16) implies $w_x^{G_s}(z) = w_x^G(z) = 0$. Second, suppose $\mathsf{dist}_x^G(z) \le \tau$. If also $\mathsf{dist}_x^{G_s}(z) \le \tau$, then

$$\eta \le \frac{w_x^{G_s}(z)}{w_x^G(z)} \le \eta^{-1},$$

so the absolute change is at most $(\eta^{-1} - 1) w_x^G(z)$. Otherwise, the distance increases from τ to $\tau + 1$, so the new weight is zero. In both subcases,

$$\left| w_x^{G_s}(z) - w_x^G(z) \right| \le \eta^{-1} w_x^G(z).$$

Finally, if $\mathsf{dist}_x^G(z) = \tau + 1$, then $w_x^G(z) = 0$ and Equation (16) yields

$$\left| w_x^{G_s}(z) - w_x^G(z) \right| \le \eta^\tau \cdot \mathbf{1}_{\left\{\mathsf{dist}_x^{G_s}(z) = \tau\right\}} \le \eta^\tau.$$

□

For every $s \in \{0,1\}^n$, let

$$A_s = \sum_{\substack{x\in\{0,1\}^\ell: \\ s\in\mathcal{T}_x}} w_x^G(z) \quad \text{and} \quad B_s = \left| \left\{ x \in \{0,1\}^\ell : s \in \mathcal{T}_x \wedge \mathsf{dist}_x^G(z) = \tau + 1 \right\} \right|.$$

Then by Claim 7.4,

$$\sum_{s\in\{0,1\}^n} |W_s - W| \le \eta^{-1} \cdot \sum_{s\in\{0,1\}^n} A_s + \eta^\tau \cdot \sum_{s\in\{0,1\}^n} B_s. \tag{17}$$

We next bound the two sums. Observe that

$$\sum_{s\in\{0,1\}^n} A_s = \sum_{s\in\{0,1\}^n} \sum_{\substack{x\in\{0,1\}^\ell:\\ s\in\mathcal{T}_x}} w_x^G(z) = \sum_{x\in\{0,1\}^\ell} t_x \cdot w_x^G(z) \le cW. \tag{18}$$

As for the second sum, double counting gives $\sum_s B_s = \sum_x |\mathcal{T}_x| \, \mathbf{1}_{\{\mathsf{dist}_x^G(z)=\tau+1\}}$. Every finite distance is at most $|\mathcal{T}_x| \le c$, so this sum is zero when $\tau = c$ and at most $c2^\ell$ otherwise. Thus,

$$\sum_{s\in\{0,1\}^n} B_s \le c \cdot 2^\ell \cdot \mathbf{1}_{\{\tau<c\}}. \tag{19}$$

Combining Equations (15) and (17) to (19),

$$\begin{aligned}\sum_{s\in\{0,1\}^n} |\phi(W_s) - \phi(W)| &\le \frac{\eta^{-1} \cdot \sum_{s\in\{0,1\}^n} A_s + \eta^\tau \cdot \sum_{s\in\{0,1\}^n} B_s}{\max\{1, W\}} \\ &\le \frac{cW \cdot \eta^{-1} + c \cdot 2^\ell \cdot \eta^\tau \cdot \mathbf{1}_{\{\tau<c\}}}{\max\{1, W\}}.\end{aligned}$$

If $\tau = c$, then the second term is 0, hence this is bounded by $c \cdot \eta^{-1}$. Otherwise, $\tau = \left\lceil \frac{\ell}{n-\gamma} \right\rceil - 1$. Therefore,

$$\ell - \tau \cdot (n - \gamma) \le n - \gamma.$$

Since $\eta = 2^{-(n-\gamma)}$, the second term is bounded by

$$c \cdot 2^\ell \cdot \eta^\tau = c \cdot 2^{\ell - \tau(n-\gamma)} \le c \cdot 2^{n-\gamma}.$$

Overall, this yields

$$\sum_{s\in\{0,1\}^n} |\phi(W_s) - \phi(W)| \le c \cdot \eta^{-1} + c \cdot 2^{n-\gamma} \le 2c \cdot 2^{n-\gamma}.$$

Averaging over $s \in \{0,1\}^n$ and maximizing over $\mathbf{v}$, z, and G, yields

$$\sigma(\mathcal{A}) \le 2c/2^\gamma.$$

□

For every $G : \{0,1\}^n \to \{0,1\}^{n+r}$, let

$$\Delta_G = \Pr_{s\leftarrow\{0,1\}^n}\left[R^{\mathcal{A}^G,G}(1^n, G(s)) = 1\right] - \Pr_{y\leftarrow\{0,1\}^{n+r}}\left[R^{\mathcal{A}^G,G}(1^n, y) = 1\right]$$

be the reduction's advantage. Let $q_G = q_G(n)$ denote the number of direct calls R makes to G, and let $q_\mathcal{A} = q_\mathcal{A}(n)$ denote the number of fresh calls R makes to $\mathcal{A}^G$. By Theorem 4.4 and Lemma 7.3, there exists $G^* : \{0,1\}^n \to \{0,1\}^{n+r}$ such that

$$|\Delta_{G^*}| \le \sqrt{4 \cdot \left(\frac{q_G}{2^n} + q_\mathcal{A} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}} \le \sqrt{4 \cdot \left(\frac{q_G}{2^n} + q_\mathcal{A} \cdot \frac{2c}{2^\gamma}\right) + \frac{1}{2^n}}.$$

Since q_G, $q_\mathcal{A}$, and c are polynomially bounded and $\gamma = \omega(\log n)$, the right-hand side is negligible, resulting in a contradiction.

□

7.1 Corollaries

Corollary 7.5 (Asymptotic condition)**.** *Let $r = r(n)$, $\ell = \ell(n)$, $c = c(n)$, and $\rho = \rho(n, \ell(n))$ be polynomially bounded. Suppose that, for all sufficiently large n,*

$$(r + \log n) \cdot \min\left\{c, \frac{\ell}{n}\right\} = o(\rho).$$

Then there is no fully black-box c-call non-adaptive construction of a ρ-bit stretch PRG from an r-bit stretch PRG.

Proof. Let $\gamma = \gamma(n)$ be chosen by the analysis below. We show that it can be chosen to satisfy the conditions of Theorem 7.1. We choose it so that $\gamma = \omega(\log n)$ and $\gamma = o(n)$, and set $\eta = 2^{-(n-\gamma)}$. Let

$$\tau = \tau(n) = \min\left\{c, \left\lceil \frac{\ell}{n-\gamma} \right\rceil - 1\right\} = O\left(\min\left\{c, \frac{\ell}{n}\right\}\right).$$

If $c = 0$, the volume is 1 and Theorem 7.1 applies directly. For $c \geq 1$, Fact 3.3 gives

$$\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c, \tau, \eta) \leq (\tau + 1)\left(c \cdot 2^{r+\gamma}\right)^{\tau}.$$

Therefore,

$$\log\left(\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c, \tau, \eta)\right) \leq \log(\tau + 1) + \tau \cdot (\log c + r + \gamma).$$

Since c and τ are polynomially bounded in n, $\log c, \log(\tau + 1) = O(\log n)$. Since $\gamma = \omega(\log n)$,

$$\log\left(\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c, \tau, \eta)\right) = O(\tau \cdot (r + \gamma)) = O\left((r + \gamma) \cdot \min\left\{c, \frac{\ell}{n}\right\}\right).$$

Write $\gamma = h \cdot \log n$ for $h = h(n) = \omega(1)$ to be chosen below. Then

$$\begin{aligned}\log\left(\mathsf{Vol}_{\{0,1\}^{n+r(n)}}(c, \tau, \eta)\right) &= O\left((r + h \cdot \log n) \cdot \min\left\{c, \frac{\ell}{n}\right\}\right) \\ &\leq O\left(h \cdot (r + \log n) \cdot \min\left\{c, \frac{\ell}{n}\right\}\right).\end{aligned}$$

Recall that by assumption

$$(r + \log n) \cdot \min\left\{c, \frac{\ell}{n}\right\} = o(\rho).$$

Then taking h such that $h \log n = o(n)$ and

$$h \cdot (r + \log n) \cdot \min\left\{c, \frac{\ell}{n}\right\} = o(\rho)$$

concludes the proof. □

We next show a lower bound on the number of calls required in a non-adaptive construction.

Corollary 7.6. *For all polynomially bounded $r = r(n)$ and $\rho = \rho(n, \ell)$, there is no $o(\frac{\rho}{r+\log n})$-call fully black-box non-adaptive construction of a ρ-bit stretch PRG from an r-bit stretch PRG.*

Proof. Assume towards contradiction that $(\mathcal{C}, R)$ is a construction, where

$$\mathcal{C} = \left\{C^{(\cdot)} : \{0,1\}^{\ell} \to \{0,1\}^{\ell+\rho}\right\}_{n\in\mathbb{N}}.$$

Let $c = c(n) = o(\frac{\rho}{r+\log n})$ denote the number of calls to the oracle. Then

$$(r + \log n) \cdot \min\left\{c, \frac{\ell}{n}\right\} \leq (r + \log n) \cdot c = o(\rho),$$

contradicting Corollary 7.5. □

Finally, we show that there is no non-adaptive construction of a length-doubling PRG from a $o(n)$-bit stretch PRG.

Corollary 7.7. *For all $r(n) = o(n)$, there is no fully black-box non-adaptive construction of a length-doubling PRG from an r-bit stretch PRG.*

Proof. Assume towards contradiction that $(\mathcal{C}, R)$ is a construction, where

$$\mathcal{C} = \left\{C^{(\cdot)} : \{0,1\}^{\ell} \to \{0,1\}^{2\ell}\right\}_{n\in\mathbb{N}}.$$

Let $c = c(n)$ denote the number of calls to the oracle. Then

$$(r + \log n) \cdot \min\left\{c, \frac{\ell}{n}\right\} \leq (r + \log n) \cdot \frac{\ell}{n} = o(1) \cdot \ell = o(\ell),$$

contradicting Corollary 7.5. □

8 Impossibility of Constructions of CRH from OWP

In this section, we give another proof of Simon's result.

Theorem 8.1 (Simon's black-box impossibility)**.** *There is no fully black-box construction of a collision-resistant hash function family from a one-way permutation.*

The proof follows the same sensitivity framework we established in Section 4. However, an adversary for a CRH does not output a bit; hence, we first generalize the framework to include adversaries that may output a string. This is done in Section 8.1. We further replace Lemma 4.3 and Theorem 4.4 with analogues for reductions that attempt to invert a function. We do this in Section 8.2.

8.1 Sensitivity for Multi-Valued Algorithms

The collision-resistance adversary returns a pair of strings rather than a bit. We therefore measure sensitivity using statistical distance. Throughout the subsection, we fix $n, n', m \in \mathbb{N}$.

Definition 8.2 (Sensitivity of algorithms)**.** *Let $\mathcal{A}^{(\cdot)}$ be an oracle-aided algorithm that on input $\mathbf{z} \in \{0,1\}^m$ and oracle access to $f : \{0,1\}^n \to \{0,1\}^{n'}$ outputs a value in a discrete space Ω, and*

let $p^f(\mathbf{z})$ denote its output distribution. For a vector of values $\mathbf{v} = (v_s)_{s\in\{0,1\}^n}$, where $v_s \in \{0,1\}^{n'}$ for every $s \in \{0,1\}^n$, and a vector $\mathbf{z} \in \{0,1\}^m$, define

$$\sigma^f(\mathcal{A}, \mathbf{v}, \mathbf{z}) = \mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\mathrm{SD}\left(p^f(\mathbf{z}), p^{f^{s\leftarrow v_s}}(\mathbf{z})\right)\right].$$

For fixed f, define $\sigma^f(\mathcal{A}) = \max_{\mathbf{v},\mathbf{z}} \sigma^f(\mathcal{A}, \mathbf{v}, \mathbf{z})$. We define the sensitivity of $\mathcal{A}$ *as*

$$\sigma(\mathcal{A}) = \max_f \sigma^f(\mathcal{A}).$$

When $\mathcal{A}$ has an additional oracle, the same definition applies with that oracle fixed. We also write $\sigma_f(\mathcal{A})$ for $\sigma^f(\mathcal{A})$.

For Boolean algorithms, statistical distance between the two Bernoulli output distributions equals the absolute difference of their acceptance probabilities, so Definition 8.2 coincides with Definition 4.1.

We next extend the sensitivity-transfer lemma to arbitrary output distributions. The proof follows the same strategy as Lemma 4.2.

Lemma 8.3 (Sensitivity transfer)**.** *Let $\mathcal{A}^{(\cdot)}$ be an oracle-aided algorithm that on input $\mathbf{z} \in \{0,1\}^m$ and oracle access to a function $f : \{0,1\}^n \to \{0,1\}^{n'}$, outputs a value distributed according to $p^f(\mathbf{z})$. Consider an oracle-aided algorithm $R^{(\cdot),(\cdot)}$ that on $y \in \{0,1\}^{n'}$ and 1^n outputs a value in a discrete space, and define $h^f(y)$ to be the output distribution*

$$h^f(y) = R^{\mathcal{A}^f, f}(1^n, y),$$

where the distribution is taken over the reduction's coins and the private coins of $\mathcal{A}^f$ over all calls to it. Let q_f denote an upper bound on the number of calls R makes to f, and let $q_{\mathcal{A}}$ denote an upper bound on the number of calls R makes to $\mathcal{A}^f$. Write $\sigma(R)$ for the sensitivity of the composed algorithm $R^{\mathcal{A}^f, f}$. Then

$$\sigma(R) \le \frac{q_f}{2^n} + q_{\mathcal{A}} \cdot \sigma(\mathcal{A}).$$

Proof. Fix $y \in \{0,1\}^{n'}$, $f : \{0,1\}^n \to \{0,1\}^{n'}$, and $\mathbf{v} = (v_s)_{s\in\{0,1\}^n}$. We bound

$$\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\mathrm{SD}\left(h^f(y), h^{f^{s\leftarrow v_s}}(y)\right)\right].$$

The proof is done by coupling the execution $R^{\mathcal{A}^f, f}(1^n, y)$ with all of the executions $R^{\mathcal{A}^{f^{s\leftarrow v_s}}, f^{s\leftarrow v_s}}(1^n, y)$ over all $s \in \{0,1\}^n$. Denote the base execution by E and for every $s \in \{0,1\}^n$, denote the perturbed execution by E_s. We couple them using the same randomness for R in all executions. As long as E and E_s have the same transcript, the message R sends to the adversary is the same string $\mathbf{z}_i \in \{0,1\}^m$. At the ith call to the adversary, couple the answers with a maximal coupling of $p^f(\mathbf{z}_i)$ and $p^{f^{s\leftarrow v_s}}(\mathbf{z}_i)$. These couplings can all use the same base answer: sample it according to $p^f(\mathbf{z}_i)$, and then sample each perturbed answer from its conditional distribution in the corresponding maximal coupling. After the first time E_s diverges from E, we complete the execution with arbitrary fresh randomness having the correct marginal distribution.

Let F_s denote the event that E_s diverges from E. Observe that if E and E_s do not diverge, then R outputs the same value in both executions. Therefore, for every $s \in \{0,1\}^n$,

$$\mathrm{SD}\left(h^f(y), h^{f^{s\leftarrow v_s}}(y)\right) \le \Pr\left[\mathsf{F}_s\right].$$

There are two ways the executions may diverge. The first is if R queries f on s, and the second is if $\mathcal{A}$ responds in execution E differently from execution E_s. Let F'_s be the event that divergence first occurred because R queried f on s, and let $\mathsf{F}_{s,i}$ be the event that divergence first occurred at the ith call to the adversary. Since R makes at most q_f queries to f, it follows that

$$\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\Pr\left[\mathsf{F}'_s\right]\right] = \frac{1}{2^n}\cdot\sum_{s\in\{0,1\}^n}\Pr\left[\mathsf{F}'_s\right] \le \frac{q_f}{2^n}.$$

Now, consider the ith call to the adversary in execution E. Let H_i denote the coupled history until the answers to this call are sampled, and for every $s\in\{0,1\}^n$, let $I_{s,i}$ be the indicator random variable for the event that E and E_s have the same transcript until this call. If the call does not occur, use an arbitrary legal input $\mathbf{z}_i$ and set $I_{s,i}=0$. Then

$$\Pr\left[\mathsf{F}_{s,i}\mid H_i\right] = I_{s,i}\cdot \mathrm{SD}\left(p^f(\mathbf{z}_i), p^{f^{s\leftarrow v_s}}(\mathbf{z}_i)\right).$$

Thus,

$$\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\Pr\left[\mathsf{F}_{s,i}\mid H_i\right]\right] \le \sigma^f(\mathcal{A}),$$

where the inequality follows from $I_{s,i}\le 1$ and Definition 8.2. Since R calls the adversary at most $q_{\mathcal{A}}$ times, combining the two bounds and averaging over the coupled histories yields

$$\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\Pr\left[\mathsf{F}_s\right]\right] \le \frac{q_f}{2^n} + q_{\mathcal{A}}\cdot\sigma^f(\mathcal{A}).$$

Maximizing over $\mathbf{v}$, y, and f proves the inequality. □

8.2 An Inversion Planting Lemma

We now replace Lemma 4.3 by a planting argument tailored to inversion probability.

Lemma 8.4 (Inversion planting)**.** *Let $R^{(\cdot)}$ be an oracle-aided algorithm that on input $y\in\{0,1\}^n$ and oracle access to $\pi:\{0,1\}^n\to\{0,1\}^n$ outputs a value in $\{0,1\}^n\cup\{\perp\}$. For every $\pi\in\Pi_n$ define*

$$\mathrm{Inv}_\pi = \mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\Pr\left[R^\pi\left(\pi(s)\right)=s\right]\right].$$

Then

$$\mathrm{E}_{\pi\leftarrow\Pi_n}\left[\mathrm{Inv}_\pi\right] \le 2\sigma(R) + \frac{1}{2^n}.$$

Proof. For every $\pi\in\Pi_n$ and $y\in\{0,1\}^n$ let $h^\pi(y)$ denote the output distribution of $R^\pi(y)$. Sample $\pi\leftarrow\Pi_n$ and $s,y\leftarrow\{0,1\}^n$ independently, and let $s'=\pi^{-1}(y)$ and $v=\pi(s)$. Define $\pi_0=\pi$ and $\pi_1=\pi^{s\leftarrow y}$, and let π_2 be obtained by swapping the values of π at s and s'. That is, when $s\ne s'$, let $\pi_2=\pi^{s\leftarrow y,s'\leftarrow v}$, and when $s=s'$, let $\pi_2=\pi$. Then π_0 and π_2 are permutations, and $\pi_2(s)=y$. Note that π_1 need not be a permutation. Moreover,

$$\pi_1=\pi_0^{s\leftarrow y} \qquad\text{and}\qquad \pi_1=\pi_2^{s'\leftarrow y}. \tag{20}$$

We first compare π_0 and π_1. Fix π and y. Since s is uniform over $\{0,1\}^n$, applying Definition 8.2 with the constant replacement vector $v_t=y$ for every $t\in\{0,1\}^n$ yields

$$\mathrm{E}_{\pi,y}\left[\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\mathrm{SD}\left(h^{\pi_0}(y),h^{\pi_1}(y)\right)\right]\right] \le \sigma(R). \tag{21}$$

Next, we compare π_2 and π_1. Observe that the map

$$(\pi, s, y) \longmapsto (\pi_2, s, s')$$

is a bijection from $\Pi_n \times \{0,1\}^n \times \{0,1\}^n$ to itself. Indeed, given (π_2, s, s'), recover π by swapping the values of π_2 at s and s', and then recover $y = \pi_2(s)$. Therefore, π_2, s, s' are independent and uniform over $\Pi_n, \{0,1\}^n, \{0,1\}^n$, respectively. Fix π_2 and s. Then $y = \pi_2(s)$ is fixed and s' is uniform over $\{0,1\}^n$. Using Equation (20) and applying Definition 8.2 with the constant replacement vector $v_t = y$ for every $t \in \{0,1\}^n$ yields

$$\mathrm{E}_{\pi_2,s}\left[\mathrm{E}_{s' \leftarrow \{0,1\}^n}\left[\mathrm{SD}\left(h^{\pi_2}(y), h^{\pi_1}(y)\right)\right]\right] \leq \sigma(R). \tag{22}$$

By the triangle inequality and Equations (21) and (22),

$$\mathrm{E}_{\pi,s,y}\left[\mathrm{SD}\left(h^{\pi_0}(y), h^{\pi_2}(y)\right)\right] \leq 2\sigma(R). \tag{23}$$

We now bound the probability assigned to s by $h^{\pi_0}(y)$. Let

$$h^{\pi}(y, s) = \Pr\left[R^{\pi}(y) = s\right].$$

Since s is independent of $\pi_0 = \pi$ and y,

$$\mathrm{E}_{\pi,s,y}\left[h^{\pi_0}(y,s)\right] = \mathrm{E}_{\pi,y}\left[\frac{1}{2^n} \cdot \sum_{s \in \{0,1\}^n} h^{\pi_0}(y,s)\right] = \frac{1}{2^n} \cdot \sum_{s \in \{0,1\}^n} \mathrm{E}_{\pi,y}\left[h^{\pi_0}(y,s)\right] \leq \frac{1}{2^n}. \tag{24}$$

The difference between the probabilities assigned to s by two distributions is at most their statistical distance. Therefore,

$$\begin{aligned}\mathrm{E}_{\pi,s,y}\left[h^{\pi_2}(y,s)\right] &\leq \mathrm{E}_{\pi,s,y}\left[h^{\pi_0}(y,s)\right] + \mathrm{E}_{\pi,s,y}\left[\mathrm{SD}\left(h^{\pi_0}(y), h^{\pi_2}(y)\right)\right] \\ &\leq \frac{1}{2^n} + 2\sigma(R),\end{aligned} \tag{25}$$

where the last inequality is by Equations (23) and (24). Finally, π_2 is a uniform permutation independent of s, and $y = \pi_2(s)$. Thus,

$$\mathrm{E}_{\pi \leftarrow \Pi_n}\left[\mathrm{Inv}_{\pi}\right] = \mathrm{E}_{\pi_2 \leftarrow \Pi_n}\left[\mathrm{E}_{s \leftarrow \{0,1\}^n}\left[h^{\pi_2}(\pi_2(s), s)\right]\right] = \mathrm{E}_{\pi,s,y}\left[h^{\pi_2}(y,s)\right] \leq \frac{1}{2^n} + 2\sigma(R).$$

□

We are now ready to prove the main result of this subsection. We show that if the π-sensitivity of an adversary is small, then there is a permutation π^* on which the reduction has small inversion probability.

Theorem 8.5. *Let $\mathcal{A}^{(\cdot)}$ be an oracle-aided algorithm that on input $\mathbf{z} \in \{0,1\}^m$ and oracle access to a function $\pi : \{0,1\}^n \to \{0,1\}^n$, outputs a value in a discrete space Ω. Consider an oracle-aided algorithm $R^{(\cdot),(\cdot)}$ that on $y \in \{0,1\}^n$ and 1^n outputs a value in $\{0,1\}^n \cup \{\perp\}$. Let q_π denote an upper bound on the number of calls R makes to π, and let $q_{\mathcal{A}}$ denote an upper bound on the number of calls R makes to $\mathcal{A}^{\pi}$.*

For every $\pi \in \Pi_n$, let

$$\mathrm{Inv}_\pi = \Pr_{s \leftarrow \{0,1\}^n}\left[R^{\mathcal{A}^\pi,\pi}(1^n, \pi(s)) = s\right].$$

Then there exists $\pi^* \in \Pi_n$ such that

$$\mathrm{Inv}_{\pi^*} \leq 2 \cdot \left(\frac{q_\pi}{2^n} + q_\mathcal{A} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}.$$

Proof. By Lemma 8.4, applied to the composed algorithm $R^{\mathcal{A}^\pi,\pi}$,

$$\mathrm{E}_{\pi \leftarrow \Pi_n}[\mathrm{Inv}_\pi] \leq 2\sigma(R) + \frac{1}{2^n}.$$

Lemma 8.3 yields

$$\sigma(R) \leq \frac{q_\pi}{2^n} + q_\mathcal{A} \cdot \sigma(\mathcal{A}), \tag{26}$$

and hence

$$\mathrm{E}_{\pi \leftarrow \Pi_n}[\mathrm{Inv}_\pi] \leq 2 \cdot \left(\frac{q_\pi}{2^n} + q_\mathcal{A} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}. \tag{27}$$

Let π^* be the lexicographically first minimizer of Inv_π. Then

$$\mathrm{Inv}_{\pi^*} = \min_{\pi \in \Pi_n}\{\mathrm{Inv}_\pi\} \leq \mathrm{E}_{\pi \leftarrow \Pi_n}[\mathrm{Inv}_\pi] \leq 2 \cdot \left(\frac{q_\pi}{2^n} + q_\mathcal{A} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n}.$$

□

8.3 Proof of Theorem 8.1

Proof of Theorem 8.1. Assume towards contradiction that $(\mathcal{H}, R)$ is a fully black-box construction of a CRH from a OWP as in Definition 3.10. Let $c = c(n)$ be a polynomial upper bound on the number of oracle calls made by one hash evaluation, and write

$$\mathcal{H} = \left\{h_k^{(\cdot)} : \{0,1\}^{\ell(n)} \to \{0,1\}^{\ell_{\mathsf{out}}(n)}\right\}_{k \in \{0,1\}^{\lambda(n)}}.$$

In the rest of the proof, we remove n from notations for brevity. When considering h_k^π where π and k are clear from context, we will further remove them from notations.

We construct an adversary $\mathcal{A}^\pi$ and show that, on the one hand, $\mathcal{A}^\pi$ finds a collision with constant probability for every π; on the other hand, its π-sensitivity is negligible. We can then invoke Theorem 8.5 to obtain a permutation π^* on which the reduction has negligible inversion probability.

The adversary. The adversary we consider is the same as in [20]. It uses the following collision-sampling procedure.

Algorithm 8.6 (Collision-sampling adversary $\mathcal{A}^{(\cdot)}$)**.**
Input: The security parameter 1^n and a hash key $k \in \{0,1\}^\lambda$.
Oracle access: A function $\pi : \{0,1\}^n \to \{0,1\}^n$.

1. *Sample* $X \leftarrow \{0,1\}^{\ell}$.
2. *Sample* $X' \leftarrow \{x' \in \{0,1\}^{\ell} : h(x') = h(X)\}$. *Note that the set is not empty since it contains* X.
3. *Output* (X, X').

Success analysis of $\mathcal{A}$. We first show that the adversary finds a collision with probability at least $1/2$.

Claim 8.7. *For every function* $\pi : \{0,1\}^n \to \{0,1\}^n$ *and every key* $k \in \{0,1\}^{\lambda}$, *Algorithm 8.6 outputs a valid collision in* $h = h_k^{\pi}$ *with probability at least*

$$1 - 2^{\ell_{\mathsf{out}} - \ell} \geq \frac{1}{2}.$$

Proof. For every $r \in \mathrm{Im}(h)$, let

$$a_r = \left| \left\{ x \in \{0,1\}^{\ell} : h(x) = r \right\} \right|.$$

Conditioned on the event $h(X) = r$, the second sample X' is uniform over a set of size a_r, hence

$$\Pr\left[X' = X \mid h(X) = r\right] = \frac{1}{a_r}.$$

Since $\Pr\left[h(X) = r\right] = a_r / 2^{\ell}$, it follows that

$$\begin{aligned}
\Pr\left[X' = X\right] &= \sum_{r \in \mathrm{Im}(h)} \Pr\left[h(X) = r\right] \cdot \Pr\left[X' = X \mid h(X) = r\right] \\
&= \sum_{r \in \mathrm{Im}(h)} \frac{a_r}{2^{\ell}} \cdot \frac{1}{a_r} \\
&= \frac{|\mathrm{Im}(h)|}{2^{\ell}} \\
&\leq 2^{\ell_{\mathsf{out}} - \ell}.
\end{aligned}$$

Whenever $X' \neq X$, we have $h(X) = h(X')$, so the two distinct points form a valid collision. Since $\ell > \ell_{\mathsf{out}}$, the stated success probability is at least $1/2$. □

Sensitivity analysis of $\mathcal{A}$. We next show that the adversary has negligible sensitivity. Towards proving this, we first show that the collision-sampling distribution is stable when the function is changed on a small set of inputs.

Lemma 8.8 (Stability of collision sampling)**.** *Let* $h, h' : \{0,1\}^{\ell} \to \{0,1\}^{\ell_{\mathsf{out}}}$ *be functions, and let*

$$B = \left\{ x \in \{0,1\}^{\ell} : h(x) \neq h'(x) \right\}$$

denote the set of disagreements. Let $\mathsf{Col}(h)$ *denote the distribution that outputs the pair* (X, X'), *where* $X \leftarrow \{0,1\}^{\ell}$ *and*

$$X' \leftarrow \left\{ x' \in \{0,1\}^{\ell} : h(x') = h(X) \right\}.$$

Thus $\mathsf{Col}(h)$ *is a distribution over ordered pairs* $(X, X') \in \{0,1\}^\ell \times \{0,1\}^\ell$. *Then*

$$\mathrm{SD}\left(\mathsf{Col}(h), \mathsf{Col}(h')\right) \le \frac{3|B|}{2^\ell}.$$

Proof. Both distributions have the same first marginal: in each experiment X is uniform over $\{0,1\}^\ell$. Because pairs with different first coordinate have disjoint support, statistical distance reduces to the average statistical distance between the conditional distributions of the second coordinate, i.e.,

$$\mathrm{SD}\left(\mathsf{Col}(h), \mathsf{Col}(h')\right) = \frac{1}{2^\ell} \sum_{x \in \{0,1\}^\ell} \mathrm{SD}\left(\mathsf{U}\left(h^{-1}(h(x))\right), \mathsf{U}\left((h')^{-1}(h'(x))\right)\right). \tag{28}$$

For $x \in B$, the corresponding statistical distance is at most 1. Hence the total contribution of these values to Equation (28) is at most

$$\frac{1}{2^\ell} \sum_{x \in B} 1 = \frac{|B|}{2^\ell}. \tag{29}$$

It remains to analyze $x \notin B$. In this case $h(x) = h'(x)$. For every $r \in \{0,1\}^{\ell_{\mathsf{out}}}$, define

$$\begin{aligned} C_r &= \left\{x \in \{0,1\}^\ell \setminus B : h(x) = r\right\} = \left\{x \in \{0,1\}^\ell \setminus B : h'(x) = r\right\}, \\ D_r &= \left\{x \in B : h(x) = r\right\}, \\ D'_r &= \left\{x \in B : h'(x) = r\right\}. \end{aligned}$$

Thus

$$h^{-1}(r) = C_r \cup D_r, \qquad (h')^{-1}(r) = C_r \cup D'_r, \qquad \text{and} \qquad D_r \cap D'_r = \emptyset.$$

Therefore the common support of $\mathsf{U}(C_r \cup U_r)$ and $\mathsf{U}(C_r \cup D'_r)$ is C_r.

Write

$$c_r = |C_r|, \qquad d_r = |D_r|, \qquad d'_r = |D'_r|.$$

We only need to consider r for which $c_r > 0$, since otherwise there is no first coordinate $x \notin B$ with common value r. For such an r, both sets $C_r \cup D_r$ and $C_r \cup D'_r$ are nonempty. Then

$$\mathrm{SD}\left(\mathsf{U}(C_r \cup D_r), \mathsf{U}(C_r \cup D'_r)\right) = \max_{\mathcal{S} \subseteq C_r \cup D_r \cup D'_r} \left| \frac{|\mathcal{S} \cap (C_R \cup D_r)|}{c_r + d_r} - \frac{|\mathcal{S} \cap (C_R \cup D'_r)|}{c_r + d'_r} \right|.$$

Since the maximum is obtained for $\mathcal{S} \in \{C_r \cup D_r, C_r \cup D'_r\}$, it follows that

$$\begin{aligned} \mathrm{SD}\left(\mathsf{U}(C_r \cup D_r), \mathsf{U}(C_r \cup D'_r)\right) &= \max\left\{1 - \frac{c_r}{c_r + d'_r}, 1 - \frac{c_r}{c_r + d_r}\right\} \\ &= \frac{\max\{d_r, d'_r\}}{c_r + \max\{d_r, d'_r\}}. \end{aligned}$$

There are exactly c_r choices of $x \notin B$ whose common value $h(x) = h'(x)$ equals r. Hence, the total contribution of all $x \notin B$ is

$$\begin{aligned}
\frac{1}{2^\ell} \sum_{x \notin B} \mathrm{SD}\left(\mathsf{U}\left(h^{-1}(h(x))\right), \mathsf{U}\left((h')^{-1}(h'(x))\right)\right) &= \frac{1}{2^\ell} \sum_{r : c_r > 0} c_r \cdot \mathrm{SD}\left(\mathsf{U}(C_r \cup D_r), \mathsf{U}(C_r \cup D'_r)\right) \\
&= \frac{1}{2^\ell} \sum_{r : c_r > 0} \frac{c_r \max\{d_r, d'_r\}}{c_r + \max\{d_r, d'_r\}} \\
&\le \frac{1}{2^\ell} \sum_{r : c_r > 0} \max\{d_r, d'_r\} \\
&\le \frac{1}{2^\ell} \sum_{r \in \{0,1\}^{\ell_{\mathsf{out}}}} (d_r + d'_r). \qquad (30)
\end{aligned}$$

Every point of B belongs to exactly one set D_r, namely the set indexed by $r = h(x)$, and to exactly one set D'_r, namely the set indexed by $r = h'(x)$. Therefore

$$\sum_{r \in \{0,1\}^{\ell_{\mathsf{out}}}} d_r = \sum_{r \in \{0,1\}^{\ell_{\mathsf{out}}}} d'_r = |B|.$$

Substituting into Equation (30), the contribution is bounded by

$$\frac{1}{2^\ell} \sum_{r \in \{0,1\}^{\ell_{\mathsf{out}}}} (d_r + d'_r) = \frac{2|B|}{2^\ell}. \qquad (31)$$

Combining Equations (29) and (31) yields

$$\mathrm{SD}\left(\mathsf{Col}(h), \mathsf{Col}(h')\right) \le \frac{|B|}{2^\ell} + \frac{2|B|}{2^\ell} = \frac{3|B|}{2^\ell}.$$

□

Lemma 8.9. *Suppose every evaluation of $h_k^\pi(x)$ makes at most $c(n)$ oracle queries. Then Algorithm 8.6 satisfies*

$$\sigma(\mathcal{A}) \le \frac{3c(n)}{2^n}.$$

Proof. Fix an arbitrary function $\pi : \{0,1\}^n \to \{0,1\}^n$, a replacement vector $\mathbf{v} = (v_s)_{s \in \{0,1\}^n}$, and a legal input $(1^n, k)$. For every $s \in \{0,1\}^n$, let

$$\pi_s = \pi^{s \leftarrow v_s}, \qquad h = h_k^\pi, \qquad h_s = h_k^{\pi_s},$$

and define

$$B_s = \left\{x \in \{0,1\}^\ell : h(x) \ne h_s(x)\right\}.$$

We show that

$$\mathrm{E}_{s \leftarrow \{0,1\}^n}\left[\mathrm{SD}\left(\mathcal{A}^\pi(1^n, k), \mathcal{A}^{\pi_s}(1^n, k)\right)\right] \le \frac{3c(n)}{2^n}.$$

By Lemma 8.8, for every $s \in \{0,1\}^n$,

$$\mathrm{SD}\left(\mathcal{A}^\pi(1^n, k), \mathcal{A}^{\pi_s}(1^n, k)\right) \le \frac{3|B_s|}{2^\ell},$$

hence it suffices to bound $\sum_{s\in\{0,1\}^n} |B_s|$. For every $x \in \{0,1\}^\ell$, let

$$Q_x = \{s \in \{0,1\}^n : \text{the evaluation of } h_k^\pi(x) \text{ queries } s\}$$

be the set of distinct oracle locations queried by the evaluation on input x. Observe that if the computation of $h_k^\pi(x)$ never queries the oracle location s, then the computations of $h_k^\pi(x)$ and $h_k^{\pi_s}(x)$ receive identical answers at every query and therefore have the same output. Consequently,

$$x \in B_s \implies s \in Q_x.$$

Therefore

$$\begin{aligned}
\sum_{s\in\{0,1\}^n} |B_s| &= \sum_{s\in\{0,1\}^n} \sum_{x\in\{0,1\}^\ell} \mathbf{1}_{x\in B_s} \\
&\le \sum_{s\in\{0,1\}^n} \sum_{x\in\{0,1\}^\ell} \mathbf{1}_{s\in Q_x} \\
&= \sum_{x\in\{0,1\}^\ell} \sum_{s\in\{0,1\}^n} \mathbf{1}_{s\in Q_x} \\
&= \sum_{x\in\{0,1\}^\ell} |Q_x|.
\end{aligned}$$

Since one evaluation makes at most $c(n)$ oracle calls, $|Q_x| \le c(n)$ for every x. Thus,

$$\sum_{s\in\{0,1\}^n} |B_s| \le c(n)2^\ell.$$

We conclude that

$$\begin{aligned}
\mathrm{E}_{s\leftarrow\{0,1\}^n}\left[\mathrm{SD}\left(\mathcal{A}^\pi(1^n,k), \mathcal{A}^{\pi_s}(1^n,k)\right)\right] &\le \frac{1}{2^n} \sum_{s\in\{0,1\}^n} \frac{3|B_s|}{2^\ell} \\
&\le \frac{3}{2^n 2^\ell} \cdot c(n) 2^\ell \\
&= \frac{3c(n)}{2^n}.
\end{aligned}$$

□

Obtaining a contradiction. For every $\pi \in \Pi_n$, let

$$\mathrm{Inv}_\pi = \Pr_{s\leftarrow\{0,1\}^n}\left[R^{\mathcal{A}^\pi,\pi}(1^n, \pi(s)) = s\right]$$

be the reduction's inversion probability. Let $q_\pi = q_\pi(n)$ denote the number of direct calls R makes to π, and let $q_\mathcal{A} = q_\mathcal{A}(n)$ denote the number of fresh calls R makes to $\mathcal{A}^\pi$. By Theorem 8.5 and Lemma 8.9, for every n there exists $\pi_n^* \in \Pi_n$ such that

$$\begin{aligned}
\mathrm{Inv}_{\pi_n^*} &\le 2 \cdot \left(\frac{q_\pi}{2^n} + q_\mathcal{A} \cdot \sigma(\mathcal{A})\right) + \frac{1}{2^n} \\
&\le 2 \cdot \left(\frac{q_\pi}{2^n} + q_\mathcal{A} \cdot \frac{3c}{2^n}\right) + \frac{1}{2^n} \\
&= \frac{1 + 2q_\pi + 6q_\mathcal{A} \cdot c}{2^n}.
\end{aligned}$$

Since q_π, $q_\mathcal{A}$, and c are polynomials, the right-hand side is negligible. On the other hand, by Claim 8.7, the adversary finds a collision with probability at least $1/2$, resulting in a contradiction. □

8.4 Comparison with Bitansky and Degwekar

Bitansky and Degwekar [3] also gave a proof of Simon's separation based on coupling arguments. Their proof shares many similarities with ours: They start from the observation that a collision finder can break the candidate hash function while revealing little about a randomly chosen entry of the underlying permutation. They then exploit this by temporarily mapping two inputs to the same inversion challenge. If the reduction is unlikely to notice the local changes involved, it cannot reliably identify the genuine preimage.

The main difference is how the two proofs establish that the changes go unnoticed. Bitansky and Degwekar smooth the queries to the collision finder and use punctured versions of the oracle to couple entire executions. We instead bound how much a single change to the permutation alters the distribution of a collision pair, on average over the changed location. Our sensitivity framework then transfers this bound to the reduction and uses it to bound the inversion probability. This separates the analysis of the collision finder from the argument for hardness of inversion.

Bibliography

[1] B. Alon, I. Dinur, and M. Venkitasubramaniam. Lower bounds on black-box constructions of pseudorandom functions. In N. Heninger and M. Rosulek, editors, *Advances in Cryptology - CRYPTO 2026 - 46th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-20, 2026, Proceedings, Part I*, volume 16800 of *Lecture Notes in Computer Science*, pages 159–189. Springer, 2026.

[2] A. Beimel, T. Malkin, and N. Mazor. Structural lower bounds on black-box constructions of pseudorandom functions. In L. Reyzin and D. Stebila, editors, *Advances in Cryptology - CRYPTO 2024 - 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2024, Proceedings, Part V*, volume 14924 of *Lecture Notes in Computer Science*, pages 459–488. Springer, 2024.

[3] N. Bitansky and A. Degwekar. On the complexity of collision resistant hash functions: New and old black-box separations. In D. Hofheinz and A. Rosen, editors, *Theory of Cryptography - 17th International Conference, TCC 2019, Nuremberg, Germany, December 1-5, 2019, Proceedings, Part I*, volume 11891 of *Lecture Notes in Computer Science*, pages 422–450. Springer, 2019.

[4] N. Bitansky, A. Degwekar, and V. Vaikuntanathan. Structure vs. hardness through the obfuscation lens. In J. Katz and H. Shacham, editors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part I*, volume 10401 of *Lecture Notes in Computer Science*, pages 696–723. Springer, 2017.

[5] D. Boneh and R. Venkatesan. Breaking RSA may not be equivalent to factoring. In K. Nyberg, editor, *Advances in Cryptology - EUROCRYPT '98, International Conference on the*

Theory and Application of Cryptographic Techniques, Espoo, Finland, May 31 - June 4, 1998, Proceedings, volume 1403 of *Lecture Notes in Computer Science*, pages 59–71. Springer, 1998.

[6] E. Bresson, J. Monnerat, and D. Vergnaud. Separation results on the "one-more" computational problems. In T. Malkin, editor, *Topics in Cryptology - CT-RSA 2008, The Cryptographers' Track at the RSA Conference 2008, San Francisco, CA, USA, April 8-11, 2008. Proceedings*, volume 4964 of *Lecture Notes in Computer Science*, pages 71–87. Springer, 2008.

[7] J. Bronson, A. Juma, and P. A. Papakonstantinou. Limits on the stretch of non-adaptive constructions of pseudo-random generators. In Y. Ishai, editor, *Theory of Cryptography - 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28-30, 2011. Proceedings*, volume 6597 of *Lecture Notes in Computer Science*, pages 504–521. Springer, 2011.

[8] M. Fischlin and D. Schröder. On the impossibility of three-move blind signature schemes. In H. Gilbert, editor, *Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco / French Riviera, May 30 - June 3, 2010. Proceedings*, volume 6110 of *Lecture Notes in Computer Science*, pages 197–215. Springer, 2010.

[9] R. Gennaro, Y. Gertner, J. Katz, and L. Trevisan. Bounds on the efficiency of generic cryptographic constructions. *SIAM J. Comput.*, 35(1):217–246, 2005. doi: 10.1137/S0097539704443276. URL https://doi.org/10.1137/S0097539704443276.

[10] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *Journal of the ACM (JACM)*, 33(4):792–807, 1986.

[11] I. Haitner, A. Rosen, and R. Shaltiel. On the (im)possibility of Arthur–Merlin witness hiding protocols. In O. Reingold, editor, *Theory of Cryptography, 6th Theory of Cryptography Conference, TCC 2009, San Francisco, CA, USA, March 15-17, 2009. Proceedings*, volume 5444 of *Lecture Notes in Computer Science*, pages 220–237. Springer, 2009.

[12] T. Holenstein and M. Sinha. Constructing a pseudorandom generator requires an almost linear number of calls. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 698–707. IEEE Computer Society, 2012.

[13] R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In D. S. Johnson, editor, *Proceedings of the 21st Annual ACM Symposium on Theory of Computing, May 14-17, 1989, Seattle, Washington, USA*, pages 44–61. ACM, 1989.

[14] L. A. Levin. One-way functions and pseudorandom generators. In *Proceedings of the seventeenth annual ACM symposium on Theory of computing*, pages 363–365, 1985.

[15] E. Miles and E. Viola. On the complexity of non-adaptively increasing the stretch of pseudorandom generators. In Y. Ishai, editor, *Theory of Cryptography - 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28-30, 2011. Proceedings*, volume 6597 of *Lecture Notes in Computer Science*, pages 522–539. Springer, 2011.

[16] E. Miles and E. Viola. On the complexity of constructing pseudorandom functions (especially when they don't exist). *J. Cryptol.*, 28(3):509–532, 2015.

[17] R. Pass. Limits of provable security from standard assumptions. In L. Fortnow and S. P. Vadhan, editors, *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 109–118. ACM, 2011.

[18] R. Pass. Unprovable security of perfect NIZK and non-interactive non-malleable commitments. In A. Sahai, editor, *Theory of Cryptography - 10th Theory of Cryptography Conference, TCC 2013, Tokyo, Japan, March 3-6, 2013. Proceedings*, volume 7785 of *Lecture Notes in Computer Science*, pages 334–354. Springer, 2013.

[19] O. Reingold, L. Trevisan, and S. P. Vadhan. Notions of reducibility between cryptographic primitives. In M. Naor, editor, *Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Cambridge, MA, USA, February 19-21, 2004, Proceedings*, volume 2951 of *Lecture Notes in Computer Science*, pages 1–20. Springer, 2004.

[20] D. R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In K. Nyberg, editor, *Advances in Cryptology - EUROCRYPT '98, International Conference on the Theory and Application of Cryptographic Techniques, Espoo, Finland, May 31 - June 4, 1998, Proceeding*, volume 1403 of *Lecture Notes in Computer Science*, pages 334–345. Springer, 1998.

[21] P. Soni and S. Tessaro. On the query complexity of constructing prfs from non-adaptive prfs. In C. Galdi and V. Kolesnikov, editors, *Security and Cryptography for Networks - 12th International Conference, SCN 2020, Amalfi, Italy, September 14-16, 2020, Proceedings*, volume 12238 of *Lecture Notes in Computer Science*, pages 546–565. Springer, 2020.