Electronic Colloquium on Computational Complexity, Revision 3 of Report No. 61 (2018)

Indistinguishability by adaptive procedures with advice, and lower bounds on hardness amplification proofs

Aryeh Grinberg* Ronen Shaltiel† Emanuele Viola‡

May 3, 2018

Abstract

We study how well can q-query decision trees distinguish between the following two distributions: (i) $R = (R_1, \ldots, R_N)$ that are i.i.d. variables, (ii) $X = (R|R \in A)$ where A is an event s.t. $\Pr[R \in A] \geq 2^{-a}$. We prove two lemmas:

Forbidden-set lemma: There exists $B \subseteq [N]$ of size $\mathrm{poly}(a, q, \frac{1}{\eta})$ such that q-query trees that do not query variables in B cannot distinguish X from R with advantage η.

Fixed-set lemma: There exists $B \subseteq [N]$ of size $\mathrm{poly}(a, q, \frac{1}{\eta})$ and $v \in \{0,1\}^B$ such that q-query trees do not distinguish $(X|X_B = v)$ from $(R|R_B = v)$ with advantage η.

The first can be seen as an extension of past work by Edmonds, Impagliazzo, Rudich and Sgall (Computational Complexity 2001), Raz (SICOMP 1998), and Shaltiel and Viola (SICOMP 2010) to *adaptive* decision trees. It is independent of recent work by Meir and Wigderson (ECCC 2017) bounding the number of $i \in [N]$ for which there exists a q-query tree that predicts X_i from the other bits.

We use the second, fixed-set lemma to prove lower bounds on black-box proofs for hardness amplification that amplify hardness from δ to $\frac{1}{2} - \epsilon$. Specifically:

- Reductions must make $q = \Omega(\log(1/\delta)/\epsilon^2)$ queries, implying a "size loss factor" of q. We also prove the lower bound $q = \Omega(\log(1/\delta)/\epsilon)$ for "error-less" hardness amplification proofs, and for direct-product lemmas. These bounds are tight.
- Reductions can be used to compute Majority on $\Omega(1/\epsilon)$ bits, implying that black box proofs cannot amplify hardness of functions that are hard against constant depth circuits (unless they are allowed to use Majority gates).

Both items extend to pseudorandom-generator constructions.

These results prove 15-year-old conjectures by Viola, and improve on three incomparable previous works (Shaltiel and Viola, SICOMP 2010; Gutfreund and Rothblum, RANDOM 2008; Artemenko and Shaltiel, Computational Complexity 2014).

*Department of computer science, University of Haifa E-mail: akir94@gmail.com.

†Department of computer science, University of Haifa. E-mail: ronen@cs.haifa.ac.il. This research was supported by ISF grant 1628/17.

‡College of Computer and Information Science, Northeastern University, E-mail: viola@ccs.neu.edu

ISSN 1433-8092

1 Introduction

In this paper we develop tools to bound the ability of adaptive procedures that make few queries to distinguish between an i.i.d. distribution on which the procedure "receives small advice" and the original i.i.d. distribution. We then use these tools to prove tight lower bounds on hardness amplification proofs.

1.1 Adaptive procedures that receive small advice

Let $R = (R_1, \ldots, R_N)$ be a collection of i.i.d. indicator random variables. Suppose that "a bits of advice" are given about R. That is, let $A \subseteq \{0,1\}^N$ be an event with $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. We can think of X as "the way that R appears to an adversary that received a bits of information on R". We are interested in the following question:

How well can a decision tree making q queries distinguish between X and R?

For simplicity, let us focus on the case where each bit R_i is uniformly distributed. (In some sense, made precise later, this is w.l.o.g.). It is instructive to consider the following two examples:

Bad bits: Consider $A = \{r : r_1 = 0\}$, then $\Pr[R \in A] = 2^{-a}$ for $a = 1$, and R_1 is a random coin, whereas X_1 is the constant zero. In other words, R_1 and X_1 are far in statistical distance. Consequently, there exists a 1-query decision tree that distinguishes R from X with large advantage.

Pointer chasing: Let $N = \ell + 2^\ell$, and for $r \in \{0,1\}^N$, we write $r = (r^1, r^2)$ where $|r^1| = \ell$ and $|r^2| = 2^\ell$. We can interpret $r^1 \in \{0,1\}^\ell$ as a number $r^1 \in [2^\ell]$ and consider $A = \left\{r : r^2_{r^1} = 0\right\}$. Namely, that the bit that r^1 "points to" in r^2 is fixed to zero. Once again, $\Pr[R \in A] = 2^{-a}$ for $a = 1$. Note that an $(\ell+1)$-query decision tree P, that queries the bits of r^1 and "follows the pointer" to decide which query to ask in r^2, distinguishes R from X with large advantage.

The two examples point out a distinction between two types of "local procedures": Adaptive procedures are general decision trees, that can decide on their next query based on the answers to past queries. We say that a q-query decision tree P is *nonadaptive*, if there exists a set $Q \subseteq [N]$ of size q, and a function f_P such that $P(x) = f_P(x_Q)$ (namely, if all "computation paths" of P query the variables in S in some order).

It is interesting to note that a q-query nonadaptive decision tree cannot substantially distinguish R from X in the case of "pointer chasing" even if q approaches 2^ℓ, whereas an adaptive $(\ell+1)$-query decision tree can.

1.1.1 Settings in which X and R are indistinguishable by shallow decision trees

We would like to identify settings in which we can argue that adaptive/nonadaptive q-query decision trees cannot substantially distinguish X from R. Given the two examples above,

we need additional constraints. We discuss two settings, the second of which is introduced in this paper.

Forbidden sets: Here one forbids decision trees from querying variables in a certain *small* "forbidden set" $B \subseteq [N]$. One shows that decision trees that do not make queries in B cannot substantially distinguish R from $X = (R|R \in A)$.

Loosely speaking, this means that except for a few "damaged variables" one can assume that X is composed of i.i.d. variables (at least from the point of view of a shallow decision tree).

Fixed sets: Here one fixes the variables in a certain *small* "fixed set" $B \subseteq [N]$ to some value v, and considers the conditional distributions $R' = (R|R_B = v)$ and $X' = (X|X_B = v) = (R|R_B = v, R \in A)$. One shows that no decision tree can substantially distinguish X' from R', even if the tree queries variables in B.

Loosely speaking, this means that we can "get rid" of correlations between bits of X (at least from the point of view of a shallow decision tree) if we are willing to fix few "damaged variables".

In both cases, given integer parameters N, q, a, and an event A such that $\Pr[R \in A] \geq 2^{-a}$, we will want that B is of size $b = \text{poly}(a, q, \frac{1}{\eta})$ where $\eta > 0$ is measuring the required statistical distance. (It can be easily observed by extending the "bad bits example" that we cannot expect $b = o(\frac{a \cdot q}{\eta})$).

1.1.2 Past work on nonadaptive procedures

A well-known lemma states that if we have N i.i.d. random variables and we condition on an event that has not too small probability, then most variables are still close to uniform.

Lemma 1.1. *Let N, a be integers. Let $R = (R_1, \ldots, R_N)$ be i.i.d. indicator random variables, let $A \subseteq \{0,1\}^N$ be an event such that $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $O(a/\eta^2)$, such that for every $i \in [N] \setminus B$, R_i and X_i are η-close.*

Lemma 1.1 has found many applications in a wide variety of contexts, see for example the 12 references in [MW17]. In our terminology, Lemma 1.1 is a forbidden set lemma for $q = 1$. An extension of Lemma 1.1 to $q > 1$ was given by Shaltiel and Viola [SV10], and is stated next.

Lemma 1.2 ([SV10])**.** *Let N, a, q be integers. Let $R = (R_1, \ldots, R_N)$ be i.i.d. indicator random variables, let $A \subseteq \{0,1\}^N$ be an event such that $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $O(a \cdot q/\eta^2)$, such that for every $Q \subseteq [N] \setminus B$ of size q, R_Q and X_Q are η-close.*

Lemma 1.2 and so in particular Lemma 1.1 can already be obtained from the techniques in the 1991 paper [EIRS01]. A proof will also be given later in this paper. In our terminology, Lemma 1.2 is a forbidden set lemma for *nonadaptive* q-query decision trees. Namely, it says that for every *nonadaptive* q-query decision tree P that does not make queries in B, P does not distinguish X and R with advantage η.

The extension of Lemma 1.1 to larger q given by Lemma 1.2 has also found several applications. The application in [SV10] concerns the complexity of *hardness amplification proofs*. This application is also a main motivation for this work and is discussed in detail below in Section 1.2. Lemma 1.2 has also found application in *data-structure lower bounds*, see [Vio12, Vio09a].

A significant shortcoming of Lemma 1.2 is that it only applies to *non-adaptive* decision trees. As a consequence, several applications of this result are also proved only in the non-adaptive setting. For example, the bounds in [SV10] on the complexity of hardness amplification proofs only apply to non-adaptive procedures. Although some of the available hardness amplification proofs are indeed non-adaptive, others are not. This point is discussed further in Section 1.2.2.

In the area of data structures, some of the lower bounds obtained using Lemma 1.2 were later generalized to the adaptive setting [Vio12, PV10]. However, in some cases this generalization is not yet available. For example, [Vio09a] proves a non-adaptive lower bound for the *matching brackets* problem, and this is not known in the *adaptive* setting.

Therefore, it will be desirable to extend Lemma 1.2 to handle general, *adaptive* decision trees. In this paper we obtain such an extension. In fact, we shall prove two extensions. The first is closest to the setting of Lemma 1.2 and gives a "forbidden set". However when applying this version, several technical difficulties arise because of adaptive procedures that may query the forbidden set in complicated ways. So we prove a "fixed-set" extension, where these difficulties disappear. We then discuss our main application to hardness amplification proofs. We believe that the results in this paper will also be useful in the study of data structures.

1.1.3 A forbidden set lemma for adaptive decision trees

In this paper we prove a forbidden set lemma for *adaptive* decision trees.

Lemma 1.3 (Forbidden set lemma for adaptive decision trees)**.** *Let N, a, q be integers. Let $R = (R_1, \ldots, R_N)$ be i.i.d. indicator random variables, let $A \subseteq \{0,1\}^N$ be an event such that $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $O(\frac{a \cdot q^3}{\eta^3}) = poly(a, q, \frac{1}{\eta})$, such that for every q-query decision tree P that does not make queries in B, $P(R)$ and $P(X)$ are η-close.*

Later in this paper we also prove an extension of this lemma where the tree may query variables in B with small probability, see Corollary 3.4.

It is illustrative to note that in the case of the "pointer chasing" example, the forbidden set lemma must put (many of) the "pointer bits" in B (as otherwise an adaptive decision tree can distinguish by querying the "pointer bits").

Lemma 1.3 is independent of recent work by Meir and Wigderson, see Corollary 1.5 in [MW17], and the follow-up work [ST17]. They prove a result similar to Lemma 1.3 but for unpredictability rather than indistinguishability. Specifically, they show that every variable X_i except those in a set B_{MW} of $O(aq/\epsilon^3)$ variables has the following property: no adaptive decision tree making q queries to other variables can predict X_i with advantage more than ϵ over random guessing. We elaborate more on the difference between the works in Remark 3.10.

1.1.4 A fixed set lemma for adaptive decision trees

Forbidden set lemmas have the drawback that they guarantee nothing in case the decision tree does make queries in B. This is unavoidable, as can be seen by the "bad bits" and "pointer chasing" examples. In this paper we propose the idea of *fixed set lemmas* where X is further conditioned to the distribution X' by fixing the variables in some set B. A corresponding conditioning is applied to R to obtain R', whose bits are independent, $|B|$ of them are fixed, and the rest are unaltered. Then, even decision trees that make queries in B cannot distinguish R' and X'. We prove the following fixed set lemma for *adaptive* decision trees.

Lemma 1.4 (Fixed set lemma for adaptive decision trees)**.** *Let N, a, q be integers. Let $R = (R_1, \ldots, R_N)$ be i.i.d. indicator random variables, let $A \subseteq \{0,1\}^N$ be an event such that $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $\leq \frac{a \cdot q}{\eta^2} = poly(a, q, \frac{1}{\eta})$, and $v \in \{0,1\}^B$, such that for $R' = (R|R_B = v)$ and $X' = (X|X_B = v) = (R|R_B = v, R \in A)$, and every q-query decision tree P, $P(R')$ and $P(X')$ are η-close.*

This loosely means that if the application allows us to fix few bits of X, then we need not worry about trees that make queries in B.

We note that one can't derive a fixed set lemma from a forbidden set lemma by fixing the bits in B. For example, consider the "pointer chasing" example. A forbidden set lemma may choose B to be the "pointer bits" (namely $B = \{1, 2, \ldots, \ell\}$). However, if these bits get fixed, then the bit they point to is also fixed and not in B, and thus a 1-query decision tree can distinguish. By using pointer chasing with several layers, this example can be extended to show that even after many applications of a nonadaptive lemma, and fixing the bad bits, one does not obtain a fixed set lemma.

Remark 1.5. *Stefano Tessaro pointed out the similarity between Lemma 1.4 and results in cryptography related to* random oracles with auxiliary input*: Theorem 2 in [Unr07] (cf. Theorem 1 in [DGK17]) and Lemma 1 in [CDGS18]. These results seem incomparable to Lemma 1.4. Lemma 1 in [CDGS18] roughly proves that the distribution X in Lemma 1.4 is indistinguishable by shallow decision trees from a* convex combination *of distributions of the form $R|R_B = v$. Instead, in Lemma 1.4 we show that the distribution $X|X_B = v$ is indistinguishable from* the single *distribution $R|R_B = v$. Another difference is that actually Lemma 1 in [CDGS18] is not stated for X but rather for an "average conditioning" of R, modeled as an*

adversary who is given $f(R)$, for a function f with bounded output length, and can make few queries to R.

1.2 Hardness amplification

Hardness amplification is a technique to transform "hard functions" into "harder functions". It is closely related to error-correcting codes and plays a fundamental role in complexity theory, derandomization, and cryptography. We give a brief survey of the aspects that are most relevant to this work. For additional background we refer the reader to Chapter 17 "Hardness Amplification and Error Correcting Codes" in the textbook [AB09], and to the discussion in [SV10].

Hardness amplification results in the complexity theoretic literature have the following black-box form: Given a function f, there is a "construction map" that produces a function Con_f. The proof provides a "reduction" Red, that converts an "adversary" D that "breaks" the (strong hardness) of Con_f, into an "adversary" C that "breaks" the (weaker hardness) of the original function. A precise definition follows:

Definition 1.6 (Black-box hardness amplification)**.** [1] *We say that two functions h_1, h_2 on the same finite domain W, p-agree if $\Pr_{X \leftarrow W}[h_1(X) = h_2(X)] \geq p$.*

A $\delta \rightarrow (\frac{1}{2} - \epsilon)$ black-box hardness amplification with input lengths k and n, and list size 2^a is a pair $(\mathsf{Con}, \mathsf{Red})$ such that:

- *A* construction Con *is a map from functions $f : \{0,1\}^k \rightarrow \{0,1\}$ to functions $\mathsf{Con}_f : \{0,1\}^n \rightarrow \{0,1\}$.*
- *A* reduction Red *is an oracle circuit $\mathsf{Red}^{(\cdot)}(x, \alpha)$ that accepts two inputs: $x \in \{0,1\}^k$ and $\alpha \in \{0,1\}^a$ (we call α a "nonuniform advice string"). Red also receives oracle access to a function $D : \{0,1\}^n \rightarrow \{0,1\}$.*

We require that for all functions $f : \{0,1\}^k \rightarrow \{0,1\}$ and $D : \{0,1\}^n \rightarrow \{0,1\}$ such that D $(\frac{1}{2} + \epsilon)$-agrees with Con_f, there exists $\alpha \in \{0,1\}^a$ such that $\mathsf{Red}^D(., \alpha)$ $(1 - \delta)$-agrees with f.

Note that for if $\delta < 2^{-k}$ then it follows that $\mathsf{Red}^D(., \alpha)$ 1-agrees with f, which means $\mathsf{Red}^D(., \alpha) = f$.

A reduction (or proof) is nonadaptive *if the queries Red makes to the oracle are non-adaptive.*

The following specific "construction maps" are extensively studied in the literature:

Yao's XOR Lemma: Let $n = t \cdot k$ for a parameter t and $\mathsf{Con}_f(x_1, \ldots, x_t) = f(x_1) \oplus \ldots \oplus f(x_t)$. This is the "mother" of all hardness amplifications, dating back to oral presentations by Yao in the 80's, cf. [GNW95]. To give an example setting of parameters,

[1] In several papers, including [SV10], instead of a single reduction Red that receives an a bit long advice string α, they define a set of reduction circuits of size 2^a. The two definitions are equivalent. We also remark that we allow the map Con to be arbitrary (with no complexity restrictions) whereas some previous work placed a bound on its complexity. This only makes our results stronger.

one can start with δ constant and then the hardness parameter ϵ decays exponentially with t.

Direct product Lemma: Let $n = t \cdot k$ and $\mathsf{Con}_f(x_1, \ldots, x_t) = (f(x_1) \circ \ldots \circ f(x_t))$. Such results are called "Concatenation Lemma" or "Direct product lemma". Note that here, Con_f is nonboolean and outputs t bits, and consequently $\frac{1}{2} + \epsilon$ should be replaced with $2^{-t} + \epsilon$.

Local list-decodable codes: Let $K = 2^k$, $N = 2^n$ and $\mathsf{Con}_f(y) = \mathsf{Enc}(f)_y$ where $\mathsf{Enc} : \{0,1\}^K \to \{0,1\}^N$ is an encoding map for a binary error-correcting code, and we view the function $f : \{0,1\}^k \to \{0,1\}$ as a K-bit string that is the truth table of f. In fact, for $\delta = 0$, it is necessary that Enc is a "locally decodable, list-decodable code", and Red is a "local list decoding algorithm". The connection between such codes and hardness amplification is established in [STV01], where it is also said that is was observed independently by other researchers (see [STV01]).

We note that black-box hardness amplification indeed lets us amplify hardness. In short, this is because if there exists a small circuit D that $(\frac{1}{2} + \epsilon)$-agrees with Con_f then the reduction gives a procedure of the form $\mathsf{Red}^D(., \alpha)$ which $(1 - \delta)$-agrees with f. Now, if $\mathsf{Red}^D(., \alpha)$ is also a small circuit then this gives a larger circuit $C(.) = \mathsf{Red}^D(., \alpha)$, composed of Red and q of oracle copies of D (where q is the number of queries made by the reduction) that $(1 - \delta)$-agrees with f. If the latter is impossible, because we started with a function f that is sufficiently hard, we have reached a contradiction, which means that D does not exist.

It is evident from this argument that to obtain hardness against a circuit class $\mathcal{C}$ one needs to start from hardness against the larger circuit class $\mathsf{Red}^{\mathcal{C}}$. Thus a natural question, and a main focus of this paper, is what is the complexity of Red.

Red requires $\mathbf{TC^0}$. Fifteen years ago Viola [Vio04, Vio06] made several conjectures regarding the complexity of hardness amplification. Informally, he conjectured that the smallest circuit class that can prove hardness amplification with $\epsilon = 1/n$ is the class TC^0 of constant-depth circuits with majority gates. This is problematic because (1) lower bounds against TC^0 are a long-standing open problem in circuit complexity, (2) the "Natural Proofs" barrier [RR97, NR04, MV15] indicates that achieving such bounds will be very difficult, and (3) average-case hardness with $\epsilon = 1/n$ is required for several important applications such as the construction of pseudorandom generators a la Nisan [Nis91], and deriving further lower bounds via the so-called "discriminator lemma" [HMP+93].

This is especially frustrating because for several important circuit classes, such as constant-depth circuits with mod p gates for prime p, or constant-depth circuits with a limited number of Majority gates, explicit hard functions are known, and in fact even functions with hardness $1/2 - o(1)$. However, we can't use hardness amplification to amplify the hardness to the point where we could use it to construct pseudorandom generators or infer additional lower bounds. For classes such as constant-depth circuits with parity gates the best known

pseudorandom generators have very poor seed length of the form $m(1-o(1))$ where m is the output length [FSUV13].

More specifically, [Vio06] conjectures that every circuit class that can prove hardness amplification to hardness $1/2-1/\epsilon$ can compute majority on $\Omega(1/\epsilon)$ bits, and must use $\Omega(1/\epsilon)$ queries. A more precise conjecture on the number of queries is $\Omega(\log(1/\delta)/\epsilon^2)$. Special cases of these conjectures are proved in [Vio06] and in subsequent works [LTW11, SV10, GR08, AS14], the last three of which are incomparable as they restrict the hardness amplification in different ways. Previous work is discussed in more detail in Section 1.2.2 below.

In this paper we prove the conjectures, thus in particular improving on three incomparable works [SV10, GR08, AS14].

1.2.1 Our results on hardness amplification proofs

First, we prove a tight query lower bound. Showing that reductions in black-box hardness amplification must make at least q queries translates to saying that when transforming a function f that is hard against circuits of size s to a function Con_f that is harder against circuits of size s', then $s' \leq s/q$. This means that a "size loss" is unavoidable in black-box hardness amplification.

Theorem 1.7 (Lower bound on the number of queries)**.** *There exist constants $\delta_0, \nu > 0$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to (\frac{1}{2} - \epsilon)$ black-box hardness amplification with input lengths k and n, and list size 2^a. Assume that:*

- *$\mathsf{Red}^{(\cdot)}$ is a size r oracle circuit, that makes at most q queries.*
- *$n, a, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$ and $2^{-2k} \leq \delta \leq \delta_0$.*

Then $q = \Omega(\log(1/\delta)/\epsilon^2)$.

The parameters achieved by Theorem 1.7 are tight up to constants, matching the parameters obtained by Klivans and Servedio [KS03] for the XOR lemma, using Impagliazzo's hard-core lemma [Imp95]. Note that the special case of $\delta = 2^{-2k}$ (which is the same as $\delta = 0$) captures worst-case to average-case hardness amplification, and then the lower bound is $q = \Omega(k/\epsilon^2)$.

We then prove that hardness amplification proofs require majority.

Theorem 1.8 (Hardness amplification proofs require majority)**.** *There exist constants $\delta_0, \nu > 0$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to (\frac{1}{2} - \epsilon)$ black-box hardness amplification with input lengths k and n, and list size 2^a. Assume that:*

- *$\mathsf{Red}^{(\cdot)}$ is a size r oracle circuit of depth $d = O(\log(1/\epsilon))$ (over a set of gates that includes the standard boolean gates with unbounded fan-in)*
- *$n, a, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$ and $\delta \leq \delta_0$.*

Then there exists a circuit C of size $\mathrm{poly}(r)$, and depth $O(d)$ that uses the gates allowed to Red, and computes the majority function on inputs of length $\Omega(1/\epsilon)$.

In particular, if $\epsilon = 1/n$ and $\mathsf{Red}(.,\alpha)$ are constant-depth circuits with And, Or, and Parity gates of unbounded fan-in, and Not gates, then its size must be $2^{n^{\Omega(1)}}$ by the known lower bounds [Raz87].

Theorem 1.8 is tight in the sense that Klivans [Kli01] observed that there are reductions that can be implemented by a constant-depth circuit with only one Majority gate. For an exposition of a simplification of Klivans' argument, due to Klivans and Vadhan, see Lectures 6 and 7 in [Vio09b].

Lower bounds on local decoding algorithms for list-decodable codes. The aforementioned results of [STV01] show that a $0 \to (\frac{1}{2} - \epsilon)$ black-box hardness amplification $(\mathsf{Con}, \mathsf{Red})$ with list-size 2^a yields $(\frac{1}{2} - \epsilon, 2^a)$-list decodable code, with an encoding map $\mathsf{Enc} : \{0,1\}^{2^k} \to \{0,1\}^{2^n}$, given by $\mathsf{Enc}(f) = \mathsf{Con}_f$ (here functions are identified with their truth tables) and Red is a "local list-decoding algorithm".[2] In this terminology, our results give lower bounds on the complexity, and on the number of queries of local list decoding algorithms for list decodable codes.

Limitations on direct-product lemmas and decoding from erasures. Another extensively studied construction map is the nonboolean map $\mathsf{Con}_f(x_1, \ldots, x_t) = (f(x_1), \ldots, f(x_t))$, and proofs for this map are known as *concatenation lemmas* or *direct product lemmas*. Our techniques also apply in this setting, and give the lower bound of $q = \Omega(\log(1/\delta)/\epsilon)$ in this case. This is tight [IJKW10]. We also consider hardness amplification in the "error-less" setting, a study initiated by Bogdanov and Safra [BS07]. We prove a query lower bound of $q = \Omega(\log(1/\delta)/\epsilon)$, which matches a construction by Watson [Wat15]. These results follow by proving lower bounds on the more general setup of "basic hardness amplification" considered by Artemenko and Shaltiel [AS14] and Watson [Wat15]. It should be noted that these proofs can be implemented by constant-depth circuits, and so computing majority is not required. This was observed in [Vio06] for the direct product lemma, and can be verified by inspection for example of [Wat15] for error-less hardness amplification, see [AS14] for discussion.

Limitations on black-box PRG constructions. Our techniques also apply to "hardness versus randomness", that is to constructing pseudorandom generators from hard functions. Loosely speaking, we show the same lower bounds for constructing pseudorandom generators with error ϵ as for amplifying hardness to $1/2 - \epsilon$.

1.2.2 More on related work, and why previous negative results do not capture all available techniques

As mentioned earlier, previous works proved special cases of the conjectures by restricting the hardness amplification in various ways. One way to restrict was requiring that the proof Red is *non-adaptive*, that is, only makes non-adaptive query to the oracle. The restriction

[2] A function $\mathsf{Enc} : \{0,1\}^K \to \{0,1\}^N$ is (ρ, ℓ)-list decodable if for every $y \in \{0,1\}^N$, there are at most ℓ elements $x \in \{0,1\}^K$ such that $\mathsf{Enc}(x)$ ρ-agrees with y.

to non-adaptive reductions is severe also because there exist hardness amplification proofs that do exploit adaptivity, such as [SU05, Uma03, GGH+07, GR08]. We note that the proofs of [SU05, Uma03] use adaptive reductions for a slightly different task: converting hard functions to pseudorandom generators. As mentioned earlier, our lower bounds also apply to pseudorandom-generator constructions.

Another way to restrict was to limit the length of the advice string a, that is, considering reductions Red that are uniform. Some reductions for hardness amplification in the literature use as little as $a = O(\log(1/\epsilon))$ bits of nonunifomity [IJKW10], but most reductions use $a = \text{poly}(1/\epsilon)$ (or more) bits of nonuniformity.

Viola [Vio06] proves that majority is necessary for hardness amplification proofs based on the Hadamard code, or on the Reed-Muller code concatenated with Hadamard. This result applies to adaptive proofs, however only if the list size a is small ($a = O(\log(1/\epsilon))$). Viola [Vio06] also proves the query lower bound $q \geq \Omega(\min\{1/\epsilon, k/\log k\})$. This result can handle large a, even $a = 2^{\Omega(k)}$, but only applied to *non-adaptive reductions*. These results were subsumed by the next three works.

Shaltiel and Viola [SV10] proved limitations on *non-adaptive reductions*. Their results are identical to Theorem 1.7 and Theorem 1.8, except that it only handles *nonadaptive* reductions. Our results extend theirs by allowing for adaptive reductions.

Gutfreund and Rothblum [GR08] extend the above result about Reed-Muller code concatenated with Hadamard to any code. That is, they can handle any hardness amplification but they require a to be small. Our results extend theirs by allowing large a.

Artemenko and Shaltiel [AS14] proved a lower bound of $q = \Omega(1/\epsilon)$ for the number of queries (allowing both adaptivity and large advice). Their approach is to consider a hardness amplification variant that corresponds to codes that are locally list-decodable from *erasures*. They called this setup "basic hardness amplification." Our results extend their work and give a (tight) lower bound of $q = \Omega(\log(1/\delta)/\epsilon)$ in that setup.

Watson [Wat15] considered an intermediate setup of "errorless amplification" and obtained a tight lower bound of $q = \Omega(\log(1/\delta)/\epsilon)$ on *nonadaptive* reductions (by reducing to the lower bound of [SV10]). Our results extend this lower bound to *adaptive* reductions.

Applebaum, Artemenko, Shaltiel and Yang [AASY15] considered a more powerful class of reductions that are allowed to be *nondeterministic* oracle circuits. They prove limitations on such reductions for the case that $\epsilon \ll 1/r$. These results are incomparable to ours.

2 Techniques

In this section we aim to give an informal overview of our arguments, trying to sum up the main ideas. The technical section of this paper includes full proofs, and does not build on this informal overview.

2.1 The forbidden and fixed set lemma

Our proofs of the new forbidden set lemma and fixed set lemma use basic information theory (as is the case for the proofs of Lemma 1.1 and Lemma 1.2). For simplicity let us focus on the case where $R_1, \ldots, R_N$ are i.i.d. and uniformly distributed. (In Section 3 we observe that it is indeed sufficient to study the case of the uniform distribution to obtain results on arbitrary i.i.d. variables).

The setting for Lemmas 1.1, 1.2, 1.3 and 1.4 is the following: Let $X = (R|R \in A)$ for A such that $\Pr[R \in A] \geq 2^{-a}$. We aim to show that X and R cannot be distinguished from uniform by shallow decision trees. It is immediate that $H(X) \geq N - a$, where H is the Shannon entropy function. All previous works in this area eventually connect entropy and statistical distance using Pinsker's inequality, which guarantees that a distribution Y over n bits is η-close to uniform if $H(Y) \geq n - \eta^2$.

2.1.1 Previous proofs for the nonadaptive case

It is instructive to first explain the argument used in Lemma 1.1 and Lemma 1.2, and point out where this approach fails for adaptive decision trees. The proof of Raz [Raz98] for Lemma 1.1 works by first using the chain rule for entropy:

$$N - a \leq H(X) = \sum_{i \in [N]} H(X_i | X_1, \ldots, X_{i-1}).$$

Then, choose $\alpha = \eta^2$, and let B be the set of indices i such that $H(X_i|X_1, \ldots, X_{i-1}) < 1 - \alpha$. By a Markov argument, there are at most $a/\alpha = a/\eta^2$ such "weak" i. For every $i \in [N] \setminus B$,

$$H(X_i) \geq H(X_i|X_1, \ldots, X_{i-1}) \geq 1 - \alpha = 1 - \eta^2,$$

which by Pinsker's inequality gives that X_i is η-close to uniform.

An extension of this argument was used in [SV10], where they choose $\alpha = \eta^2/q$ and for $i_1, \ldots, i_q \notin B$,

$$H(X_{i_1}, \ldots, X_{i_q}) = \sum_{j \in [q]} H(X_{i_j} | X_{i_1}, \ldots, X_{i_{j-1}}) \geq \sum_{j \in [q]} H(X_{i_j} | X_1, X_2, \ldots, X_{i_j - 1}) \geq q \cdot (1 - \alpha) = q - \eta^2,$$

which by Pinsker's inequality gives that $(X_{i_1}, \ldots, X_{i_q})$ is η-close to uniform.

It is instructive to consider that in the pointer chasing example, this argument produces $B = \emptyset$ and does not identify the indices of the pointer. Loosely speaking, this means that the criteria used by the proofs above for finding "bad indices" and placing them in B isn't suited for adaptive decision trees.

2.1.2 The forbidden set lemma

We explain the argument for Lemma 1.3. We need to come up with a criteria for selecting indices that does identify the "pointer bits" in the pointer chasing example. We use the

following idea (inspired by a similar argument from [EIRS01]). We say that an $i \in [N]$ is α-*weak* if $H(X_i|X_{[N]\setminus\{i\}}) < 1-\alpha$. A key observation is that this criteria (which is less stringent than the one used above) does identify the bits of the pointer in the pointer chasing example. Moreover, we can bound the number of α-weak bits, by the following iterative process: while there is an α-weak bit i', remove it, place it in B and continue. In an iteration of this process, by the chain rule:

$$H(X_{[N]\setminus\{i'\}}) = H(X) - H(X_{i'}|X_{[N]\setminus\{i'\}}) \geq N - a - (1-\alpha) = (N-1) - (a-\alpha),$$

and so in each iteration the gap between the bit-length of X and its entropy decreases by α. The initial gap is a, and so, we can have at most a/α iterations, as entropy is bounded above by bit-length. We will choose $\alpha = \text{poly}(\eta/q)$, so that after the last iteration we have a "forbidden set" B of size $b \leq a/\alpha = \text{poly}(a, q, 1/\eta)$.

To conclude we need to show that if an adaptive q-query decision tree P that does not make queries in B distinguishes X from R with advantage η, then there exists some $i' \in [N] \setminus B$ that is not α-weak (which gives a contradiction). Essentially, we use the "distinguisher to predictor hybrid argument" [GM84, BM84, Yao82] to obtain an index i' such that the bit $X_{i'}$ can be predicted from $X_{[N]\setminus\{i'\}}$ with large advantage over random guessing, showing that i' is α-weak.

2.1.3 The fixed set lemma

We explain the argument for Lemma 1.4. The proof will follow the same overall approach of the forbidden set lemma. However, this time, we will show that if a q-query decision tree distinguishes X from uniform, then we can fix a few bits of X, and reduce the gap between its bit-length and its entropy.

More precisely, we will show that if there exists a decision tree P that distinguishes X from R with advantage η, then for $\alpha = \eta^2$, there exist $i_1, \ldots, i_q \in [N]$, and $v_1, \ldots, v_q \in \{0,1\}$ such that:

$$H(X|X_{i_1} = v_1, \ldots, X_{i_q} = v_q) \geq (N-q) - (a-\alpha). \tag{1}$$

For this purpose, let $I = (I_1, \ldots, I_q)$ be the indices of the variables queried by P on input X. Note I is a random variable that is a function of X. By the chain rule we have that:

$$H(X) = H(X, X_{I_1}, \ldots, X_{I_q}) = H(X_{I_1}, \ldots, X_{I_q}) + H(X|X_{I_1}, \ldots, X_{I_q})$$

This gives that:

$$H(X|X_{I_1}, \ldots, X_{I_q}) = H(X) - H(X_{I_1}, \ldots, X_{I_q}) \geq N - a - (q-\alpha) = (N-q) - (a-\alpha),$$

where the inequality follows by Pinsker's because the answers $(X_{I_1}, \ldots, X_{I_q})$ are not η-close to uniform. Equation (1) now follows by an averaging argument that fixes X_I and hence I.

Equation (1) gives that we are able to fix q variables, and decrease the gap between the bit-length of X and its entropy by α. Once again, this gap is initially less than a, and so, this can happen at most a/α times. Consequently, by iteratively applying this process, we end up with a distribution where we fixed at most $q \cdot a/\alpha = \text{poly}(a, q, 1/\eta)$ bits to some specific values. The final distribution has that the bits that we did not fix cannot be distinguished from uniform by a q-query decision tree. The precise argument appears in Section 3.

2.2 Lower bounds on hardness amplification

Past work of Viola [Vio06] and Shaltiel and Viola [SV10] provides a framework that can be used in conjunction with lemmas of the type discussed in the previous section to obtain lower bounds on black-box hardness amplification. Theorem 1.7 (on the number of queries) directly follows by extending the approach of [SV10] using Lemma 1.4 (instead of Lemma 1.2). Proving Theorem 1.8 (on "hardness amplification implies majority") raises additional difficulties that do not come up in the nonadaptive case. We start by a high-level explanation of the approach of [SV10].

2.2.1 The Zoom lemma

Let Noise_p^N denote the distribution of N i.i.d. indicator random variables, where each is one with probability p. The approach of [SV10] is to show a "Zoom lemma" saying that: under certain conditions, a black-box hardness amplification $(\mathsf{Con}, \mathsf{Red})$ implies a procedure P on $N = 2^n$ variables that distinguishes $\mathsf{Noise}_{\frac{1}{2}}^N$ from $\mathsf{Noise}_{\frac{1}{2}-\epsilon}^N$ with probability roughly $1 - \delta$. The complexity of this procedure is inherited by the complexity of Con and Red specifically:

- If Red makes at most q queries, then P can be implemented by a q-query decision tree.
- We can view an element in $\{0,1\}^N$ as a function $D : \{0,1\}^n \to \{0,1\}$. In this notation, the distinguishing procedure P is an oracle procedure $P^{(\cdot)}$ that receives oracle access to D that is chosen from either $\mathsf{Noise}_{\frac{1}{2}}^N$ or $\mathsf{Noise}_{\frac{1}{2}-\epsilon}^N$. Using this terminology, $P^D = \mathsf{Red}^{\mathsf{Con}_{f \oplus D}}(x, \alpha)$ for some specific function f, input x and advice string α that are chosen in the proof.

Our first step is to use our new tools to prove a zoom lemma for adaptive reductions. The lemma appears in Section 4. A side benefit of our new approach is that our new tools simplify the proof of the zoom lemma (even in the nonadaptive case).

The argument for the zoom lemma. The high level idea of the proof of the zoom lemma is to fix some function $f : \{0,1\}^k \to \{0,1\}$ and consider the behavior of the reduction when given oracle access to $\mathsf{Con}_f \oplus \mathsf{Noise}_{\frac{1}{2}}^N$ and $\mathsf{Con}_f \oplus \mathsf{Noise}_{\frac{1}{2}-2\epsilon}^N$. In the first case, the noise completely masks Con_f and the reduction receives no information about f. This means that for a random f, Red is unlikely to 0.51-agree with f. In the second case, for any f, $\mathsf{Con}_f \oplus \mathsf{Noise}_{\frac{1}{2}-2\epsilon}^N$ is (w.h.p.) an oracle on which there exists α (which can be an arbitrary function of $\mathsf{Noise}_{\frac{1}{2}-2\epsilon}^N$) on which $\mathsf{Red}(\cdot, \alpha)$ needs to $(1 - \delta)$-agree with f. This intuitively means that the procedure Red can distinguish $\mathsf{Noise}_{\frac{1}{2}-2\epsilon}^N$ from uniform when it receives a "advice bits". Thus, in order to prove the zoom lemma, it is sufficient to show that Red cannot distinguish $R = \mathsf{Noise}_{\frac{1}{2}-2\epsilon}^N$ from $X = (R|R \in A)$ where A is an event of probability 2^{-a}. This is the setup considered in Section 1.1.

Lower bound on the number of queries Theorem 1.7 immediately follows from the first item in the zoom lemma, as a q query decision tree that distinguishes $\mathsf{Noise}^N_{\frac{1}{2}}$ from $\mathsf{Noise}^N_{\frac{1}{2}-\epsilon}$ with probability $1-\delta$, must make $\Omega(\log(1/\delta)/\epsilon^2)$ queries, cf. [SV10]. The precise statement appears in Section 5.1.

Hardness amplification proofs require majority Viola [Vio06] and Shaltiel and Viola [SV10], used an idea of Sudan (see discussion in [Vio06]) to give a reduction from the task of computing Majority on $\frac{1}{\epsilon}$ bits, to the task of distinguishing $\mathsf{Noise}^N_{\frac{1}{2}}$ from $\mathsf{Noise}^N_{\frac{1}{2}-\epsilon}$. For constant distinguishing advantage, this reduction transforms a constant depth distinguisher into a constant depth circuit (of polynomially related size) that computes Majority. Thus, in order to obtain Theorem 1.8 we need to simulate the computation $\mathsf{Red}^{\mathsf{Con}_{f\oplus D}}(x,\alpha)$ (from the second item of the zoom lemma) by a small constant depth circuit. In this computation, x, α, and f are fixed, while D is not.

Simulating this computation raises another difficulty: it is not clear how to compute Con_f. Although in some cases this can be done (see Footnote 6) in general it is not clear. This problem is even more pronounced in our extensions to pseudorandom-generator construction, where the relevant oracle is the NP function which checks if the string is in the support of the generator.

To overcome this difficulty we adapt an idea of [GR08] which removes the need to compute Con_f. Roughly, using a hybrid argument we can arrange so that there exists a depth i in the computation of $\mathsf{Red}^{\mathsf{Con}_{f\oplus D}}(x,\alpha)$ so that the queries at depth $< i$ have a fixed answer (where Con_f can be hardwired), and the queries at depth $> i$ have a completely uniform answer (independent of Con_f) . This holds however only for queries not in the fixed set B: we note that even with this idea we need a fixed-set lemma, as otherwise we do not know how to control the query answers at depth larger than i.

Pseudorandom-generator constructions. The above ideas can be extended in a rather straightforward way to pseudorandom-generator constructions. Roughly, the oracle $\mathsf{Con}_f \oplus D$ is replace by the oracle $\mathsf{Dist}_f \oplus D$ where Dist_f is the indicator function of the support of the pseudorandom generator.

Organization. In Section 3 we prove our forbidden and fixed set lemmas. In Section 4 we state and prove the zoom lemma. In Section 5 we prove our lower bounds on hardness amplification proofs and pseudorandom generator constructions. In particular, we prove theorems 1.7 and 1.8 from the introduction. We conclude in Section 6 with a discussion of open problems.

3 Forbidden and fixed set lemmas for adaptive decision trees

In this section we state and prove the forbidden set lemma (Lemma 1.3) and the fixed set lemma (Lemma 1.4) for the case of variables over an alphabet that is not necessarily binary. The proofs rely on the notion of Shannon entropy $H(X)$ of a random variable X, defined as $H(X) := \sum_x \Pr[X = x] \cdot \log(1/\Pr[X = x])$, and the related notion of conditional entropy $H(X|Y)$ of a random variable X conditioned to another random variable Y, defined as $H(X|Y) := E_{y \in Y} H(X|Y = y)$ (cf. [CT06, Chapter 2]). We list next a few standard properties of entropy that we will use extensively in this section.

Lemma 3.1. *Entropy satisfies the following.*

1. Chain rule: *For any random variables X and Y we have $H(X, Y) = H(X) + H(Y|X)$ [CT06, Theorem 2.5.1].*
2. Conditioning reduces entropy: *For any random variables X, Y, Z we have $H(X|Y) \geq H(X|Y, Z)$.*
3. High entropy implies near-uniformity (Pinsker's inequality): *Let V be a random variable taking values in a set S and suppose that $H(V) \geq \log |S| - \alpha$; then V is $c \cdot \sqrt{\alpha}$-close to uniform over S, where $c = \frac{1}{\sqrt{2}} < 1$. [CK82, Chapter 3; Exercise 17].*

3.1 Decision trees with variables in a finite alphabet

We consider a general case in which the N variables of a decision tree are not boolean, but rather from a finite alphabet Σ. We call such decision trees Σ^N-decision trees. Each internal node in such a decision tree has $|\Sigma|$ children, corresponding to each of the possible $|\Sigma|$ answers to the value of the queried variable.

Definition 3.2 (adaptive and non-adaptive decision trees)**.** *Let Σ be a finite alphabet, and V be some finite set. (We think of V as a set of variables, and typically $V = [N]$ for some integer N. We think of the set Σ^V as the set of inputs to a decision tree.)*

*We say that a function $P : \Sigma^V \to O$ (for some finite set O) is implemented by a q-*query Σ^V*-*decision-tree *if there exists a depth q decision tree, where each internal node is labeled by a "variable" $i \in V$, and has $|\Sigma|$ children (each labeled by an element of Σ), and leaves are labeled by elements of O. Every $z \in \Sigma^V$, defines a path from the root to a leaf (in the obvious way), and $P(z)$ is the label of that leaf.*

In the special case that $V = [N]$ we call the tree a Σ^N-decision tree.

A decision tree is nonadaptive *if exists a sequence of q distinct $y_1, \ldots, y_q \in V$ such that every path of the decision trees queries these variables in this order, and note that in that case there exists some function $f_P : \Sigma^q \to O$, such that $P(z) = f_P(z_{y_1}, \ldots, z_{y_q})$.*

In addition to the added generality, moving to a large alphabet Σ, has the advantage that the forbidden set lemma and fixed set lemma for i.i.d. variables (chosen according to

a distribution that isn't necessarily uniform) will follow by proving the special case where each of the i.i.d. variables is uniform over a sufficiently large alphabet Σ. This will allow us to concentrate on distances to the *uniform distribution*, which by Pinsker's lemma (Lemma 3.1) is reduced to understanding the entropy of distributions.[3]

3.2 The forbidden set lemma

We restate lemma 1.3 for decision trees over some finite alphabet Σ (rather than for Boolean alphabet).

Lemma 3.3 (Forbidden set lemma)**.** *Let N, a, q be integers. Let $R = (R_1, \ldots, R_N)$ be i.i.d. random variables where each one is over some finite set Σ, let $A \subseteq \Sigma^N$ be an event such that $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $\leq \frac{a \cdot q^3}{\eta^3} = poly(a, q, \frac{1}{\eta})$, such that for every q-query Σ^N-decision tree P that does not make queries in B, $P(R)$ and $P(X)$ are η-close.*

Note that the R_i may not be uniform over Σ. We can state the following corollary in which we allow P to make queries in B, as long as the probability that $P(X)$ makes a query in B is small. Note that we don't have to explicitly require that the probability that $P(R)$ makes a query in B is small.

Corollary 3.4. *Let N, a, q be integers. Let $R = (R_1, \ldots, R_N)$ be i.i.d. random variables (where each one is over some finite set Σ), let $A \subseteq \Sigma^N$ be an event such that $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. For every $\eta > 0, \beta > 0$, there exists a set $B \subseteq [N]$ of size $poly(a, q, \frac{1}{\eta})$, such that for every q-query Σ^N-decision tree P, if $P(X)$ makes a query in B with probability at most β then $P(R)$ and $P(X)$ are $(\eta + \beta)$-close.*

Proof. (of Corollary 3.4) Let B be the set guaranteed by Lemma 3.3. Let P' be a randomized decision tree that on input $z = (z_1, \ldots, z_N)$, simulates $P(z)$ with the following modification: Whenever P wants to make a query to z_i for $i \in B$, P' does not make the query, and instead samples a fresh, uniform answer A distributed like R_i, and supplies it to P. It immediately follows that $\Pr[P(X) \neq P'(X)] \leq \beta$, and that $P(R)$ is distributed exactly like $P'(R)$.

Assume towards a contradiction that $|\Pr[P(X) = 1] - \Pr[P(R)]| > \beta + \eta$. By the triangle inequality $|\Pr[P'(X) = 1] - \Pr[P'(R) = 1]| > \eta$, and by an averaging argument the randomness of P' can be fixed to yield a deterministic q-query decision tree P'' with the same property. Note that P'' never makes a query in B. This contradicts Lemma 3.3. □

Lemma 3.3 follows directly from the following lemma (that can be thought of as the special case in which variables R_i are uniformly distributed over $\Sigma = \{0,1\}^m$).

[3] We mention that several past works, such as Raz [Raz98], took a different approach and used Pinsker's lemma in a more general setting where it connects the statistical distance between two distributions to the informational divergence (also know as Kullback Leibler distance) between the two distributions. The two approaches are equivalent, and we choose the former, to make the proofs more transparent.

Lemma 3.5 (Entropy version of forbidden set lemma)**.** *Let N, m, a, q be integers. Let $Z = (Z_1, \ldots, Z_N)$ be a distribution over $(\{0,1\}^m)^N$ with $H(Z) \geq N \cdot m - a$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $O(\frac{aq^3}{\eta^3}) = poly(a, q, \frac{1}{\eta})$ such that for every q-query $(\{0,1\}^m)^N$-decision tree P, that does not make queries in B, $P(Z)$ is η-close to $P(U_{N \cdot m})$.*

We are now ready to derive Lemma 3.3 from Lemma 3.5.

Proof. (of Lemma 3.3) Let $R = (R_1, \ldots, R_N)$ be i.i.d. random variables where each one is distributed according to some distribution Q on Σ. There exists an integer m and $g : \{0,1\}^m \to \Sigma$, such that $g(U_m) = Q$.[4] Consider the probability space in which $Z = (Z_1, \ldots, Z_N)$ is chosen from $U_{N \cdot m}$, and for every $i \in [N]$, $R_i = g(Z_i)$. We can imagine that R is chosen this way, and interpret $A \subseteq \Sigma^N$ as

$$A' = \{(z_1, \ldots, z_N) : (g(z_1), \ldots, g(z_N)) \in A\}.$$

Note that $R \in A$ iff $Z \in A'$, and $\Pr[Z \in A'] = \Pr[R \in A] \geq 2^{-a}$. This implies that $H(Z) \geq N \cdot m - a$. Moreover, any q-query Σ^N-decision tree can be simulated by a q-query $(\{0,1\}^m)^N$-decision tree (in the obvious way). Thus, we can apply Lemma 3.5 on Z and obtain the consequence of Lemma 3.3. □

In the next subsection we prove Lemma 3.5.

3.2.1 Proof of Lemma 3.5

In this section we prove Lemma 3.3. The first step is inspired by an argument used in [EIRS01]. Namely, we will identify $i \in [N]$ which are weak, in the following sense.

Definition 3.6 (Weak indices)**.** *We say that $i \in [N]$ is α-weak with respect to $T \subseteq [N]$ if $H(Z_i | Z_{T \setminus \{i\}}) < m - \alpha$.*

Lemma 3.7. *For every $\alpha > 0$, there exists a set $B \subseteq [N]$ of size $b \leq \frac{a}{\alpha}$ such that every $i \in T := [N] \setminus B$ is not α-weak with respect to T.*

Proof. Consider the following iterative process. We start with $B = \emptyset$, and $T = [N]$. In each step, we have some set $B \subseteq [N]$, and let $T = [N] \setminus B$. We will keep the invariant that at step j:

- $|B| = j$.
- $H(Z_T) \geq (N - j) \cdot m - a + j \cdot \alpha$.

[4] It may be that Q assigns probabilities that are not multiples of 2^{-m} to some elements, and then this is true only up to some arbitrarily close precision. As we are allowed statistical distance of $\eta > 0$, we can ignore this technicality.

Note that this indeed holds in the beginning where $j = 0$. We now describe a step in the process: By the chain rule, for every $i \in [T]$

$$H(Z_T) = H(Z_{T\setminus\{i\}}) + H(Z_i|Z_{T\setminus\{i\}}).$$

If there exists $i \in T$ that is α-weak with respect to T then,

$$\begin{aligned} H(Z_{T\setminus\{i\}}) &= H(Z_T) - H(Z_i|Z_{T\setminus\{i\}}) \\ &\geq (N-j)\cdot m - a + j\cdot\alpha - (m-\alpha) \\ &= (N-(j+1))\cdot m - a + (j+1)\cdot\alpha. \end{aligned}$$

Therefore, we keep the invariant if we add i to B and continue. We stop this iterative process, if at some step j, there is no α-weak $i \in T$ with respect to T. Note that if the process continues for j steps, then Z_T is a distribution over $(N-j)\cdot m$ bits, that has entropy $\geq (N-j)\cdot m - a + j\cdot\alpha$. The entropy of Z_T is upper bounded by the length of Z_T, and therefore the process must stop within $\frac{a}{\alpha}$ steps. □

We apply Lemma 3.7 with $\alpha = (\frac{\eta}{2\cdot q})^2$ and obtain a set B of size $b \leq \frac{a}{\alpha} = O(\frac{aq^3}{\eta^3})$. Let $T = [N] \setminus B$.

We assume (for contradiction) that there exists a q-query Σ^N-decision tree P that does not make queries in B, and

$$|\Pr[P(Z) = 1] - \Pr[P(U_{N\cdot m}) = 1]| > \eta.$$

We consider the following hybrid executions of P:

Definition 3.8 (Hybrid executions)**.** *For $0 \leq j \leq q$, we consider the following experiment E^j:*

- *The probability space is over independently chosen Z and $R \leftarrow U_{N\cdot m}$.*
- *We simulate a run of P where in the first j queries, we answer queries using R, and in the remaining queries we answer queries using Z.*
- *Let $H^j = (H^j_1, \ldots, H^j_q)$ be the q answers that P obtained in experiment E^j.*
- *Note that the output of P is completely determined by a sequence $a_1, \ldots, a_q$ of answers, and we will denote this output by $\hat{P}(a_1, \ldots, a_q)$.*
- *Let $p^j = \Pr[\hat{P}(H^j) = 1]$ (namely the probability that P answers one in experiment E^j).*

Note that E^0 is the experiment where P is run on Z, and E^q is the experiment where P is run on R. We have that $|p^0 - p^q| > \eta$, and therefore, there exists $j \in [q]$ such that $|p^{j-1} - p^j| > \frac{\eta}{q}$.

Note that the first $j-1$ answers in both H_j and H_{j-1} are distributed identically, and are composed of independent uniform random variables. By averaging, it follows that there exists values $a_1, \ldots, a_{j-1} \in \{0,1\}^m$ such that for:

- $\hat{H}^{j-1} = (H^{j-1}|H_1^{j-1} = a_1, \ldots, H_{j-1}^{j-1} = a_{j-1}) = (a_1, \ldots, a_{j-1}, H_j^{j-1}, \ldots, H_q^{j-1})$, and
- $\hat{H}^{j} = (H^{j}|H_1^{j} = a_1, \ldots, H_{j-1}^{j} = a_{j-1}) = (a_1, \ldots, a_{j-1}, H_j^{j}, \ldots, H_q^{j})$,

it holds that: $|\Pr[\hat{P}(\hat{H}^{j-1}) = 1] - \Pr[\hat{P}(\hat{H}^{j}) = 1] > \frac{\eta}{q}$.

In both $\hat{H}^{j-1}$ and $\hat{H}^{j}$, the first $j-1$ answers are fixed to $a_1, \ldots, a_{j-1}$. Therefore, both computations follow the same path in the decision tree, and reach the same fixed node after $j-1$ queries. Let Q be the subtree of P that starts at that node (note that Q makes at most q queries). Let i' be the variable that is queried at the root of Q. Let $\tilde{H}^{j-1}, \tilde{H}^{j}$ be the $q-(j-1)$ final answers of $\hat{H}^{j-1}$ and $\hat{H}^{j}$ respectively (namely, these are the answers in experiments E^{j-1} and E^{j} that we have not fixed). We have that:

$$|\Pr[\hat{Q}(\tilde{H}^{j-1}) = 1] - \Pr[\hat{Q}(\tilde{H}^{j}) = 1] > \frac{\eta}{q}.$$

Namely, Q distinguishges between the following two scenarios:

- When its answers are from $\tilde{H}^{j-1}$: Namely, the query on i' is answered by $R_{i'}$ and remaining queries are answered using Z.
- When its answers are from $\tilde{H}^{j}$: Namely, the query on i' is answered by $Z_{i'}$ and remaining queries are answered using Z.

We now return to viewing Q as a function over $(\{0,1\}^m)^N$. At this point, we can forget that this function is implemented by a decision tree, and only recall that it does not depend on variables in B, which in particular means that $i' \in T$. To simplify notation we can reorder the variables and assume w.l.o.g. that $i' = 1$, and $T = [N-b]$. With this choice we have that there exists a function $Q : (\{0,1\}^m)^{N-b} \to \{0,1\}$ such that:

$$|\Pr[Q(U_m, Z_2, \ldots, Z_{N-b}) = 1] - \Pr[Q(Z_1, Z_2, \ldots, Z_{N-b}) = 1]| > \frac{\eta}{q}.$$

This is a contradiction to the fact that i' is not α-weak with respect to T, as is shown in the next lemma.

Lemma 3.9. *The distributions $(U_m, Z_2, \ldots, Z_{N-b})$ and $(Z_1, Z_2, \ldots, Z_{N-b})$ are $\frac{\eta}{q}$-close.*

Proof. We have that 1 is not α-weak in $T = [N-b]$. This means that $H(Z_1|Z_2, \ldots, Z_{N-b}) \geq m - \alpha$. Let

$$G = \left\{(z_2, \ldots, z_{N-b}) : H(Z_1|(Z_2, \ldots, Z_{N-b}) = (z_2, \ldots, z_{N-b})) \geq m - \alpha^{2/3}\right\}.$$

By Markov's inequality, $\Pr[(Z_2, \ldots, Z_{N-b}) \in G] > 1 - \alpha^{1/3}$. By Pinsker's lemma (Lemma 3.1) for every $(z_2, \ldots, z_{N-b}) \in G$, $(Z_1|(Z_2, \ldots, Z_{N-b}) = (z_2, \ldots, z_{N-b}))$ is $\alpha^{\frac{1}{3}}$-close to U_m. Overall, it follows that $(U_m, Z_2, \ldots, Z_{N-b})$ and $(Z_1, Z_2, \ldots, Z_{N-b})$ are ϵ-close for $\epsilon = 2 \cdot \alpha^{1/3} \leq \frac{\eta}{q}$. □

This concludes the proof of Lemma 3.5.

Remark 3.10. *Recent work by Meir and Wigderson [MW17] (with quantitative improvements by Smal and Talebanfard [ST17]) consider a notion that is closely related to weak indices. Loosely speaking, recall that we say that an index i is weak with respect to T, if Z_i can be predicted from $Z_{T\setminus\{i\}}$. Let's say that i is (ϵ, q)-weak with respect to T, if there exists a q-query decision tree that predicts Z_i with probability $\frac{1}{2}+\epsilon$ given access to $Z_{T\setminus\{i\}}$. (We remark that the prediction success is measured in different units with this definition, that only makes sense in case Z_i is a bit). With this notation, the aforementioned works show that if Z is over N bits, and has $H(Z) \geq N - a$ then, the number of indices i that are (ϵ, q)-weak with respect to $[N]$ is at most poly$(q \cdot a/\epsilon)$. (In fact, a stronger claim is true that holds for bounded-width DNF.) This is incomparable to Lemma 3.7. The latter is stronger (as it doesn't require the predictor to be a decision tree) while the former gives "with respect to" $[N]$ rather than $T = [N] \setminus B$ (as is the case in Lemma 3.7). We further remark that (with some care) it is possible to use the result of Meir and Wigderson (in place of Lemma 3.7) in our proof of Lemma 3.5.*

3.3 The fixed set lemma

We restate lemma 1.4 for decision trees over some finite alphabet Σ (rather than for Boolean alphabet).

Lemma 3.11 (Fixed set lemma)**.** *Let N, a, q be integers. Let $R = (R_1, \ldots, R_N)$ be i.i.d. indicator random variables (where each one is over some finite set Σ), let $A \subseteq \Sigma^N$ be an event such that $\Pr[R \in A] \geq 2^{-a}$, and let $X = (R|R \in A)$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $\frac{a \cdot q}{\eta^2} = poly(a, q, \frac{1}{\eta})$, and $v \in \{0,1\}^B$ such that for $R' = (R|R_B = v)$ and $X' = (X|X_B = v) = (R|R_B = v, R \in A)$, and every q-query decision tree P, $P(R')$ and $P(X')$ are η-close.*

Lemma 3.11 follows directly from the following lemma (that can be thought of as the special case in which variables R_i are uniformly distributed over $\Sigma = \{0,1\}^m$).

Lemma 3.12 (Entropy version of fixed set lemma)**.** *Let N, m, a, q be integers. Let $Z = (Z_1, \ldots, Z_N)$ be a distribution over $(\{0,1\}^m)^N$ with $H(Z) \geq N \cdot m - a$, and let $R \leftarrow U_{N \cdot m}$. For every $\eta > 0$, there exists a set $B \subseteq [N]$ of size $\frac{a \cdot q}{\eta^2} = poly(a, q, \frac{1}{\eta})$, and $v \in (\{0,1\}^m)^B$ such that the following holds: Let $Z' = (Z|Z_B = v)$ and let $R' = (R|R_B = v)$. For every q-query $(\{0,1\}^m)^N$-decision tree P, $P(Z')$ and $P(R')$ are η-close.*

We can now derive Lemma 3.11 from Lemma 3.12. The argument is identical to the way we derived Lemma 3.3 from Lemma 3.5. We repeat it below for completeness.

Proof. (of Lemma 3.11) Let $R = (R_1, \ldots, R_N)$ be i.i.d. random variables where each one is distributed according to some distribution Q on Σ. There exists an integer m and $g : \{0,1\}^m \to \Sigma$, such that $g(U_m) = Q$.[5] Consider the probability space in which $Z =$

[5]It may be that Q assigns probabilities that are not multiples of 2^{-m} to some elements, and then this is true only up to some arbitrarily close precision. As we are allowed statistical distance of $\eta > 0$, we can ignore this technicality.

$(Z_1, \ldots, Z_N)$ is chosen from $U_{N \cdot m}$, and for every $i \in [N]$, $R_i = g(Z_i)$. We can imagine that R is chosen this way, and interpret $A \subseteq \Sigma^N$ as

$$A' = \{(z_1, \ldots, z_N) : (g(z_1), \ldots, g(z_N)) \in A\}.$$

Note that $R \in A$ iff $Z \in A'$, and $\Pr[Z \in A'] = \Pr[R \in A] \geq 2^{-a}$. This implies that $H(Z) \geq N \cdot m - a$. Moreover, any q-query Σ^N-decision tree can be simulated by a q-query $(\{0,1\}^m)^N$-decision tree (in the obvious way). Thus, we can apply Lemma 3.12 on Z and obtain the consequence of Lemma 3.11. □

In the next subsection we prove Lemma 3.12.

3.3.1 Proof of Lemma 3.12

The next Lemma shows that if there exists a decision tree that distinguishes, then we can fix the variables along one of its computation paths, and decrease the "entropy deficiency".

Lemma 3.13. *Let N, m, q be integers. Let $Z = (Z_1, \ldots, Z_N)$ be a distribution over $(\{0,1\}^m)^N$, and let $R \leftarrow U_{N \cdot m}$. If there exists a q-query $(\{0,1\}^m)^N$-decision tree P such that $P(Z)$ and $P(R)$ are not η-close, then there exists a set $B \subseteq N$ of size q, and $v \in (\{0,1\}^m)^B$ such that $H(Z|Z_B = v) \geq H(Z) - q \cdot m + \eta^2$.*

Proof. On input $x = (x_1, \ldots, x_N)$, the query y_j made by $P(x_1, \ldots, x_N)$ at step j is completely determined by answers to previous queries: $x_{y_1}, \ldots, x_{y_{j-1}}$. Let f_j be the function that determines the j'th query, namely, for every j, the j'th query of P on $(x_1, \ldots, x_N)$ is given by:

$$y_j = f_j(x_{y_1}, \ldots, x_{y_{j-1}}).$$

The output of P on input $x = (x_1, \ldots, x_N)$ is completely determined by $x_{y_1}, \ldots, x_{y_q}$. Let g be the function that determines this output. Namely,

$$P(x_1, \ldots, x_N) = g(x_{y_1}, \ldots, x_{y_q}).$$

Let $Y = (Y_1, \ldots, Y_q)$ be the queries of P on input Z, and let $Y' = (Y'_1, \ldots, Y'_q)$ be the queries of P on input R.

We have that $P(Z)$ is not η-close to $P(R)$. This implies that Z_Y is not η-close to $R_{Y'}$. Note that by definition $R_{Y'}$ is a uniform string of length $m \cdot q$. Thus, we have that $Z_{Y'}$ is not η-close to $U_{m \cdot q}$. By Pinsker's lemma (Lemma 3.1) this implies that: $H(Z_Y) < q \cdot m - \eta^2$

By the chain rule for entropy, and the fact that Z_Y is a function of Z we have that:

$$H(Z) = H(Z, Z_Y) = H(Z_Y) + H(Z|Z_Y).$$

We use that to show:

$$H(Z|Z_Y) = H(Z) - H(Z_Y) \geq H(Z) - (q \cdot m - \eta^2).$$

By averaging, there exists $z' \in (\{0,1\}^m)^q$ such that:

$$H(Z|Z_Y = z') \geq H(Z|Z_Y) \geq H(Z) - q \cdot m + \eta^2.$$

Note that $(Y|Z_Y = z')$ is a constant rather than a random variable. Specifically, there is a fixed $y = (y_1, \ldots, y_q)$ such that $(Y|Z_Y = z') = y$. This is because fixing the answers to q queries, also fixes the variables queried by a decision tree. Thus, setting $B = \{y_1, \ldots, y_q\}$ (and we can assume w.l.o.g. that the q queries are distinct) we have that:

$$H(Z|Z_B = z') \geq H(Z) - q \cdot m + \eta^2,$$

as required. □

We are now ready to prove Lemma 3.12.

Proof. (of Lemma 3.12) We apply Lemma 3.12 iteratively, until there is no q-query $(\{0,1\}^m)^N$-decision tree P for which $P(Z)$ and $P(R)$ are not η-close. At step i, we have fixed $i \cdot q$ variables, let B^i (of size $i \cdot q$) be the set of variables we have fixed, and $v^i \in \Sigma^{B^i}$ be the values to which we fixed them. At this point we hold a distribution $Z^i = (Z|B^i = v^i)$ which has $N - i \cdot q$ variables that were not yet fixed. Therefore, $H(Z^i) \leq (N - i \cdot q) \cdot m$. On the other hand, as we started with $H(Z^0) \geq N \cdot m - a$, by Lemma 3.13 we have that at step i,

$$H(Z^i) \geq H(Z^0) - i \cdot q \cdot m + i \cdot \eta^2 \geq N \cdot m - a - i \cdot q \cdot m + i \cdot \eta^2 = (N - i \cdot q) \cdot m - a + i \cdot \eta^2.$$

It follows that this process must stop after at most a/η^2 steps, and so the final B^i is of size no more than $\frac{a \cdot q}{\eta^2}$ as required. When the process stops we are guaranteed that there does not exist a q-query $(\{0,1\}^m)^N$-decision tree P for which $P(Z|Z_{B^i} = v^i)$ and $P(R|R_{B^i} = v^i)$ are not η-close. □

4 The zoom lemma

In this section we prove the zoom lemma. Given a black-box hardness amplification $(\mathsf{Con}, \mathsf{Red})$ the lemma allows to "zoom in" on a particular reduction $\mathsf{Red}(x, \alpha)$ that can be used to distinguish uniform noise from noise that is ϵ-close to uniform.

We first need a couple of definitions and a lemma. First we define "noise."

Definition 4.1. *We use Noise_p^N to denote the distribution of N i.i.d. random variables over $\{0,1\}$, where each one has probability p to be one. For some finite set V, we use Noise_p^V to denote such a distribution over $N = |V|$ variables, indexed by elements in V (rather than by $[V]$).*

We have two kinds of random access procedures: The first is Σ^V-decision trees, and the second is oracle circuits $C^{(\cdot)}$. It will be natural to view an input to a tree as an oracle to a circuit, and vice-versa.

Definition 4.2 (Identification of functions and truth tables)**.** *For a finite alphabet Σ and a finite set V, we can view elements $D \in \Sigma^V$ in two ways:*

- *As* functions *$D : V \to \Sigma$.*
- *As* strings*, namely we order V in some way, and view D as a string of length $|V|$ over alphabet Σ defined by $D = (D_y)_{y \in V}$.*

We will allow ourselves to hold both views (without introducing different notations for the two views).

We shall also need to talk about hard functions.

Definition 4.3 (Hard functions)**.** *Let $H \subseteq \{0,1\}^k$ be a set. We say that a function $f : \{0,1\}^k \to \{0,1\}$ is ρ-hard for for circuits of size s, if for every circuit C of size s, $\Pr_{x \leftarrow \{0,1\}^k}[C(x) = f(x)] \leq p$ for $p = \frac{1}{2} + \rho$.*

The following lemma follows by a standard counting argument.

Lemma 4.4 (Existence of hard functions)**.** *There exists a constant $\lambda > 0$ such that for every sufficiently large k, there exists a function $f : \{0,1\}^k \to \{0,1\}$ that is $\frac{1}{s}$-hard for circuits of size $s = 2^{\lambda \cdot k}$.*

We can now state the zoom lemma. Following [SV10] we want to use Red to distinguish noise $1/2 - \epsilon$ from noise $1/2$. The next lemma gives a fixed advice α and a fixed input x for which $\mathsf{Red}(x, \alpha)$ distinguishes.

Lemma 4.5. *There exist constants $\delta_0, \nu > 0$ and $d > 1$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to (\frac{1}{2} - \epsilon)$ black-box hardness amplification with input lengths k and n, and list size 2^a. Suppose for every x and α the reduction Red is a circuit of size r that makes at most q oracle queries. Assume $n, a, q, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$, and $\delta \leq \delta_0$. Let $\eta := \delta + 2^{-\nu \cdot k}$. Let $f : \{0,1\}^k \to \{0,1\}$ be a function that is $1/200$-hard for circuits of size $s = r^d$.*

There exists a set $B \subseteq \{0,1\}^n$ of size $(aq/\eta)^{O(1)}$, a string $v \in \{0,1\}^B$, $x \in \{0,1\}^k$ and $\alpha \in \{0,1\}^a$ such that

- $\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}^{\mathsf{Con}_{f \oplus D}}(x, \alpha) = f(x) | D(B) = v] \geq 1 - 2\sqrt{\delta} - 2^{-\nu \cdot k}$,
- $\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}}}[\mathsf{Red}^{\mathsf{Con}_{f \oplus D}}(x, \alpha) = f(x) | D(B) = v] \leq 0.51$.

4.1 Proof

In this proof we fix $V = \{0,1\}^n$. We first show that there exists a string $\alpha \in \{0,1\}^a$ for which the reduction succeeds with probability about 2^{-a} on the oracle $\mathsf{Noise}^V_{\frac{1}{2}-2\epsilon} \oplus \mathsf{Con}_f$.

Lemma 4.6. *There exist $\alpha \in \{0,1\}^a$ and $A \subseteq \{0,1\}^V$ such that:*

$$\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[D \in A] \geq 2^{-(a+1)}$$

and for every $D : \{0,1\}^n \to \{0,1\}$ such that $D \in A$,

$$\Pr_{X \leftarrow U_k}[\mathsf{Red}^{D \oplus \mathsf{Con}_f}(X, \alpha) = f(X)] \geq 1 - \delta.$$

Proof. The proof amounts to a Chernoff bound and an averaging argument. Specifically, by a Chernoff bound, with probability $1 - 2^{-\Omega(\epsilon^2 \cdot 2^n)}$ over $\mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}$ the Hamming weight of the obtained 2^n-bit long string D is less than $(\frac{1}{2} - \epsilon) \cdot 2^n$. This means that $\mathsf{Con}_f \oplus D$ $(\frac{1}{2} + \epsilon)$-agrees with Con_f. Therefore, for every $D \in \{0,1\}^V$ with that property, there exists $\alpha \in \{0,1\}^a$ such that

$$\Pr_{X \leftarrow U_k}[\mathsf{Red}^{D \oplus \mathsf{Con}_f}(X, \alpha) = f(X)] \geq 1 - \delta.$$

Therefore, by an averaging argument, there must exist some $\alpha \in \{0,1\}^a$, and $A \subseteq \{0,1\}^V$ that satisfy the required properties. □

We now apply the fixed set lemma (Lemma 3.11) with the following choices:

- Let $\Sigma = \{0,1\}$ and we consider the distribution $\mathsf{Con}_f \oplus \mathsf{Noise}^1_{\frac{1}{2}-2\epsilon}$ on Σ.
- Let $N = 2^n$, and we identify $V = \{0,1\}^n$ with $[N]$.
- We use the event $A \subseteq \{0,1\}^V$ which we interpret as $A \subseteq \{0,1\}^N$.

Recall that the parameter η is chosen in the statement of the lemma. We obtain a set $B \subseteq V$ of size $\mathrm{poly}(a, q, 1/\eta)$, and $v \in \{0,1\}^B$ such that for every $x \in \{0,1\}^k$,

$$\left| \Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}(x, \alpha)^{\mathsf{Con}_{f \oplus D}} = 1 | D(B) = v] - \Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}(x, \alpha)^{\mathsf{Con}_{f \oplus D}} = 1 | D(B) = v, D \in A] \right| \leq \eta. \quad (2)$$

This follows as for every $x \in \{0,1\}^k$, $\mathsf{Red}^{\mathsf{Con}_{f \oplus D}}(x, \alpha)$ can be viewed as q-query $\{0,1\}^V$-decision tree with input $D \in \{0,1\}^V$.

It follows from Lemma 4.6 that:

$$\Pr_{X \leftarrow U_k, D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}(x, \alpha)^{\mathsf{Con}_{f \oplus D}} = f(X) | D(B) = v, D \in A] \geq 1 - \delta.$$

Specifically, the lemma guaranteed this for every $D \in A$. So in particular it also holds for every $D \in A$ that satisfies the further condition $D(B) = v$.

By a Markov argument, for a $(1-\sqrt{\delta})$-fraction of $x \in \{0,1\}^k$,

$$\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}^{\mathsf{Con}_{f \oplus D}}(x,\alpha)(D) = f(x) | D(B) = v, D \in A] \geq 1 - \sqrt{\delta}.$$

Combining this equation with Equation (2) yields

$$\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}^{\mathsf{Con}_{f \oplus D}}(x,\alpha)(D) = f(x) | D(B) = v] \geq 1 - 2\sqrt{\delta} - 2^{-\nu \cdot k}. \tag{3}$$

Recall that f is $\frac{1}{200}$-hard for circuits of size s. Using the hardness of f, the bound on b, and the restriction on the size of Red, it follows that:

Lemma 4.7. $\Pr_{X \leftarrow \{0,1\}^k, D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}}}[\mathsf{Red}^{D \oplus \mathsf{Con}_f}(X,\alpha) = f(X) | D(B) = v] \leq \frac{1}{2} + \frac{1}{200}$.

Proof. Consider the randomized circuit R that on input x, simulates $\mathsf{Red}^{(\cdot)}(x,\alpha)$ and whenever Red makes an oracle query y, then if $y \in B$, then R answers the query by v_y, and otherwise it answers by a fresh random coin. We claim that for every $x \in \{0,1\}^k$,

$$\Pr[R(x) = f(x)] = \Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}}}[\mathsf{Red}^{D \oplus \mathsf{Con}_f}(x,\alpha) = f(X) | D(B) = v].$$

This is because the distribution $\mathsf{Con}_f \oplus \mathsf{Noise}^V_{\frac{1}{2}}$ is the same as $\mathsf{Noise}^V_{\frac{1}{2}}$. Note that R is a (distribution over) circuits of size $r + O(n \cdot b) \leq r + O(n \cdot r^{d-2})$. By choosing the constant d to be sufficiently large, R is of size at most $s = r^d$. By averaging, we get that there exists a fixing for R's random coins so that the obtained deterministic circuit R' satisfies:

$$\Pr_{X \leftarrow \{0,1\}^k, D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}}}[\mathsf{Red}^{D \oplus \mathsf{Con}_f}(X,\alpha) = f(X) | D(B) = v] = \Pr_{X \leftarrow \{0,1\}^k}(R'(X) = f(X)] \leq \frac{1}{2} + \frac{1}{200}.$$

Where the last inequality follows by the fact that f is $\frac{1}{200}$-hard for size s circuits. □

By a Markov argument we obtain that for a $\frac{1}{200}$-fraction of $x \in \{0,1\}^k$,

$$\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}}}[\mathsf{Red}^{D \oplus \mathsf{Con}_f}(x,\alpha) = f(x) | D(B) = v] \leq 0.51. \tag{4}$$

We can pick the constant δ_0 so that for $\delta \leq \delta_0$, $\sqrt{\delta} < \frac{1}{200}$, and by a union bound, there must exists $x' \in \{0,1\}^k$ such that for this x', both Equation (3) and (4) hold. This concludes the proof.

4.2 Making Con_f easy to compute

The zoom lemma proved in the previous section is sufficient to prove our lower bound on the number of queries. For our results on the necessity of majority we need to simulate the circuits $\mathsf{Red}^{\mathsf{Con}_{f\oplus D}}(x,\alpha)$. The straightforward simulation involves computing Con_f. Although this can be done in some cases[6] in general it is not clear how to do that. We now use an argument of Gutfreund and Rothblum [GR08] that removes the need to compute Con_f altogether.

Lemma 4.8. *Let the hypothesis of the zoom lemma be satisfied. Further assume that $\mathsf{Red}^{\cdot}(x,\alpha)$ is an oracle circuit of depth H. Then there exists a circuit $C:\{0,1\}^q\to\{0,1\}$ of size poly(r) and depth $O(H)$ such that*

$$\left|\Pr\left[C(\mathsf{Noise}^q_{1/2-2\epsilon})=1\right]-\Pr\left[C(\mathsf{Noise}^q_{1/2})=1\right]\right|\geq\Omega(1/H).$$

Proof. We apply the zoom lemma to obtain B, v, x, and α. Here, we don't expect to gain in the case that δ is very small, and will therefore assume that δ is a sufficiently small constant, so that it doesn't affect the size of B. If $y\in B\subseteq\{0,1\}^n$ we write $v(y)$ for the corresponding bit of v. Because $\mathsf{Red}^{\cdot}(x,\alpha)$ has depth H we can think of its gates as being arranged in H layers, where Layer 1 is the input and Layer H is the output.

Let $U_1,U_2,\ldots,U_H$ be sampled from $\mathsf{Noise}^N_{1/2}$ and $B_1,B_2,\ldots,B_H$ be sampled from $\mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}$, where $V=\{0,1\}^n$. Consider the function g which maps H input oracles $o_1,o_2,\ldots,o_H:\{0,1\}^n\to\{0,1\}$ to a bit as follows. g simulates $\mathsf{Red}^{(\cdot)}(x,\alpha)$. Whenever Red makes an oracle query $y\in\{0,1\}^n$, g answers it as follows. If $y\in B$ then the answer is $(\mathsf{Con}_f\oplus v)(y)$. If $y\notin B$ and the query is made at level i then g answers it with o_i.

Note that by definition

$$\Pr[g(\mathsf{Con}_f\oplus U_1,\mathsf{Con}_f\oplus U_2,\ldots,\mathsf{Con}_f\oplus U_H)=f(x)]$$
$$=\Pr_{D\leftarrow\mathsf{Noise}^V_{\frac{1}{2}}}[\mathsf{Red}^{\mathsf{Con}_{f\oplus D}}(x,\alpha)=f(x)|D(B)=v],$$

and

$$\Pr[g(\mathsf{Con}_f\oplus B_1,\mathsf{Con}_f\oplus B_2,\ldots,\mathsf{Con}_f\oplus B_H)=f(x)]$$
$$=\Pr_{D\leftarrow\mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}^{\mathsf{Con}_{f\oplus D}}(x,\alpha)=f(x)|D(B)=v].$$

[6]For XOR lemma, this can be accomplished using parity gates as follows. Pick f that in addition to being hard for size s only depends on $O(\log s)$ bits and therefore can be computed with slightly larger circuit size poly(s). Using such an f one can verify that Con_f can also be computed in size poly(s) using parity gates. One can also handle Con that is obtained by concatenating the Reed-Muller and the Hadamard code reasoning similarly and following [Vio06], Section 6.2.2.

By the conclusion of the zoom lemma, using that δ is sufficiently small

$$\Big|\Pr[g(\mathsf{Con}_f \oplus B_1, \mathsf{Con}_f \oplus B_2, \ldots, \mathsf{Con}_f \oplus B_H) = f(x)] \\ - \Pr[g(\mathsf{Con}_f \oplus U_1, \mathsf{Con}_f \oplus U_2, \ldots, \mathsf{Con}_f \oplus U_H) = f(x)]\Big| \geq \Omega(1).$$

We can now consider $H+1$ hybrid distributions where the first $i-1$ oracles are B_j and the rest are U_j. By the triangle inequality, there exists an i such that the i and $i+1$ hybrid exhibit a gap of $\Omega(1/H)$:

$$\Big|\Pr[g(\mathsf{Con}_f \oplus B_1, \ldots, \mathsf{Con}_f \oplus B_{i-1}, \mathsf{Con}_f \oplus B_i, \mathsf{Con}_f \oplus U_{i+1}, \ldots, \mathsf{Con}_f \oplus U_H) = f(x)] \\ -\Pr[g(\mathsf{Con}_f \oplus B_1, \ldots, \mathsf{Con}_f \oplus B_{i-1}, \mathsf{Con}_f \oplus U_i, \mathsf{Con}_f \oplus U_{i+1}, \ldots, \mathsf{Con}_f \oplus U_H) = f(x)]\Big| \geq \Omega(1/H).$$

Note that $\mathsf{Con}_f \oplus U_j = U_j$ hence we get

$$\Big|\Pr[g(\mathsf{Con}_f \oplus B_1, \ldots, \mathsf{Con}_f \oplus B_{i-1}, \mathsf{Con}_f \oplus B_i, U_{i+1}, \ldots, U_H) = f(x)] \\ - \Pr[g(\mathsf{Con}_f \oplus B_1, \ldots, \mathsf{Con}_f \oplus B_{i-1}, \mathsf{Con}_f \oplus U_i, U_{i+1}, \ldots, U_H) = f(x)]\Big| \geq \Omega(1/H).$$

By averaging we can fix the first $i-1$ oracles to maintain the gap. Thus there exist functions $b_j : \{0,1\}^n \to \{0,1\}$ such that

$$\Big|\underbrace{\Pr[g(\mathsf{Con}_f \oplus b_1, \ldots, \mathsf{Con}_f \oplus b_{i-1}, \mathsf{Con}_f \oplus B_i, U_{i+1}, \ldots, U_H) = f(x)]}_{A} \\ - \underbrace{\Pr[g(\mathsf{Con}_f \oplus b_1, \ldots, \mathsf{Con}_f \oplus b_{i-1}, \mathsf{Con}_f \oplus U_i, U_{i+1}, \ldots, U_H) = f(x)]}_{B}\Big| \geq \Omega(1/H).$$

Consider the computations of Red corresponding to the above evaluations of g. Note that the answers to the queries in the first $i-1$ layers are fixed. Hence they can be hardwired in the circuit. This makes the queries in Layer i fixed as well. So one can hardwire the values of Con_f at those queries as well. The next queries are not fixed, but they are answered by a random bit, except at points in B.

The circuit C can now be given. First, we claim that there exists a distribution on circuits C' such that

$$\Pr_{C', \mathsf{Noise}^q_{1/2-2\epsilon}}\Big[C'(\mathsf{Noise}^q_{1/2-2\epsilon}) = f(x)\Big] = A$$

$$\Pr_{C', \mathsf{Noise}^q_{1/2}}\Big[C'(\mathsf{Noise}^q_{1/2}) = f(x)\Big] = B.$$

The circuit C' answers queries as follows. First, it checks if a query lands in B. If so it answers it with $\mathsf{Con}_f \oplus v$. This computation can be implemented by a circuit of constant depth and size polynomial in $|B|$. Because $|B| \leq \mathrm{poly}(r)$, this check can be implemented in the required resources. Otherwise, recall from above that the answers to the queries in the first $i-1$ levels are hardwired. The answers to the queries in level i are answered by picking the "noise" part from the input and again hardwiring the values of Con_f. The queries at levels larger than i are answered by flipping a coin.

By an averaging argument we can fix the latter coins and maintain the gap. Then we can define C to be C' with this fixing, possibly complementing the output. $\square$

5 Limitations on black-box hardness amplification and PRG constructions

In this section we use the zoom lemma to prove lower bounds on hardness amplification proofs. In Section 5.1 we prove Theorem 1.7 (showing a lower bound for the number of queries). In Section 5.2 we prove Theorem 1.8 (showing that hardness amplification proofs imply majority). In Section 5.3 we consider a variant of hardness amplification task (that was studied by Watson [Wat15] under the name "errorless amplification" and Artemenko and Shaltiel [AS14] under the name "basic hardness amplification"). This form of hardness amplification corresponds to a coding theoretic setting where Red is a local list decoding algorithm that recovers from *erasures* rather than from *errors* (as in standard hardness amplification). We prove tight lower bounds on the number of queries required in this setting. Finally, in Section 5.4 we state and prove our limitations on black-box PRG constructions.

5.1 Lower bound on the number of queries

In this section we prove Theorem 1.7. We need a lemma from [SV10] giving a lower bound on the number q of bits sufficient to distinguish $\mathsf{Noise}^q_{1/2}$ from $\mathsf{Noise}^q_{1/2-\epsilon}$. See [SV10] for further discussion and related results.

Lemma 5.1. *Let $g : \{0,1\}^q \to \{0,1\}$ be a function such that*

1. $\Pr_{\mathsf{Noise}^q_{1/2-\epsilon}}\left[g(\mathsf{Noise}^q_{1/2-\epsilon}) = 1\right] \leq p \leq 0.4$, *and*
2. $\Pr_{\mathsf{Noise}^q_{1/2}}\left[g(\mathsf{Noise}^q_{1/2}) = 1\right] \geq 0.49$.

Then $q \geq \Omega(\log(1/p)/\epsilon^2)$.

Proof of Theorem 1.7. We apply the Zoom Lemma 4.5, where the function f is obtained from Lemma 4.4. We claim there exists a function g such that, for both $z = 0$ and $z = \epsilon$: the distribution of $t(D)$ for D sampled from $\mathsf{Noise}^q_{1/2-z}$ is the same as the distribution of $(\mathsf{Red}^{\mathsf{Con}_{f\oplus D}}(x,\alpha)|D(B) = v)$ for D sampled from $\mathsf{Noise}^V_{1/2-z}$. The function g simulates Red, answering queries in B with the corresponding value of v, and the others using the input. Note that x, α and f are fixed.

Hence either g or $1-g$ satisfies the hypothesis of Lemma 5.1, and the proof is concluded. □

5.2 Hardness amplification proofs require majority

We need another lemma from [SV10]. The lemma says that any circuit that distinguishes $\mathsf{Noise}^q_{1/2}$ from $\mathsf{Noise}^q_{1/2-\epsilon}$ can be used to compute majority on $1/\epsilon$ bits.

Lemma 5.2 (Distinguishing implies majority)**.** *Let $g : \{0,1\}^q \to \{0,1\}$ be a function such that*

$$|\Pr\left[g(\mathsf{Noise}^q_{1/2-\epsilon}) = 1\right] - \Pr\left[g(\mathsf{Noise}^q_{1/2}) = 1\right]| \geq \frac{1}{\log(1/\epsilon)}.$$

Then there exists a circuit of size $(q/\epsilon)^{O(1)}$ and depth $O(1)$ with oracle access to g that computes the Majority function on $1/\epsilon$ bits.

Proof of Theorem 1.8. We apply the Zoom Lemma 4.5 and Lemma 4.8, where the function f is obtained from Lemma 4.4, and obtain a circuit C of depth $O(d)$ that distinguishes $\mathsf{Noise}^q_{\frac{1}{2}-2\epsilon}$ from $\mathsf{Noise}^q_{\frac{1}{2}}$ with advantage $\Omega(1/d)$. The theorem now follows from Lemma 5.2, using the requirement that $d = O(\log(1/\epsilon))$. □

5.3 Lower bounds on the erasure version of hardness amplification

We now consider a variant of hardness amplification. While Definition 1.6 corresponds to to locally list decodable codes against *errors*, the variant we consider here corresponds to locally list decodable codes against *erasures*. We therefore call this variant "hardness amplification with erasures".

We start by giving a definition of black-box hardness amplification with erasures. We need the following notion of "agreement with erasures", which extends the standard notion of agreement in Definition 1.6.

Definition 5.3 (erasure-agreement of functions)**.** *Let $g_1 : W \to O \cup \{\perp\}$, $g_2 : W \to O$ be functions, where W is a finite domain and O does not include the special symbol '$\perp$'. We say that g_1 p-erasure-agrees with g_2 if g_1 p-agrees with g_2, and for every $w \in W$, $g_1(w) \neq \perp \Rightarrow g_1(w) = g_2(w)$.*

Note that unlike agreement of functions, this relation isn't symmetric.

The following definition is analogous to Definition 1.6, and this form of hardness amplification was termed "basic hardness amplification" in [AS14]. Loosely speaking, it requires less than Definition 1.6 as the reduction is required to perform only if D erasure-agrees with Con_f.

Definition 5.4 (black-box hardness amplification with erasures [AS14][7])**.** *A $\delta \to \epsilon$ black-box erasure hardness amplification with input lengths k and n, list size 2^a, and output length o is a pair $(\mathsf{Con}, \mathsf{Red})$ such that:*

- *A* construction Con *is a map from functions $f : \{0,1\}^k \to \{0,1\}$ to functions $\mathsf{Con}_f : \{0,1\}^n \to \{0,1\}^o$.*
- *A* reduction Red *is an oracle circuit $\mathsf{Red}^{(\cdot)}(x, \alpha)$ that accepts two inputs: $x \in \{0,1\}^k$ and $\alpha \in \{0,1\}^a$ (we call α a "nonuniform advice string"). Red also receives oracle access to a function $D : \{0,1\}^n \to \{0,1\}^o$.*

[7] We use a different terminology than Artemenko and Shaltiel [AS14]. In [AS14], this task is called "function generic basic hardness amplification".

We require that for all functions $f : \{0,1\}^k \to \{0,1\}$ and $D : \{0,1\}^n \to \{0,1\}^o$ such that D ϵ-erasure-agrees with Con_f, there exists $\alpha \in \{0,1\}^a$ such that $\mathsf{Red}^D(x,\alpha)$ $(1-\delta)$-agrees with f. Note that for if $\delta < 2^{-k}$ then it follows that $\mathsf{Red}^D(x,\alpha)$ 1-agrees with f.

A few comments are in order:

- Given a function D that ϵ-erasure agrees with a boolean function g, we can consider the (probabilistic) function D' which answers like D, except that whenever D answers '$\perp$', D' tosses a coin. It follows that D $(\frac{1}{2} + \epsilon/2)$-agrees with g.

- Consequently, any $\delta \to \frac{1}{2} - \epsilon$ black-box amplification translates into a $\delta \to \epsilon/2$ black-box erasure hardness amplification. Meaning that, lower bounds on black-box erasure hardness amplification immediately translate into lower bounds on (standard) hardness amplification.

- Every black-box proof for a concatenation (or direct product) lemma, namely the case that $\mathsf{Con}_f(x_1,\ldots,x_t) = (f(x_1),\ldots,f(x_t))$, (and note that Con_f isn't boolean) can be transformed into a black-box erasure hardness amplification, and so, lower bounds on black-box erasure hardness amplification apply to concatenation lemmas (or direct product lemmas).

- Another variant termed "errorless amplification", was considered by Bogdanov and Safra [BS07] and Watson [Wat15]. In our terminology, this corresponds to the more stringent requirement in Definition 5.4 that $\mathsf{Red}^D(x,\alpha)$ $1-\delta$-erasure-agrees with f. (That is, the reduction is allowed to answer $\perp$ with probability δ, but is not allowed to err). As this requires more from the reduction, any errorless amplification is in particular erasure hardness amplification, and lower bounds for black-box erasure hardness amplification translate to the case of errorless amplification.

- Unlike (standard) hardness amplification – erasure hardness amplification, errorless amplification, and direct product lemmas "*do not* imply majority". Namely, there are reductions in the literature which for small ϵ, are constant depth circuits without majority gates [IJKW10, Wat15]. This was observed already in [Vio06] for the direct product lemma, and can be verified by inspection for example of [Wat15] for error-less hardness amplification, see [AS14] for discussion. Moreover, these reductions make *less* queries than is possible for (standard) hardness amplification, and achieve $q = O(\frac{\log(1/\delta)}{\epsilon})$ whereas by Theorem 1.7 $q = \Omega(\frac{\log(1/\delta)}{\epsilon^2})$ are required for black-box hardness amplification.

In light of the last comment, in the case of black-box erasure hardness amplification, we can only hope to prove lower bounds on the number of queries, and are shooting for lower bounds of the form $q = \Omega(\frac{\log(1/\delta)}{\epsilon})$.

Watson [Wat15] proved such a lower bound for *nonadaptive* reductions, by observing that the proof of Shaltiel and Viola [SV10] extends to the setup of erasures, and gives this bound.

Artemenko and Shaltiel proved a lower bound of $q = \Omega(\frac{1}{\epsilon})$ which holds even for *adaptive* reductions.

In this paper we prove a tight lower bound of $q = \Omega(\frac{\log(1/\delta)}{\epsilon})$ on black-box erasure hardness amplification that holds for adaptive reductions.

Theorem 5.5 (Lower bound on the number of queries for erasures)**.** *There exist constants $\delta_0, \nu > 0$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to \epsilon$ black-box erasure hardness amplification with input lengths k and n, list size 2^a and output length o. Assume that:*

- $\mathsf{Red}^{(\cdot)}$ *is a size r oracle circuit, that makes at most q (possibly adaptive) queries.*
- $n, a, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$ *and* $2^{-2k} \leq \delta \leq \delta_0$.

Then $q = \Omega(\frac{\log(1/\delta)}{\epsilon})$.

The proof of Theorem 5.5 follows by modifying the proof of Theorem 1.7 as follows: We prove a slightly modified version of the zoom lemma (Lemma 4.5). Specifically, for $a, b \in \{0,1\}$ we define: $a \circledast b$ to be a, if $b = 0$, and $\perp$ otherwise. That is, the bit b "decides" whether the bit a is kept or erased. As in the case of "$\oplus$" we extend the definition of "$\circledast$" to strings and functions.

In the modified version of the zoom lemma, we replace the oracles $Con_f \oplus \mathsf{Noise}_s$ for $s = \frac{1}{2}$ or $s = \frac{1}{2} - 2\epsilon$ in the lemma, by $\mathsf{Con}_f \circledast \mathsf{Noise}_s$ for $s = 1$ or $s = 1 - 2\epsilon$ (respectively). Loosely speaking, this is analogous in the sense that $\mathsf{Con}_f \oplus \mathsf{Noise}_{\frac{1}{2}}$ and $\mathsf{Con}_f \circledast \mathsf{Noise}_1$ both "wipe out" the information in Con_f making such an oracle useless for a reduction, while $\mathsf{Con}_f \oplus \mathsf{Noise}_{\frac{1}{2}-2\epsilon}$ and $\mathsf{Con}_f \circledast \mathsf{Noise}_{1-2\epsilon}$ are oracles on which the respective reductions should succeed. We state this version of the modified zoom-lemma below:

Lemma 5.6. *There exist constants $\delta_0, \nu > 0$ and $d > 1$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to (\frac{1}{2} - \epsilon)$ black-box erasure hardness amplification with input lengths k and n, list size 2^a and output length o. Suppose for every x and α the reduction Red is a circuit of size r that makes at most q oracle queries. Assume $n, a, q, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$, and $\delta \leq \delta_0$. Let $\eta := \delta + 2^{-\nu \cdot k}$. Let $f : \{0,1\}^k \to \{0,1\}$ be a function that is $1/200$-hard for circuits of size $s = r^d$.*

There exists a set $B \subseteq \{0,1\}^n$ of size $(aq/\eta)^{O(1)}$, a string $v \in \{0,1\}^B$, $x \in \{0,1\}^k$ and $\alpha \in \{0,1\}^a$ such that

- $\Pr_{D \leftarrow \mathsf{Noise}_{1-2\epsilon}^V}[\mathsf{Red}^{\mathsf{Con}_f \circledast D}(x, \alpha) = f(x) | D(B) = v] \geq 1 - 2\sqrt{\delta} - 2^{-\nu \cdot k}$,
- $\Pr_{D \leftarrow \mathsf{Noise}_1^V}[\mathsf{Red}^{\mathsf{Con}_f \circledast D}(x, \alpha) = f(x) | D(B) = v] \leq 0.51$.

The proof of Lemma 5.6 follows using exactly the same argument as in Lemma 4.5 (with the modification explained above). Theorem 5.5 follows in the same way as Theorem 1.7 follows from Lemma 4.5, using the fact that if a function $g : \{0,1\}^q \to \{0,1\}$ distinguishes Noise_1^q from $\mathsf{Noise}_{1-\epsilon}^q$ with advantage $1 - \delta$, then $q = \Omega(\frac{\log(1/\delta)}{\epsilon})$.

5.4 Limitations on black-box constructions of PRGs from hard functions

Our lower bounds on black-box hardness amplification can be extended to the related setup of "black-box constructions of pseudorandom generators from hard functions". This setup is often referred to as "hardness versus randomness" or "pseudorandom generators in the Nisan-Wigderson setting". This is an extensive and highly successful line of research that is often the motivation for hardness amplification [NW94, BFNW93, Imp95, IW97, IW01, STV01, SU05, Uma03].

All the constructions above use the "hybrid argument" [GM84, BM84, Yao82], and consequently, in order to construct PRGs with error ϵ' that outputs m bits, they need to amplify hardness and achieve "amplified hardness" of $\epsilon < \epsilon'/n$. (Some of the constructions [SU05, Uma03] do not explicitly use hardness amplification. However, these constructions "amplify hardness by themselves", and yield hardness amplification as explained in [SU06]).

In this section we will consider the problem of constructing PRGs from hard functions (without explicitly requiring a hardness amplification step). We do not know whether hardness amplification to $\epsilon < 1/n$ is necessary in this setup. (The reader is referred to [FSUV13] for a study on this problem, which shows that in some weak sense, the loss of the hybrid argument can sometimes be avoided). Our techniques can be used to prove limitations on black-box PRG constructions that yield PRGs with low error.

Formal definition of black-box PRG constructions. Below, we give a definition of black-box PRG constructions. The main difference from hardness amplification is that in this setup:

- The task of Con_f is to construct a "PRG" stretching ℓ bits to n bits.
- The reduction Red expects to receive oracle access to a function $D : \{0,1\}^n \to \{0,1\}$ that "breaks" the PRG Con_f.

A precise definition follows.

Definition 5.7. *A function $D : \{0,1\}^n \to \{0,1\}$ ϵ-breaks a function $G : \{0,1\}^\ell \to \{0,1\}^n$ if:*

$$|\Pr_{S \leftarrow U_\ell}[D(G(S)) = 1] - \Pr_{Y \leftarrow U_n}[D(Y) = 1]| > \epsilon$$

Definition 5.8 (black-box PRG constructions)**.** *A $\delta \to \epsilon$ black-box PRG coonstruction with input lengths k, seed length ℓ, output length n, and list size 2^a, is a pair $(\mathsf{Con}, \mathsf{Red})$ such that:*

- *A* construction Con *is a map from functions $f : \{0,1\}^k \to \{0,1\}$ to functions $\mathsf{Con}_f : \{0,1\}^\ell \to \{0,1\}^n$.*
- *A* reduction Red *is an oracle circuit $\mathsf{Red}^{(\cdot)}(x, \alpha)$ that accepts two inputs: $x \in \{0,1\}^k$ and $\alpha \in \{0,1\}^a$ (we call α a "nonuniform advice string"). Red also receives oracle access to a function $D : \{0,1\}^n \to \{0,1\}$.*

We require that for all functions $f : \{0,1\}^k \to \{0,1\}$ and $D : \{0,1\}^n \to \{0,1\}$ such that D ϵ-breaks Con_f, there exists $\alpha \in \{0,1\}^a$ such that $\mathsf{Red}^D(x,\alpha)$ $(1-\delta)$-agrees with f. Note that for if $\delta < 2^{-k}$ then it follows that $\mathsf{Red}^D(x,\alpha)$ 1-agrees with f.

We remark that black-box PRG constructions are closely related to seeded extractors [Tre01] (see for example discussion in [Sha02]) and local list-decoding algorithms to "extractor codes" [TZ04]. Loosely speaking, this relation is "of the same kind" of black-box hardness amplification to error correcting codes.

Our results on black-box PRG constructions. We can prove the following extension of Theorems 1.7 and Theorem 1.8.

Theorem 5.9 (Lower bound on the number of queries for PRGs)**.** *There exist constants $\delta_0, \nu > 0$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to \epsilon$ black-box PRG construction with input lengths k, seed length ℓ, output length n, and list size 2^a. Assume that:*

- *$\mathsf{Red}^{(\cdot)}$ is a size r oracle circuit, that makes at most q (possibly adaptive) queries.*
- *$n, a, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$, $2^{-2k} \leq \delta \leq \delta_0$ and $\ell < n$.*

Then $q = \Omega(\frac{\log(1/\delta)}{\epsilon})$.

Theorem 5.10 (Hardness amplification implies majority)**.** *There exist constants $\delta_0, \nu > 0$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to \epsilon$ black-box PRG construction with input lengths k, seed length ℓ, output length n, and list size 2^a. Assume that:*

- *$\mathsf{Red}^{(\cdot)}$ is a size r oracle circuit of depth $d = O(\log(1/\epsilon))$ (over a set of gates that includes the standard boolean gates with unbounded fan-in)*
- *$n, a, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$ and $\delta \leq \delta_0$.*

Then there exists a circuit R of size poly(r), and depth $O(d)$ that uses the gates allowed to Red, and computes the majority function on inputs of length $\Omega(1/\epsilon)$.

Adapting the proofs to PRG constructions. We sketch the proofs of Theorem 5.9 and Theorem 5.10. The high level idea is to prove the following zoom lemma in this setting. The lemma is identical to Lemma 4.5 except that, in the oracle, Con_f is replace by a function Dist_f which we now define: on input $y \in \{0,1\}^n$, $\mathsf{Dist}_f(y)$ answers one iff there exists $s \in \{0,1\}^\ell$ such that $\mathsf{Con}_f(s) = y$.

Lemma 5.11 (Zoom Lemma for black-box PRG constructions)**.** *There exist constants $\delta_0, \nu > 0$ and $d > 1$ such that: Let $(\mathsf{Con}, \mathsf{Red})$ be a $\delta \to \epsilon$ black-box PRG construction with input lengths k, seed length ℓ, output length n, and list size 2^a. Suppose for every x and α the reduction Red is a circuit of size r that makes at most q oracle queries. Assume $n, a, q, \frac{1}{\epsilon} \leq r \leq 2^{\nu \cdot k}$, $\delta \leq \delta_0$, $\ell < n$. Let $\eta := \delta + 2^{-\nu \cdot k}$. Let $f : \{0,1\}^k \to \{0,1\}$ be a function that is $1/200$-hard for circuits of size $s = r^d$.*

There exists a set $B \subseteq \{0,1\}^n$ of size $(aq/\eta)^{O(1)}$, a string $v \in \{0,1\}^B$, $x \in \{0,1\}^k$ and $\alpha \in \{0,1\}^a$ such that

- $\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}-2\epsilon}}[\mathsf{Red}^{\mathsf{Dist}_f \oplus D}(x, \alpha) = f(x) | D(B) = v] \geq 1 - 2\sqrt{\delta} - 2^{-\nu \cdot k}$,
- $\Pr_{D \leftarrow \mathsf{Noise}^V_{\frac{1}{2}}}[\mathsf{Red}^{\mathsf{Dist}_f \oplus D}(x, \alpha) = f(x) | D(B) = v] \leq 0.51$.

The high level idea is the following. We consider two oracles: $\mathsf{Dist}_f \oplus \mathsf{Noise}^N_{\frac{1}{2}}$, and $\mathsf{Dist}_f \oplus \mathsf{Noise}^N_{\frac{1}{2}-2\epsilon}$. In the first oracle, the noise completely masks out the information in Dist_f, and so a reduction Red the receives access to this oracle cannot succeed. The second oracle ϵ-breaks Con_f, because $\ell < n$, and so with this oracle the reduction must succeed. From here the proof of the zoom lemma proceeds as in the case of hardness amplification. The two theorems now follow from the zoom lemma, in the same manner as in the case of hardness amplification.

6 Open problems

This paper concludes a line of research initiated in [Vio04, Vio06] by establishing that hardness amplification requires majority and many queries. Recall that a function f can be written as the majority of a polynomial number of functions from a class C if and only if for any distribution on the inputs there exists a function in C that computes f correctly with probability $\geq 1/2 + 1/\text{poly}$. (One direction can be proved via boosting [Fre95, Section 2.2] or min-max/linear-programming duality [GHR92, Section 5]. The other direction follows from the "discriminator lemma" of [HMP+93].) Hence we offer the following alternative interpretation of hardness amplification:

Black-box hardness amplification is a process that takes a function f that is *already* $1/2 - \epsilon$ *hard under some distribution*, and produces another function f' that has roughly the same hardness *under the uniform distribution.*

A fundamental question remains: is hardness amplification *false*? In particular, is the XOR lemma true for restricted circuit classes, such as constant-depth circuits with parity gates? One can show that the XOR lemma is false for the class of constant-depth circuits with one majority gate. This follows by the bounds in [ABFR94], and the fact that the XOR lemma applied to parity is again just parity. But that is essentially the only counterexample that we know.

Our results apply to black-box techniques. We remark that one possible way to break the black-box barrier is to come up with proofs that use the fact that D is a small circuit (bypassing the black-box limitations). A potential non-black-box approach was presented by Gutfreund, Shaltiel and Ta-Shma [GST07] (see also [Ats06, Gut06, GT07]) in a very specific scenario that has some similarity to "worst-case to average case reductions in NP". The techniques of these papers provably break black-box limitations in a related setting. See discussion by Gutfreund and Ta-Shma [GT07].

Another question, already highlighted in [SV10], is whether the construction of pseudorandom generators *with constant error* requires majority.

Acknowledgments. We are grateful to Iftach Haitner for very helpful discussions.

References

[AASY15] Benny Applebaum, Sergei Artemenko, Ronen Shaltiel, and Guang Yang. Incompressible functions, relative-error extractors, and the power of nondeterministic reductions (extended abstract). In *Conf. on Computational Complexity (CCC)*, pages 582–600, 2015.

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity*. Cambridge University Press, 2009. A modern approach.

[ABFR94] James Aspnes, Richard Beigel, Merrick Furst, and Steven Rudich. The expressive power of voting polynomials. *Combinatorica. An Journal on Combinatorics and the Theory of Computing*, 14(2):135–148, 1994.

[AS14] Sergei Artemenko and Ronen Shaltiel. Lower bounds on the query complexity of non-uniform and adaptive reductions showing hardness amplification. *Computational Complexity*, 23(1):43–83, 2014.

[Ats06] Albert Atserias. Distinguishing SAT from polynomial-size circuits, through black-box queries. In *IEEE Conf. on Computational Complexity (CCC)*, pages 88–95, 2006.

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

[BM84] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. on Computing*, 13(4):850–864, November 1984.

[BS07] Andrej Bogdanov and Muli Safra. Hardness amplification for errorless heuristics. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 418–426, 2007.

[CDGS18] Sandro Coretti, Yevgeniy Dodis, Siyao Guo, and John Steinberger. Random oracles and non-uniformity. In *Int. Conf. on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, 2018.

[CK82] Imre Csiszar and Janos Korner. *Information Theory: Coding Theorems for Discrete Memoryless Systems*. Academic Press, Inc., 1982.

[CT06] Thomas Cover and Joy Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

[DGK17] Yevgeniy Dodis, Siyao Guo, and Jonathan Katz. Fixing cracks in the concrete: Random oracles with auxiliary input, revisited. In *Int. Conf. on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pages 473–495, 2017.

[EIRS01] Jeff Edmonds, Russell Impagliazzo, Steven Rudich, and Jiri Sgall. Communication complexity towards lower bounds on circuit depth. *Computational Complexity*, 10(3):210–246, 2001.

[Fre95] Yoav Freund. Boosting a weak learning algorithm by majority. *Information and Computation*, 121(2):256–285, 1995.

[FSUV13] Bill Fefferman, Ronen Shaltiel, Christopher Umans, and Emanuele Viola. On beating the hybrid argument. *Theory of Computing*, 9:809–843, 2013.

[GGH+07] Shafi Goldwasser, Dan Gutfreund, Alexander Healy, Tali Kaufman, and Guy N. Rothblum. Verifying and decoding in constant depth. In *ACM Symp. on the Theory of Computing (STOC)*, pages 440–449, 2007.

[GHR92] Mikael Goldmann, Johan Håstad, and Alexander A. Razborov. Majority gates vs. general weighted threshold gates. *Computational Complexity*, 2:277–300, 1992.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. of Computer and System Sciences*, 28(2):270–299, April 1984.

[GNW95] Oded Goldreich, Noam Nisan, and Avi Wigderson. On Yao's XOR lemma. Technical Report TR95–050, *Electronic Colloquium on Computational Complexity*, March 1995. www.eccc.uni-trier.de/.

[GR08] Dan Gutfreund and Guy Rothblum. The complexity of local list decoding. In *12th Intl. Workshop on Randomization and Computation (RANDOM)*, 2008.

[GST07] Dan Gutfreund, Ronen Shaltiel, and Amnon Ta-Shma. If NP languages are hard on the worst-case, then it is easy to find their hard instances. *Computational Complexity*, 16(4):412–441, 2007.

[GT07] Dan Gutfreund and Amnon Ta-Shma. Worst-case to average-case reductions revisited. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 10th International Workshop, APPROX 2007, and 11th International Workshop, RANDOM 2007, Princeton, NJ, USA, August 20-22, 2007, Proceedings*, pages 569–583, 2007.

[Gut06] Dan Gutfreund. Worst-case vs. algorithmic average-case complexity in the polynomial-time hierarchy. In *Workshop on Randomization and Computation (RANDOM)*, pages 386–397, 2006.

[HMP+93] András Hajnal, Wolfgang Maass, Pavel Pudlák, Márió Szegedy, and György Turán. Threshold circuits of bounded depth. *J. of Computer and System Sciences*, 46(2):129–154, 1993.

[IJKW10] Russell Impagliazzo, Ragesh Jaiswal, Valentine Kabanets, and Avi Wigderson. Uniform direct product theorems: Simplified, optimized, and derandomized. *SIAM J. on Computing*, 39(4):1637–1665, 2010.

[Imp95] Russell Impagliazzo. Hard-core distributions for somewhat hard problems. In *IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 538–545, 1995.

[IW97] Russell Impagliazzo and Avi Wigderson. $P = BPP$ if E requires exponential circuits: Derandomizing the XOR lemma. In *29th ACM Symp. on the Theory of Computing (STOC)*, pages 220–229. ACM, 1997.

[IW01] Russell Impagliazzo and Avi Wigderson. Randomness vs time: Derandomization under a uniform assumption. *J. of Computer and System Sciences*, 63(4):672–688, 2001.

[Kli01] Adam R. Klivans. On the derandomization of constant depth circuits. In *Workshop on Randomization and Computation (RANDOM)*. Springer, 2001.

[KS03] Adam Klivans and Rocco A. Servedio. Boosting and hard-core sets. *Machine Learning*, 53(3):217–238, 2003.

[LTW11] Chi-Jen Lu, Shi-Chun Tsai, and Hsin-Lung Wu. Complexity of hard-core set proofs. *Computational Complexity*, 20(1):145–171, 2011.

[MV15] Eric Miles and Emanuele Viola. Substitution-permutation networks, pseudorandom functions, and natural proofs. *J. of the ACM*, 62(6), 2015.

[MW17] Or Meir and Avi Wigderson. Prediction from partial information and hindsight, with application to circuit lower bounds. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:149, 2017.

[Nis91] Noam Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica. An Journal on Combinatorics and the Theory of Computing*, 11(1):63–70, 1991.

[NR04] Moni Naor and Omer Reingold. Number-theoretic constructions of efficient pseudo-random functions. *J. of the ACM*, 51(2):231–262, 2004.

[NW94] Noam Nisan and Avi Wigderson. Hardness vs randomness. *J. of Computer and System Sciences*, 49(2):149–167, 1994.

[PV10] Mihai Pătraşcu and Emanuele Viola. Cell-probe lower bounds for succinct partial sums. In *21th ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 117–122, 2010.

[Raz87] Alexander Razborov. Lower bounds on the dimension of schemes of bounded depth in a complete basis containing the logical addition function. *Akademiya Nauk SSSR. Matematicheskie Zametki*, 41(4):598–607, 1987. English translation in Mathematical Notes of the Academy of Sci. of the USSR, 41(4):333-338, 1987.

[Raz98] Ran Raz. A parallel repetition theorem. *SIAM J. on Computing*, 27(3):763–803, 1998.

[RR97] Alexander Razborov and Steven Rudich. Natural proofs. *J. of Computer and System Sciences*, 55(1):24–35, August 1997.

[Sha02] Ronen Shaltiel. Recent developments in explicit constructions of extractors. *Bulletin of the European Association for Theoretical Computer Science*, 77:182–194, 2002.

[ST17] Alexander Smal and Navid Talebanfard. Prediction from partial information and hindsight, an alternative proof. *Electronic Colloquium on Computational Complexity (ECCC)*, 24:191, 2017.

[STV01] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the XOR lemma. *J. of Computer and System Sciences*, 62(2):236–266, 2001.

[SU05] Ronen Shaltiel and Christopher Umans. Simple extractors for all min-entropies and a new pseudorandom generator. *J. of the ACM*, 52(2):172–216, 2005.

[SU06] Ronen Shaltiel and Christopher Umans. Pseudorandomness for approximate counting and sampling. *Computational Complexity*, 15(4):298–341, 2006.

[SV10] Ronen Shaltiel and Emanuele Viola. Hardness amplification proofs require majority. *SIAM J. on Computing*, 39(7):3122–3154, 2010.

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

[TZ04] Amnon Ta-Shma and David Zuckerman. Extractor codes. *IEEE Trans. Information Theory*, 50(12):3015–3025, 2004.

[Uma03] Christopher Umans. Pseudo-random generators for all hardnesses. *J. of Computer and System Sciences*, 67(2):419–440, 2003. Special issue on STOC2002 (Montreal, QC).

[Unr07] Dominique Unruh. Random oracles and auxiliary input. In *Int. Cryptology Conf. (CRYPTO)*, pages 205–223, 2007.

[Vio04] Emanuele Viola. The complexity of constructing pseudorandom generators from hard functions. *Computational Complexity*, 13(3-4):147–188, 2004.

[Vio06] Emanuele Viola. *The Complexity of Hardness Amplification and Derandomization*. PhD thesis, Harvard University, 2006.

[Vio09a] Emanuele Viola. Cell-probe lower bounds for prefix sums, 2009. arXiv:0906.1370v1.

[Vio09b] Emanuele Viola. Gems of theoretical computer science. Lecture notes of the class taught at Northeastern University. Available at http://www.ccs.neu.edu/home/viola/classes/gems-08/index.html, 2009.

[Vio12] Emanuele Viola. Bit-probe lower bounds for succinct data structures. *SIAM J. on Computing*, 41(6):1593?–1604, 2012.

[Wat15] Thomas Watson. Query complexity in errorless hardness amplification. *Computational Complexity*, 24(4):823–850, 2015.

[Yao82] Andrew Yao. Theory and applications of trapdoor functions. In *23rd IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 80–91. IEEE, 1982.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il