On the Complexity of Parallel Hardness Amplification for One-Way Functions

Chi-Jen Lu

Institute of Information Science, Academia Sinica, Taipei, Taiwan
cjlu@iis.sinica.edu.tw

Abstract. We prove complexity lower bounds for the tasks of hardness amplification of one-way functions and construction of pseudo-random generators from one-way functions, which are realized non-adaptively in black-box ways.

First, we consider the task of converting a one-way function $f : \{0,1\}^n \to \{0,1\}^m$ into a harder one-way function $\bar{f} : \{0,1\}^{\bar{n}} \to \{0,1\}^{\bar{m}}$, with $\bar{n}, \bar{m} \le \text{poly}(n)$, in a black-box way. The hardness is measured as the fraction of inputs any polynomial-size circuit must fail to invert. We show that to use a constant-depth circuit to amplify hardness beyond a polynomial factor, its size must exceed $2^{\text{poly}(n)}$, and to amplify hardness beyond a $2^{o(n)}$ factor, its size must exceed $2^{2^{o(n)}}$. Moreover, for a constant-depth circuit to amplify hardness beyond an $n^{1+o(1)}$ factor in a security preserving way (with $\bar{n} = O(n)$), it size must exceed $2^{n^{o(1)}}$.

Next, we show that if a constant-depth polynomial-size circuit can amplify hardness beyond a polynomial factor in a weakly black-box way, then it must basically embed a hard function in itself. In fact, one can derive from such an amplification procedure a highly parallel one-way function, which is computable by an NC^0 circuit (constant-depth polynomial-size circuit with bounded fan-in gates).

Finally, we consider the task of constructing a pseudo-random generator $G : \{0,1\}^{\bar{n}} \to \{0,1\}^{\bar{m}}$ from a strongly one-way function $f : \{0,1\}^n \to \{0,1\}^m$ in a black-box way. We show that any such a construction realized by a constant-depth $2^{n^{o(1)}}$-size circuit can only have a sublinear stretch (with $\bar{m} - \bar{n} = o(\bar{n})$).

1 Introduction

One of the most fundamental notions in cryptography is that of one-way functions. Informally speaking, a one-way function is a function which is easy to compute but hard to invert. The adversaries we consider here are polynomial-size circuits, which are non-uniform versions of polynomial-time algorithms. We measure the hardness of a one-way function as the fraction of n-bit inputs on which such adversaries must fail to invert. A one-way function with hardness larger than $1 - 1/\text{poly}(n)$ is called a strongly one-way function, which is known to be sufficient for building a large number of cryptographical primitives. Can we further weaken the hardness assumption? Can we start from a one-way function which is only hard to invert in a worst-case sense (with hardness 2^{-n})? This has been a long-standing open problem in cryptography.

It is known that one can start from a weakly one-way function, a one-way function with hardness at least $1/\text{poly}(n)$. The transformation from a weakly one-way function to a strongly one-way function was first discovered by Yao [21], using the so-called direct product approach. The direct product approach has the advantage of being extremely simple and highly parallel. However, the drawback is that it blows up the input length and thus degrades the security (the hardness of the new function is now measured against much smaller circuits). Ideally, one would like to have a security preserving hardness amplification, in which the new function's input length is only increased by a constant factor. Goldreich et al. [7] gave the first security preserving hardness amplification which transforms any weakly one-way *permutation* to a strongly one-way *permutation* of the same input length. Their approach is based on taking random walks on expander graphs and is much more involved than the direct product approach. Moreover, the transformation requires a higher complexity and seems sequential in nature. Therefore, even if the initial function can be evaluated efficiently in parallel, it is not clear if the resulting function will be so. This raises the following question: can a security preserving hardness amplification be carried out in parallel or in a low complexity class?

Another fundamental primitive in cryptography is pseudo-random generator, which stretches a short random seed into a longer random-looking string. A celebrated result due to Håstad et al. shows that a pseudo-random generator can be constructed from any strongly one-way function [9]. A crucial parameter of a pseudo-random generator $G : \{0,1\}^r \rightarrow \{0,1\}^{r+s}$ is its stretch s. In several cryptographical applications, we need the stretch to be at least linear. The pseudo-random generator construction in [9] only has a sublinear stretch. In particular, the hard-core function approach can only extract $O(\log n)$ pseudo-random bits from a one-way function. Given a pseudo-random generator of sublinear stretch, one can increase the stretch to linear, but the known construction appears inherently sequential. In [20], Viola asked the question: can the construction of pseudo-random generators with linear stretch from one-way functions be realized efficiently in parallel?

In fact, a more general question is: can cryptographic constructions (or reductions) be realized in a low complexity class? Very little is known for the questions we raised above. For the task of hardness amplifications and pseudo-random generator constructions, there has been no success in realizing them in a low complexity class. Could they be impossible tasks? We would like to say so by showing that they basically all require a high complexity. However, it is not clear what this means. For example, suppose there indeed exists a strongly one-way function computed by a low-complexity procedure, then it gives a trivial hardness amplification procedure of low complexity: just ignore the initial weakly one-way function and compute the strongly one-way function from scratch.

Black-Box Constructions. One important paradigm of cryptographic constructions is the so-called black-box constructions [12], in which one cryptographic primitive is used as a black box to construct another cryptographic primitive. Call a hardness amplification for one-way functions a *black-box* one if the following two conditions hold. First, the initial function f is given as a black-box to construct the new function $\bar{f}$. That is, there is an oracle algorithm AMP such that $\bar{f} = \text{AMP}^f$, so $\bar{f}$ only uses f as an oracle and does not depend on the internal structure of f. Second, the hardness of the new function $\bar{f}$ is proved in a black-box way. That is, there is an oracle Turing machine DEC, such that given any A breaking the hardness of $\bar{f}$, DEC using A as an oracle can break the hardness of f. Again, DEC only uses A as an oracle and does not depend on the internal structure of A. We assume that the procedure DEC makes only a polynomial number of queries to the oracle, and we will study the complexity needed to realize the procedure AMP. In fact, all previous hardness amplification results (and almost all cryptographic reductions) were done in such a black-box way, so it is important to understand its limitation.

A hardness amplification is called a *weakly black-box* one if only the first condition above is required while the second is dropped, namely, without requiring the hardness of the new one-way function to be guaranteed in a black-box way. Note that it seems difficult to obtain negative results for weakly black-box constructions, because one could always build the function $\bar{f}$ from scratch if it exists (without relying on the function f). Therefore, showing that this is indeed the case is usually the best one could expect.

Similarly, one can also define the notion of black-box construction of pseudo-random generators from one-way functions.

Previous Lower Bound Results. Lin, Trevisan, and Wee [14] provided complexity lower bounds for black-box hardness amplification of one-way functions. They showed that to amplify a δ-hard function to an $(1-\varepsilon)$-hard function in a black-box way, the procedure AMP must make $q = \Omega((1/\delta)\log(1/\varepsilon))$ queries to the oracle, and the resulting new function must have an input length longer than that of the initial function by $\Omega(\log(1/\varepsilon)) - O(\log q)$ bits. They also showed that if there exists a weakly black-box transformation from a δ-hard *permutation* to an $(1-\varepsilon)$-hard *permutation* beating this lower bound, then one-way permutations exist unconditionally.

Viola [20] provided a complexity lower bound for black-box construction of pseudo-random generators from strongly one-way functions. He introduced the notion of *parallel* black-box construction, in which the procedure AMP works in the following way. Given an input $\bar{x} \in \{0,1\}^{\bar{n}}$, AMP first generates *non-adaptive* queries $x_1, \ldots, x_t \in \{0,1\}^n$ and an AC^0 (constant-depth polynomial-size) circuit A, then accesses the oracle f at these t places to obtain the values $y_1 = f(x_1), \ldots, y_t = f(x_t)$, and finally computes the value $A(y_1, \ldots, y_t)$ as its output. He then showed that if the procedure AMP is realized in this way, then the resulting pseudo-random generator can only have a sublinear stretch.

In a different setting, Lu, Tsai, and Wu [15] considered the hardness of computing Boolean functions instead of inverting one-way functions. They provided complexity lower bounds for procedures which amplify this kind of hardness.

Our Results. We adopt Viola's model [20] and consider hardness amplifications and pseudo-random generator constructions realized in a parallel (non-adaptive) way. Our first result shows that any black-box hardness amplification realized by a low-complexity procedure can not increase the hardness substantially. More precisely, consider any black-box hardness amplification which maps any ε-hard function $f : \{0,1\}^n \to \{0,1\}^m$ to an $\bar{\varepsilon}$-hard function $\bar{f} : \{0,1\}^{\bar{n}} \to \{0,1\}^{\bar{m}}$ with $\bar{n}, \bar{m} \leq \text{poly}(n)$. We show that a constant-depth circuit of $2^{\text{poly}(n)}$ size cannot amplify the hardness to any $\bar{\varepsilon} > \varepsilon \cdot \text{poly}(n)$, and a constant-depth circuit of $2^{2^{o(n)}}$ size cannot amplify the hardness to any $\bar{\varepsilon} > \varepsilon \cdot 2^{o(n)}$. This implies that a procedure in polynomial hierarchy (PH) cannot amplify hardness beyond a polynomial factor, and an alternating Turing machine with constant alternations and $2^{o(n)}$ time ($\mathsf{ATIME}(O(1), 2^{o(n)})$) cannot amplify hardness beyond a $2^{o(n)}$ factor. As a result, a procedure in PH cannot transform a one-way function with hardness lower than $1/\text{poly}(n)$ into a one-way function with constant hardness (let alone a strongly one-way function), and a procedure in $\mathsf{ATIME}(O(1), 2^{o(n)})$ cannot transform a one-way function with worst-case hardness into a weakly one-way function (let alone a strongly one-way function). Note that not only do we rule out the possibility of using a polynomial-time procedure for doing such hardness amplifications (as is usually hoped for in cryptography), we show that even a procedure in a high complexity class, such as PH (or $\mathsf{ATIME}(O(1), 2^{o(n)})$), can not do the job. This just demonstrates how difficult the task is. Moreover, we show that to have $\bar{n} = O(n)$, a constant-depth circuit of $2^{n^{o(1)}}$ size cannot amplify the hardness to any $\bar{\varepsilon} > \varepsilon \cdot n^{1+o(1)}$. This explains why the security preserving hardness amplification procedures of [7, 4] are sequential while the parallel hardness amplification procedure by direct product [21] blows up the input length: they are all done in a black-box way.

Our second result shows that if a parallel weakly black-box hardness amplification can increase the hardness substantially, then it must basically embed a one-way function in itself. More precisely, consider any weakly black-box hardness amplification which maps any ε-hard function $f : \{0,1\}^n \to \{0,1\}^m$ to an $\bar{\varepsilon}$-hard function $\bar{f} : \{0,1\}^{\bar{n}} \to \{0,1\}^{\bar{m}}$. We show that if an AC^0 circuit can amplify the hardness to $\bar{\varepsilon} > \sqrt{\varepsilon} \cdot \text{poly}(n)$, then one can derive from it a one-way function computable in NC^1 with hardness roughly $\bar{\varepsilon}$. From [2], this implies the existence of a one-way function computable in NC^0. This is interesting in the following sense. Consider one-way functions which are computed in polynomial time or even in a higher complexity class. It is possible for a low-complexity procedure, say in AC^0, to amplify hardness for such functions, for example using the direct product approach [21]. However, if it amplifies hardness beyond a polynomial factor, we can derive from such an amplification procedure a one-way function which is computable in NC^0, an extremely low complexity class.

Our third result extends Viola's lower bound for black-box constructions of pseudo-random generators [20]. We show that any black-box construction of pseudo-random generators from strongly one-way functions realized by a constant-depth circuit can only have a sublinear stretch unless the circuit size is exponential. This improves the super-polynomial lower bound of Viola [20].

Our Techniques. We follow the approach of Viola [20], which relies on the fact that applying random restrictions on the input of AC^0 circuits are likely to make their output bits biased since such circuits are insensitive to noise on their input [13, 3]. A similar idea was also used in [15]. However, since our setting is different, we have different problems to solve.

Assume that an AC^0 circuit can amplify hardness beyond a certain bound (the idea can be generalized to a larger class of circuits). It is known that a random function f is likely to be one-way. As shown in [20], it is still likely to be so even with a random restriction ρ applied to its output bits, as long as ρ gives each output bit the symbol $\star$ (leave the bit free) at a rate above some threshold. On the other hand, AC^0 circuits are likely to become biased after applying a random restriction on its input. As the rate of $\star$ decreases, the effect a random f on $\textsc{Amp}^{f\restriction_\rho}(\bar{x})$ becomes smaller, for any input $\bar{x}$. If the rate of $\star$ is small enough, the functions $\textsc{Amp}^{f\restriction_\rho}$'s for most f become close to each other (agreeing with each other on most inputs). As a result, they are close to some fixed function (depending on ρ) which can then be used as an oracle to invert $f\restriction_\rho$. This would lead to a contradiction, and we could conclude that such hardness amplification cannot be realized by AC^0 circuits.

However, there is an obstacle in front us. In order to guarantee that the functions $\textsc{Amp}^{f\restriction_\rho}$'s for most f are close to each other, we need the random restriction to give $\star$ in a very low rate. Had we applied a conventional random restriction, say from [5, 8] (as was done in [20]), we would end up having too few free bits left in $f(x)$ for almost every x, and consequently $f\restriction_\rho$ would not be one-way for most f. To overcome this problem, we would like the $\star$'s to appear in a somewhat clustered fashion: for any x, either $f(x)$ has no $\star$ at all, or it has a sufficient number of $\star$'s. This motivates us to consider a new kind of random restriction (described in Section 3), and we show that it also makes the output bits of AC^0 circuits highly biased.

This new kind of random restriction also helps us improve the result of Viola. In [20], a super-polynomial size lower bound was shown for black-box constructions of pseudo-random generators from one-way functions. What prevents the argument there from getting a better bound is exactly the same obstacle we just discussed above. Namely, to guarantee $f\restriction_\rho$ being one-way using a conventional random restriction, the rate of $\star$ cannot be too low, which fails to make the output bits of larger circuits biased enough. With the help of our new random restriction, we are able to overcome this problem and obtain an exponential lower bound.

Another technical contribution of ours is in the derivation of one-way functions from weakly black-box hardness amplification procedures. In the different setting of Boolean functions, if a function $f : \{0,1\}^n \to \{0,1\}$ agrees on most inputs with a hard-to-compute function $f' : \{0,1\}^n \to \{0,1\}$ (any adversary fails to compute f' correctly on a large portion of inputs), then f itself must also be hard enough, which can be proved in a black-box way. However, this does not seem to be the case for one-way functions. That is, even though a function $f : \{0,1\}^n \to \{0,1\}^m$ is close to a hard-to-invert function $f' : \{0,1\}^n \to \{0,1\}^m$, it is not clear if f itself must also be hard to invert. In fact, this cannot be proved in a black-box way (more in Section 5). The technique in [14] faces the same problem, and the result there is only on weakly hardness amplification which produces one-way *permutations*, since the injective condition makes the problem disappear. As we consider a more restricted type of hardness amplification, that realizable in parallel, we are able overcome this difficulty and obtain results for weakly hardness amplification which produces general one-way *functions*.

2 Preliminaries

For any $n \in \mathbb{N}$, let $[n]$ denote the set $\{1, 2, \ldots, n\}$ and let $\mathcal{U}_n$ denote the uniform distribution over the set $\{0,1\}^n$. When sampling from a finite set, the default distribution we use is the uniform one. For a string $x \in \Sigma^n$, let x_i, for $i \in [n]$, denote the entry in the i'th dimension of x, and let x_I, for $I \subseteq [n]$, denote the substring of x which is the projection of x onto those dimensions in I.

We will consider functions computed by Boolean circuits of AND/OR/NOT gates. Let $\mathsf{NC^i}$ denote the class of functions computed by circuits of depth $O(\log^i n)$ and size $\mathrm{poly}(n)$ with *bounded* fan-in gates. Let $\mathsf{AC}(d, s)$ denote the class of functions computed by circuits of depth d and size s with *unbounded* fan-in gates. Let $\mathsf{AC^0}(s)$ denote the class $\mathsf{AC}(O(1), s)$, and note that $\mathsf{AC^0}(\mathrm{poly}(n))$ corresponds to the standard complexity class $\mathsf{AC^0}$. Let $\mathsf{ATIME}(d, t)$ denote the class of functions computed by alternating Turing machines in time t with d alternations. The class $\mathsf{ATIME}(O(1), \mathrm{poly}(n))$ corresponds to the polynomial-time hierarchy PH. More information about complexity classes can be found in standard textbooks, such as [18].

Next, we will introduce the notion of one-way functions and pseudo-random generators. Informally speaking, a function is called a one-way function if it is easy to compute but hard to invert. For a many-to-one function f, we say that an algorithm M inverts $f(x)$ if $M(f(x))$ is in the preimage of $f(x)$, namely, $f(M(f(x))) = f(x)$. When we mention a function $f : \{0,1\}^n \to \{0,1\}^m$, we usually mean a sequence of functions $(f : \{0,1\}^n \to \{0,1\}^{m(n)})_{n\in\mathbb{N}}$, and when we make a statement about f, we usually mean that it holds for any sufficiently large $n \in \mathbb{N}$.

Definition 1. *A function $f : \{0,1\}^n \to \{0,1\}^m$ is (n, m, ε)-hard, or ε-hard for short, if for any polynomial-size circuit M, $\Pr_{x\in\mathcal{U}_n}\left[M^f \text{ fails to invert } f(x)\right] \geq \varepsilon$. A function $f : \{0,1\}^n \to \{0,1\}^m$ is an (n, m, ε)-OWF, or ε-OWF for short, if it can be computed in polynomial time but is ε-hard to invert.*

A pseudo-random generator is a function which stretches a short random seed into a longer random-looking string.

Definition 2. *A function $M : \{0,1\}^m \to \{0,1\}$ ε-distinguishes a function $g : \{0,1\}^n \to \{0,1\}^m$ if $|\Pr_{x\in\mathcal{U}_n}[M(g(x)) = 1] - \Pr_{y\in\mathcal{U}_m}[M(y) = 1]| > \varepsilon$. A function $g : \{0,1\}^n \to \{0,1\}^m$, with $n < m$, is an (n, m, ε)-PRG, or ε-PRG for short, if it can be computed in polynomial time, but no polynomial-size circuit can ε-distinguish g.*

2.1 Black-Box Constructions

Next, we introduce the notion of black-box hardness amplification.

Definition 3. *A black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions consists of two oracle algorithms* Amp *and* Dec *satisfying the following two conditions. First, for any $f : \{0,1\}^n \to \{0,1\}^m$,* Amp^f *is a function from $\{0,1\}^{\bar{n}}$ to $\{0,1\}^{\bar{m}}$. Second,* Dec *makes at most* $\text{poly}(n)$ *oracle queries, and for any $f : \{0,1\}^n \to \{0,1\}^m$ and $\bar{M} : \{0,1\}^{\bar{m}} \to \{0,1\}^{\bar{n}}$, if* $\Pr_{\bar{x} \in \mathcal{U}_{\bar{n}}}[\bar{M}$ *inverts* $\text{Amp}^f(\bar{x})] > 1 - \bar{\varepsilon}$,[1] *then* $\Pr_{x \in \mathcal{U}_n}[\text{Dec}^{\bar{M}, f}$ *inverts* $f(x)] > 1 - \varepsilon$.

Here the transformation of the initial function f into a harder function is done in a black-box way, as the harder function Amp^f only uses f as an oracle. Furthermore, the hardness of Amp^f is also guaranteed in a black-box way, in the sense that any algorithm $\bar{M}$ breaking the hardness condition of Amp^f can be used as an oracle for Dec to break the hardness condition of f. We call Amp the *encoding procedure* and Dec the *decoding procedure*.

A weaker notion is the following weakly black-box hardness amplification, in which only the encoding is required to be done in a black-box way.

Definition 4. *A weakly black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions consists of an oracle algorithm* Amp *such that* Amp^f *is $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard given any (n, m, ε)-hard function f.*

Following [20], we consider the notion of *parallel* black-box hardness amplification. In [20], only the case with $d = O(1)$ and $s \leq \text{poly}(n)$ was considered, but here we allow arbitrary d and s. This makes our impossibility results stronger, since we rule out a larger class of hardness amplification procedures.

Definition 5. *We say that a black-box hardness amplification is realized by* $\mathsf{AC}(d, s)$ *if the following additional condition holds. Given any $\bar{x} \in \{0,1\}^{\bar{n}}$,* Amp *first produces an* $\mathsf{AC}(d, s)$ *circuit A and makes $t \leq \text{poly}(s)$* non-adaptive *queries $x_1, \ldots, x_t \in \{0,1\}^n$ to the oracle to obtain answers $y_1, \ldots, y_t \in \{0,1\}^m$, and then computes its output as $A(y_1, \ldots, y_t)$.*

Note that $x_1, \ldots, x_t$ and A only depend on $\bar{x}$ and are independent of the oracle f. For the black-box case, no complexity constraint is placed on the part of generating the queries and the circuit, which again makes our impossibility results stronger. For the weakly black-box case, we need this part to be computed by an $\mathsf{AC}(d, s)$ circuit too, since we want to derive from the procedure Amp an efficiently computable one-way function. Similarly, one can define the notion of black-box construction of pseudo-random generators from hard functions, which is omitted here and can be found in [20].

---

[1] Here we consider the case that $\bar{M}$ does not query Amp^f. This makes such hardness amplification easier to find and our impossibility results stronger.

2.2 Limited Independence

A sequence of random variables is called k-wise independent if any k of them are independent. It is well known that such a space can be sampled in a randomness-efficient way.

Fact 1 *Any k-wise independent random variables $X_1, \ldots, X_N \in V$ can be generated in polynomial time using a seed of length $O(k(\log N + \log |V|))$.*

A sequence of variables is called (k, δ)-wise independent if any k of them together has a statistical distance at most δ to the uniform distribution. We need efficient constructions of such a space from [16, 1]. From this, we can obtain the following, whose proof is omitted due to the space constraint.

Lemma 1. *Suppose $b \geq t^2/\varepsilon^3$. Then there exists a family $\bar{\mathcal{H}}$ of hash functions from $\{0,1\}^n$ to $[b]$ which can be sampled using a seed of length $r_0 = O(\log n + \log b + \log(1/\varepsilon))$ and satisfies the following two properties.*

1. *For any distinct $x_1, \ldots, x_t \in \{0,1\}^n$, the probability over $h \in \bar{\mathcal{H}}$ that $h(x_i) = h(x_j)$ for some $i \neq j$ is at most $o(\varepsilon)$.*
2. *For any $S \subseteq [b]$ of size $3\varepsilon b$, the probability over $h \in \bar{\mathcal{H}}$ that $h(x) \in S$ for less than 2ε fraction of x is at most $o(\varepsilon)$.*

2.3 Fourier Analysis

As in [20], we will apply Fourier analysis on Boolean functions. For $N \in \mathbb{N}$ and $I \subseteq [N]$, define the function $\chi^I : \{-1,1\}^n \to \{-1,1\}$ as $\chi^I(x) = \prod_{i \in I} x_i$ for any $x \in \{-1,1\}^N$. For any $C : \{-1,1\}^N \to \{-1,1\}$ and any $I \subseteq [N]$, let $\hat{C(I)} = \mathrm{E}_{x \in \{-1,1\}^N}[C(x) \cdot \chi^I(x)]$. Here are some useful facts.

Fact 2 *For any $C : \{-1,1\}^N \to \{-1,1\}$ and for any $x \in \{-1,1\}^N$, $C(x) = \sum_I \hat{C}(I) \cdot \chi^I(x)$.*

Lemma 2. *[19] For any $C : \{-1,1\}^N \to \{-1,1\} \in \mathsf{AC}(d,s)$, $\sum_I \hat{C}(I)^2 (1 - 2\delta)^{|I|} \geq 1 - O(\delta \log^{d-1} s)$.*

3 Random Restriction

We will need the notion of random restriction [5, 8]. A restriction ρ on m variables is an element of $\{0,1,\star\}^m$, or seen as a function $\rho : [m] \to \{0,1,\star\}$. A variable is fixed by ρ if it receives a value in $\{0,1\}$ while a variable remains free if it receives the symbol $\star$. For a string $y \in \{0,1\}^m$ and a restriction $\rho \in \{0,1,\star\}^m$, let $y\restriction_\rho \in \{0,1\}^m$ be the restriction of y with respect to ρ: for $i \in [m]$, the i'th bit of $y\restriction_\rho$ is y_i if $\rho_i = \star$ and is ρ_i if $\rho_i \in \{0,1\}$. For a string $z \in \{0,1,\star\}^m$, let $\#_\star(z)$ denote the number of i's such that $z_i = \star$.

As in [20], we will consider applying a random restriction to a function $f : \{0,1\}^n \to \{0,1\}^m$ in the following sense. Take a restriction $\rho \in \{0,1,\star\}^{2^n m}$, seen as a function $\rho : \{0,1\}^n \to \{0,1,\star\}^m$, let $f\restriction_\rho$ be the function from $\{0,1\}^n$ to $\{0,1\}^m$ such that for $x \in \{0,1\}^n$, $f\restriction_\rho(x) = f(x)\restriction_{\rho(x)}$, the result of applying the restriction $\rho(x) \in \{0,1,\star\}^m$ on $f(x) \in \{0,1\}^m$.

Let $\mathcal{R}_\delta^m$ denote the random restriction (distribution over restrictions) on m variables such that each variable independently receives the symbol $\star$ with probability δ, the value 1 with probability $(1-\delta)/2$, and the value 0 with probability $(1-\delta)/2$. For our purpose later, we will need a new kind of random restriction.

Definition 6. *Let $\mathcal{R}_{\alpha,\beta}^{1,m}$ be the random restriction on m variables defined as $\mathcal{R}_{\alpha,\beta}^{1,m} = \alpha \cdot \mathcal{R}_\beta^m + (1-\alpha)\cdot \mathcal{R}_0^m$. That is, $\mathcal{R}_{\alpha,\beta}^{1,m}$ distributes as $\mathcal{R}_\beta^m$ with probability α and as $\mathcal{R}_0^m = \mathcal{U}_m$ with probability $1-\alpha$. Let $\mathcal{R}_{\alpha,\beta}^{t,m}$ be the random restriction on tm variables, defined as $\mathcal{R}_{\alpha,\beta}^{t,m} = (\mathcal{R}_{\alpha,\beta}^{1,m})^t$, namely, t independent copies of $\mathcal{R}_{\alpha,\beta}^{1,m}$.*

It is known that AC^0 circuits are insensitive to noise and (standard kind of) random restrictions are likely to make their output values highly biased [13, 3, 20]. We show that this is still true with respect to our new kind of random restrictions.

Lemma 3. *For any $C : \{0,1\}^{tm} \to \{0,1\} \in \mathsf{AC}(d,s)$, the probability over $\rho \in \mathcal{R}_{\alpha,\beta}^{t,m}$ and $y, y' \in \mathcal{U}_{tm}$ that $C(y\restriction_\rho) \neq C(y'\restriction_\rho)$ is at most $O(\alpha\beta \log^{d-1} s)$.*

Proof. We would like to apply Fourier analysis on C, so for now let us use $\{-1,1\}$ for the binary values $\{0,1\}$. Partition the tm input positions evenly into t parts $B_1, \ldots, B_t$ of size m, with $B_i = \{(i-1)m+1, \ldots, im\}$.

We know that $\Pr_{\rho;y,y'}[C(y\restriction_\rho) \neq C(y'\restriction_\rho)] = \frac{1}{2}\left(1 - \mathrm{E}_{\rho;y,y'}[C(y\restriction_\rho)\cdot C(y'\restriction_\rho)]\right)$. From Fact 2, $\mathrm{E}_{\rho;y,y'}[C(y\restriction_\rho)\cdot C(y'\restriction_\rho)]$ is equal to

$$\begin{aligned}
&\mathop{\mathrm{E}}_{\rho;y,y'}\left[\left(\sum_{I\subseteq[tm]} \hat{C}(I)\chi^I(y\restriction_\rho)\right)\cdot\left(\sum_{J\subseteq[tm]} \hat{C}(J)\chi^J(y'\restriction_\rho)\right)\right] \\
&= \sum_{I,J\subseteq[tm]} \hat{C}(I)\cdot\hat{C}(J)\cdot \mathop{\mathrm{E}}_{\rho;y,y'}\left[\chi^I(y\restriction_\rho)\cdot\chi^J(y'\restriction_\rho)\right].
\end{aligned}$$

To bound the expectation $\mathrm{E}_{\rho;y,y'}\left[\chi^I(y\restriction_\rho)\cdot\chi^J(y'\restriction_\rho)\right]$, consider two cases.

Case 1: $I \neq J$. There must exist some block B_i such that $B_i \cap I \neq B_i \cap J$. Observe that $\mathrm{E}_{\rho;y,y'}[\chi^I(y\restriction_\rho)\cdot\chi^J(y'\restriction_\rho)]$ is qual to

$$\begin{aligned}
&\mathop{\mathrm{E}}_{\rho;y,y'}\left[(\chi^{I\cap B_i}(y\restriction_\rho)\cdot\chi^{J\cap B_i}(y'\restriction_\rho))(\chi^{I\setminus B_i}(y\restriction_\rho)\cdot\chi^{J\setminus B_i}(y'\restriction_\rho))\right] \\
&= \mathop{\mathrm{E}}_{\rho;y,y'}\left[\chi^{I\cap B_i}(y\restriction_\rho)\cdot\chi^{J\cap B_i}(y'\restriction_\rho)\right]\mathop{\mathrm{E}}_{\rho;y,y'}\left[\chi^{I\setminus B_i}(y\restriction_\rho)\cdot\chi^{J\setminus B_i}(y'\restriction_\rho)\right],
\end{aligned}$$

where the second equality is because $\chi^{I\cap B_i}(y\restriction_\rho)\cdot\chi^{J\cap B_i}(y'\restriction_\rho)$ and $\chi^{I\setminus B_i}(y\restriction_\rho)\cdot\chi^{J\setminus B_i}(y'\restriction_\rho)$ are distributed independently. Note that

$$\mathop{\mathrm{E}}_{\rho;y,y'}\left[\chi^{I\cap B_i}(y\restriction_\rho)\cdot\chi^{J\cap B_i}(y'\restriction_\rho)\right] = \mathop{\mathrm{E}}_{\rho_i;y_i,y_i'}\left[\chi^{I\cap B_i}(y_i\restriction_{\rho_i})\cdot\chi^{J\cap B_i}(y_i'\restriction_{\rho_i})\right],$$

with $\rho_i \in \mathcal{R}^{1,m}_{\alpha,\beta} = (1-\alpha)\cdot\mathcal{R}^m_0 + \alpha\cdot\mathcal{R}^m_\beta$ and $y_i, y'_i \in \mathcal{U}_m$, so the expectation is

$$(1-\alpha)\cdot \mathop{\mathrm{E}}_{\rho_i\in\mathcal{R}^m_0;y_i,y'_i}\left[\chi^{I\cap B_i}(y_i{\restriction_{\rho_i}})\cdot\chi^{J\cap B_i}(y'_i{\restriction_{\rho_i}})\right]$$
$$+\alpha\cdot \mathop{\mathrm{E}}_{\rho_i\in\mathcal{R}^m_\beta;y_i,y'_i}\left[\chi^{I\cap B_i}(y_i{\restriction_{\rho}})\cdot\chi^{J\cap B_i}(y'_i{\restriction_{\rho_i}})\right],$$

which is $0+0=0$. This implies that $\mathrm{E}_{\rho;y,y'}[\chi^I(y{\restriction_\rho})\cdot\chi^J(y'{\restriction_\rho})]=0$ when $I\neq J$.

Case 2: $I=J$. Partition I into t parts $I_1,\ldots,I_t$ where $I_i = I\cap B_i$. Then,

$$\begin{aligned}
\mathop{\mathrm{E}}_{\rho;y,y'}\left[\chi^I(y{\restriction_\rho})\cdot\chi^I(y'{\restriction_\rho})\right] &= \mathop{\mathrm{E}}_{\rho;y,y'}\left[\prod_{i\in[t]}\chi^{I_i}(y_i{\restriction_{\rho_i}})\cdot\chi^{I_i}(y'_i{\restriction_{\rho_i}})\right]\\
&=\prod_{i\in[t]}\mathop{\mathrm{E}}_{\rho_i;y_i,y'_i}\left[\chi^{I_i}(y_i{\restriction_{\rho_i}})\cdot\chi^{I_i}(y'_i{\restriction_{\rho_i}})\right]\\
&=\prod_{i\in[t]}\left((1-\alpha)\cdot 1+\alpha\cdot(1-\beta)^{|I_i|}\right)\\
&\geq\prod_{i\in[t]}(1-\alpha\beta)^{|I_i|}\\
&=(1-\alpha\beta)^{|I|},
\end{aligned}$$

where the inequality follows from Jensen's inequality.[2]

Combining the two cases, we have $\mathrm{E}_{\rho;y,y'}\left[C(y{\restriction_\rho})\cdot C(y'{\restriction_\rho})\right]$ equal to

$$\sum_I \hat{C}(I)^2\cdot\mathop{\mathrm{E}}_{\rho;y,y'}\left[\chi^I(y{\restriction_\rho})\cdot\chi^I(y'{\restriction_\rho})\right]\geq\sum_I\hat{C}(I)^2\cdot(1-\alpha\beta)^{|I|},$$

which equals to $1-O(\alpha\beta\log^{d-1}s)$ by Lemma 2. Then,

$$\Pr_{\rho;y,y'}\left[C(y{\restriction_\rho})\neq C(y'{\restriction_\rho})\right]=\frac{1}{2}\left(1-\mathop{\mathrm{E}}_{\rho;y,y'}\left[C(y{\restriction_\rho})\cdot C(y'{\restriction_\rho})\right]\right)=O(\alpha\beta\log^{d-1}s).$$

□

Note that a random restriction from $\mathcal{R}^{1,m}_{\alpha,\beta}$ can be sampled using a seed of length $\ell_1+m\ell_2$ consisting of $m+1$ parts. The first part of the seed has length $\ell_1=O(\log(1/\alpha))$ and is used to determine whether the restriction $\mathcal{R}^m_\beta$ or $\mathcal{R}^m_0$ is applied. The remaining m parts of the seed, each of length $\ell_2=O(\log(1/\beta))$, are used to generate the m symbols in $\{0,1,\star\}$. For simplicity, we use a longer seed of length $\ell=(m+1)\ell_0$ and let each part have the same length $\ell_0=\max(\ell_1,\ell_2)$.

Furthermore, there is an $\mathsf{AC}^0(\mathrm{poly}(\ell))$ circuit W which given such a random seed of length ℓ produces the random restriction $\mathcal{R}^{1,m}_{\alpha,\beta}$. Thus, a random restriction from $\mathcal{R}^{b,m}_{\alpha,\beta}$ can be sampled using a seed of length $b\ell$ and produced by an $\mathsf{AC}^0(\mathrm{poly}(b\ell))$ circuit W^b, the concatenation of b independent copies of W.

[2] Consider the function $f(x)=(1-\beta x)^k$, which is convex for x in the interval $[0,1]$. Then $(1-\alpha)\cdot1+\alpha\cdot(1-\beta)^k=(1-\alpha)\cdot f(0)+\alpha\cdot f(1)\geq f((1-\alpha)\cdot0+\alpha\cdot1)=(1-\alpha\beta)^k$.

4 Black-Box Hardness Amplification

In this section, we study black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions. We will show that no such hardness amplification realized by $\mathsf{AC}^0(2^{\mathrm{poly}(n)})$ can amplify the hardness to *any* $\bar{\varepsilon} > \varepsilon \cdot \mathrm{poly}(n)$ while keeping the function's output or input length to $\mathrm{poly}(n)$. Our main technical result is the following.

Theorem 1. *No black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions can be realized by $\mathsf{AC}(d, s)$ with $\varepsilon \leq \bar{\varepsilon} \cdot \gamma$, for any $\gamma \leq o(m/(\bar{m} \log^{d+1} s))$ and any $s \geq \mathrm{poly}(n)$.*

Since any $\mathsf{ATIME}(d, t)$ computation with an oracle can be simulated by an $\mathsf{AC}(O(d), 2^{O(dt)})$ circuit with oracle answers given as part of its input, we have the following. In particular, with $\bar{m} \leq \mathrm{poly}(m)$, no such hardness amplification can be realized in PH for any $\bar{\varepsilon} \geq \varepsilon \cdot n^{\omega(1)}$, and nor can it be realized in $\mathsf{ATIME}(O(1), 2^{o(n)})$ for any $\bar{\varepsilon} \geq \varepsilon \cdot 2^{\Omega(n)}$.

Corollary 1. *No black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions can be realized in $\mathsf{ATIME}(d, t)$ with $\varepsilon \leq \bar{\varepsilon} \cdot m/(\bar{m} \cdot t^{cd})$ for some constant c.*

Theorem 1 states that a low-complexity procedure cannot amplify the hardness substantially without blowing up the output length. Next, we show that one cannot avoid blowing up the input length either. In particular, no $\mathsf{AC}^0(2^{n^{o(1)}})$ circuit can amplify hardness beyond an $n^{1+o(1)}$ factor in a security preserving way (with $\bar{n} = O(n)$).

Theorem 2. *No black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions can be realized by $\mathsf{AC}(d, s)$ with $\varepsilon \leq \bar{\varepsilon} \cdot \gamma$, for any $\gamma \leq o(n/(\bar{n} \log^{2d+1} s))$ when $s \geq 2^{\Omega(n^{1/(d-1)})}$, or for any $\gamma \leq o(n/(\bar{n} n^{(2d+1)/(d-1)}))$ when $s \leq 2^{O(n^{1/(d-1)})}$.*

4.1 Proof of Theorem 1

Assume that such a hardness amplification exists, with $\textsc{Amp}$ realized by $\mathsf{AC}(d, s)$ and $\varepsilon = o(\bar{\varepsilon} \cdot m/(\bar{m} \log^{d+1} s))$. We will show that this leads to a contradiction. The idea is the following. First, we show that for a random function f and a suitable random restriction ρ, the resulting function $f\lceil_\rho$ is likely to be one-way. The key is to show that for a sufficient number of x, ρ leaves enough bits in $f(x)$ free. Next, we show that such a random restriction is likely to kill off the effect of a random function f on $\textsc{Amp}^{f\lceil_\rho}$ so that the functions $\textsc{Amp}^{f\lceil_\rho}$'s for most f's are close to each other. The key is to show that an $\mathsf{AC}(d, s)$ circuit is likely to become highly biased after such a random restriction. This yields a way to invert $\textsc{Amp}^{f\lceil_\rho}$ well for most f's, which can then be used as an oracle to invert $f\lceil_\rho$, and we have a contradiction. To make sure that both conditions above hold, we need the random restriction to give $\star$'s at a very small rate but in a clustered way:

$f(x)$ receives no $\star$ at all for most x, but gets an enough number of $\star$'s for the rest. This motivates us to consider the new random restriction $\mathcal{R}_{\alpha,\beta}^{b,m}$ introduced in Section 3.

As in [20], we would like to make sure that a restriction does not give away too much information about the input, so that the function $f\lceil_\rho$ is one-way even given ρ. Therefore we will hash the input from the space $\{0,1\}^n$ down to a smaller space $[b]$ before applying the restriction from $\mathcal{R}_{\alpha,\beta}^{b,m}$. Here we choose the following parameters:

$$\alpha = 2\varepsilon,\ \beta = (\log^2 s)/m, \text{ and } b = t^2/\varepsilon^3.$$

Let $\mathcal{H}$ denote the set of functions from $\{0,1\}^n$ to $[b]$. Then define our random restriction $\mathcal{R}$ as the uniform distribution over the set of restrictions $\sigma \circ h : \{0,1\}^n \to \{0,1,\star\}^m$, with $h \in \mathcal{H}$ and $\sigma \in \mathcal{R}_{\alpha,\beta}^{b,m}$. Let $\mathcal{F}$ denote the set of functions from $\{0,1\}^n$ to $\{0,1\}^m$.

Definition 7. *We call a restriction $\rho : \{0,1\}^n \to \{0,1,\star\}^m$* good *if both of the following two conditions hold:*

1. $\Pr_{x\in\mathcal{U}_n}[\#_\star(\rho(x)) \geq \beta m/2] \geq (2/3)\alpha$.
2. $\Pr_{\bar{x}\in\mathcal{U}_{\bar{n}};f,f'\in\mathcal{F}}[\textsc{Amp}^{f\lceil_\rho}(\bar{x}) \neq \textsc{Amp}^{f'\lceil_\rho}(\bar{x})] = o(\bar{\varepsilon})$.

Note that if we use a traditional random restriction (of [5, 8]) as in [20], it is unlikely to have both conditions hold at the same time, because the second condition requires a low rate of $\star$ (lower than $\bar{\varepsilon}/(\bar{m}\log^{d-1} s)$) which makes the first condition unlikely to hold. On the other hand, using our new random restriction, we can have both conditions hold with high probability.

Lemma 4. $\Pr_{\rho\in\mathcal{R}}[\rho$ *is not good*$] = o(1)$.

Due to the space limitation, we defer the proof to the journal version and only sketch the idea here. To show that the first condition fails with a small probability, note that about α fraction of x's are turned "on" in the sense that it receives the restriction from $\mathcal{R}_\beta^m$ and should have $\#_\star(\rho(x))$ about βm, so large deviation from this has a small probability. To show that the second condition fails with a small probability, note that for any $\bar{x} \in \{0,1\}^{\bar{n}}$, most $\rho \in \mathcal{R}$ can kill off the effect of a random function f so that the value $\textsc{Amp}^{f\lceil_\rho}(\bar{x})$ is the same for most $f \in \mathcal{F}$, which is guaranteed by Lemma 3, with $\alpha\beta = O((\varepsilon \log^2 s)/m) = o(\bar{\varepsilon}/(\bar{m}\log^{d-1} s))$.

Next, we show that for a good ρ, the function $f\lceil_\rho$ is ε-hard for most $f \in \mathcal{F}$. In fact, as will be needed later, we prove hardness against slightly stronger algorithms: algorithms which can depend on ρ and have arbitrarily high complexity but make only a polynomial number of queries to $f\lceil_\rho$.

Lemma 5. *For any good ρ, for any M_ρ making at most* $\mathrm{poly}(n)$ *oracle queries,* $\Pr_{x\in\mathcal{U}_n,f\in\mathcal{F}}[M_\rho^{f\lceil_\rho}$ *inverts* $f\lceil_\rho(x)] \leq 1-\varepsilon$.

Due to space limitation, we defer the proof to the journal version. The argument is somewhat standard, which can be modified, say, from [20, 6].

This implies that for any good ρ, the function $\bar{A}_\rho$, defined by $\bar{A}_\rho(\bar{x}) = \max \arg_z \Pr_{f\in\mathcal{F}}[\textsc{Amp}^{f\restriction_\rho}(\bar{x}) = z]$, is close to $\textsc{Amp}^{f\restriction_\rho}$ for most f, because

$$\Pr_{\bar{x},f}\left[\bar{A}_\rho(\bar{x}) \neq \textsc{Amp}^{f\restriction_\rho}(\bar{x})\right] \leq \Pr_{\bar{x},f,f'}\left[\textsc{Amp}^{f\restriction_\rho}(\bar{x}) \neq \textsc{Amp}^{f'\restriction_\rho}(\bar{x})\right] = o(\bar{\varepsilon}).$$

This then provides us a way to invert the function $\textsc{Amp}^{f\restriction_\rho}$.

Lemma 6. *For any good ρ, there exists a function $\bar{M}_\rho : \{0,1\}^{\bar{m}} \to \{0,1\}^{\bar{n}}$ such that $\Pr_{\bar{x}\in\mathcal{U}_{\bar{n}},f\in\mathcal{F}}[\bar{M}_\rho$ inverts $\textsc{Amp}^{f\restriction_\rho}(\bar{x})] \geq 1 - o(\bar{\varepsilon})$.*

Proof. Fix any good ρ, and let $\bar{M}_\rho$ be the function which on input $\bar{y}$ outputs a random element in the set $\bar{A}_\rho^{-1}(\bar{y})$. Then $\Pr_{\bar{x},f}[\bar{M}_\rho$ fails to invert $\textsc{Amp}^{f\restriction_\rho}(\bar{x})]$ is

$$\begin{aligned}
&\Pr_{\bar{x},f}\left[\textsc{Amp}^{f\restriction_\rho}(\bar{M}_\rho(\textsc{Amp}^{f\restriction_\rho}(\bar{x}))) \neq \textsc{Amp}^{f\restriction_\rho}(\bar{x})\right]\\
&\leq \Pr_{\bar{x},f}\left[\textsc{Amp}^{f\restriction_\rho}(\bar{M}_\rho(\bar{A}_\rho(\bar{x}))) \neq \bar{A}_\rho(\bar{x})\right] + \Pr_{\bar{x},f}\left[\bar{A}_\rho(\bar{x}) \neq \textsc{Amp}^{f\restriction_\rho}(\bar{x})\right]\\
&< \sum_{\bar{y}} \Pr_{\bar{x}}\left[\bar{A}_\rho(\bar{x}) = \bar{y}\right] \cdot \Pr_{\bar{x},f}\left[\textsc{Amp}^{f\restriction_\rho}(\bar{M}_\rho(\bar{y}))) \neq \bar{y} \mid \bar{A}_\rho(\bar{x}) = \bar{y}\right] + o(\bar{\varepsilon})\\
&= \sum_{\bar{y}} \Pr_{\bar{x}}\left[\bar{A}_\rho(\bar{x}) = \bar{y}\right] \cdot \Pr_{\bar{x},\bar{x}',f}\left[\textsc{Amp}^{f\restriction_\rho}(\bar{x}') \neq \bar{y} \mid \bar{A}_\rho(\bar{x}) = \bar{A}_\rho(\bar{x}') = \bar{y}\right] + o(\bar{\varepsilon})\\
&= \sum_{\bar{y}} \Pr_{\bar{x}}\left[\bar{A}_\rho(\bar{x}) = \bar{y}\right] \cdot \Pr_{\bar{x}',f}\left[\textsc{Amp}^{f\restriction_\rho}(\bar{x}') \neq \bar{A}_\rho(\bar{x}') \mid \bar{A}_\rho(\bar{x}') = \bar{y}\right] + o(\bar{\varepsilon})\\
&= \Pr_{\bar{x},f}\left[\bar{A}_\rho(\bar{x}) \neq \textsc{Amp}^{f\restriction_\rho}(\bar{x})\right] + o(\bar{\varepsilon})\\
&= o(\bar{\varepsilon}).
\end{aligned}$$

□

From Lemma 6 and Definition 3, for any good ρ, a Markov's inequality implies that for most $f \in \mathcal{F}$, the function $M_\rho^{f\restriction_\rho} = \textsc{Dec}^{\bar{M}_\rho, f\restriction_\rho}$ can achieve $\Pr_x[M_\rho^{f\restriction_\rho}$ inverts $f\restriction_\rho(x)] > 1-\varepsilon$. This contradicts Lemma 5 since $\textsc{Dec}$ makes at most a polynomial number of queries to the oracle. Therefore, no such hardness amplification is possible, which proves Theorem 1.

4.2 Proof of Theorem 2

Let $\bar{\mathcal{H}}$ denote the family of hash functions from $\{0,1\}^m$ to $\{0,1\}^{3n}$ derived from a $(2, 2^{-3n})$-wise independent space. We will use the construction of [1], based on finite fields of characteristic two, with each function in the family specified by $O(n)$ bits. Then using ideas from [11, 10], given the specification of a function $h \in \bar{\mathcal{H}}$ and an input $x \in \{0,1\}^n$, one can compute $h(x)$ by an $\mathsf{AC}(d, 2^{O(n^{1/(d-1)})})$ circuit.

The key to the theorem is the following, which says that one can transform a hard function $f : \{0,1\}^n \to \{0,1\}^m$ with any $m \le \text{poly}(n)$ into a hard function $f' : \{0,1\}^{n'} \to \{0,1\}^{m'}$ with $n', m' = O(n)$.

Lemma 7. *A black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions can be realized in $\mathsf{AC}(d, 2^{O(n^{1/(d-1)})})$ with $\bar{\varepsilon} = \varepsilon - 2^{-n+1}$, $\bar{n} = O(n)$, and $\bar{m} = O(n)$.*

Proof. Given any ε-hard function $f : \{0,1\}^n \to \{0,1\}^m$, define the function $f' = \text{Amp}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ as

$$f'(x, h) = (h(f(x)), h),$$

with $x \in \{0,1\}^n$ and $h \in \bar{\mathcal{H}}$. Thus, $n' = n + O(n) = O(n)$ and $m' = 3n + O(n) = O(n)$. From the discussion at the beginning, Amp can be realized in $\mathsf{AC}(d, 2^{O(n^{1/(d-1)})})$.

Next, we prove the hardness of f' in a black-box way. Suppose M' is a function which inverts f' with probability more than $1 - (\varepsilon - 2^{-n+1})$. Consider the function $M = \text{Dec}^{M'}$, which on input $y \in \{0,1\}^m$ generates a random $h \in \bar{\mathcal{H}}$, calls $M'(h(y), h)$, and outputs the first component from the answer. We will show that M inverts f with probability more than $1 - \varepsilon$. Let M_h denote the function M with the random choice h. Call $h \in \bar{\mathcal{H}}$ *colliding* if there exist x, x' with $f(x) \neq f(x')$ and $h(f(x)) = h(f(x'))$. Then, $\Pr_{x \in \mathcal{U}_n}[M \text{ inverts } f(x)]$ is

$$\begin{aligned}
&\Pr_{x \in \mathcal{U}_n, h \in \bar{\mathcal{H}}}[f(M_h(f(x))) = f(x)] \\
&\ge \Pr_{x \in \mathcal{U}_n, h \in \bar{\mathcal{H}}}[h(f(M_h(f(x)))) = h(f(x)) \ \wedge \ h \text{ is not colliding}] \\
&\ge \Pr_{x \in \mathcal{U}_n, h \in \bar{\mathcal{H}}}[f'(M(f'(x, h))) = f'(x, h)] - \Pr_{h \in \bar{\mathcal{H}}}[h \text{ is colliding}] \\
&> 1 - (\varepsilon - 2^{-n+1}) - 2^{2n}(2^{-3n} + 2^{-3n}) \\
&= 1 - \varepsilon.
\end{aligned}$$

This proves the lemma. □

Consider any black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions realized by $\mathsf{AC}(d, s)$, with $\varepsilon \le \bar{\varepsilon} \cdot \gamma$. Assume we have $s \ge 2^{\Omega(n^{1/(d-1)})}$ and $\gamma \le o(n/(\bar{n} \log^{2d+1} s))$. Then by combining this with Lemma 7, we get a black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}', \bar{m}', \bar{\varepsilon}')$-hard functions realized by $\mathsf{AC}(2d, s')$, with $\bar{m}' = O(\bar{n})$, $s' = O(s)$, and $\varepsilon \le \bar{\varepsilon}' \cdot \gamma'$, for $\gamma' \le o(m/(\bar{m}' \log^{2d+1} s'))$, which contradicts Theorem 1. Therefore, no such hardness amplification can exist. Next, assume we have $s \le 2^{O(n^{1/(d-1)})}$ and $\gamma \le o(n/(\bar{n} \cdot n^{(2d+1)/(d-1)})$. Combining this with Lemma 7, we get a black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}', \bar{m}', \bar{\varepsilon}')$-hard functions realized by $\mathsf{AC}(2d, s')$, with $\bar{m}' = O(\bar{n})$, $s' \le 2^{O(n^{1/(d-1)})}$, and $\varepsilon \le \bar{\varepsilon}' \cdot \gamma'$, for $\gamma' \le o(m/(\bar{m}' n^{(2d+1)/(d-1)}) = o(m/(\bar{m}' \log^{2d+1} s'))$, which contradicts Theorem 1. Thus, no such hardness amplification can exist either. This completes the proof of Theorem 2.

5 Weakly Black-Box Hardness Amplification

In this section, we consider weakly black-box hardness amplifications from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \bar{\varepsilon})$-hard functions. Suppose such an amplification procedure, consisting of both the query-generation part and the answer-combination part, can be computed in AC^0. We will show that if it can amplify the hardness beyond a polynomial factor, then one can derive from it a highly-parallel one-way function. To simplify the presentation, we do not attempt to derive the strongest possible result here.

Theorem 3. *Suppose a weakly black-box hardness amplification from (n, m, ε)-hard functions to $(\bar{n}, \bar{m}, \sqrt{\bar{\varepsilon}})$-hard functions can be computed in AC^0 with $\varepsilon \le \bar{\varepsilon} \cdot \gamma$, for $\gamma < m/(\bar{m} \cdot \mathrm{poly}(\log n))$ and $\bar{\varepsilon} \ge 1/\mathrm{poly}(n)$. Then one can obtain from it a $(1 - o(1))\sqrt{\bar{\varepsilon}}$-OWF computable in NC^0.*

We will give the proof of Theorem 3 in Section 5.2. It will rely on a derandomized version of the random restriction $\mathcal{R}$ used in the previous section, which is discussed next.

5.1 Pseudo-Random Restriction

Set the parameters $\alpha = 2\varepsilon, \beta = (\log^2 s)/m, b = t^2/\varepsilon^3$ as in the previous section, and suppose $\varepsilon < \bar{\varepsilon} \cdot m/(\bar{m} \cdot \mathrm{poly}(\log n))$. Now we describe our choice of pseudo-random restriction $\bar{\rho} : \{0,1\}^n \to \{0,1,\star\}^m$. Again, we will first hash $\{0,1\}^n$ down to a smaller space $[b]$. Following [20], we would like to replace the random hash function by a pseudo-random one, but a more careful choice is needed. Here we use the family $\bar{\mathcal{H}}$ of hash functions in Lemma 1. Then we would like to replace the random restriction $\mathcal{R}^{b,m}_{\alpha,\beta}$ by a pseudo-random one, such that it is still good with high probability. For this, we need the following two constructions. (Recall from Section 3 that a random restriction from $\mathcal{R}^{b,m}_{\alpha,\beta}$ can be generated by a circuit $W^b \in \mathsf{AC}^0 : \{0,1\}^{b\ell} \to (\{0,1,\star\}^m)^b$ using a random seed of length $b\ell = b(m+1)\ell_0$.)

- Let $\textsc{Ind} : \{0,1\}^{r_1} \to \{0,1\}^{b\ell}$ be the generator defined as follows, with $r_1 = \mathrm{poly}(\log n)$. First, use the input as the seed for the generator in Fact 1 to produce b random variables over $\{0,1\}^{O(\ell_0 + \log m)}$ that are pairwise independent. Next, take each variable as the seed for the generator in Fact 1 to generate $m+1$ new random variables over $\{0,1\}^{\ell_0}$ that are 3-wise independent. The output of $\textsc{Ind}$ is the concatenation of these $b(m+1)$ new random variables over $\{0,1\}^{\ell_0}$.
- Let $\textsc{Nis} : \{0,1\}^{r_2} \to \{0,1\}^{b\ell}$ be Nisan's $o(\bar{\varepsilon})$-PRG for AC^0 circuits [17], with $r_2 = \mathrm{poly}(\log n)$.

Our pseudo-random restriction $\bar{\mathcal{R}}$ is the uniform distribution over the set of restrictions $\bar{\rho}_{h,z_1,z_2}$, with $(h, z_1, z_2) \in \{0,1\}^{r_0} \times \{0,1\}^{r_1} \times \{0,1\}^{r_2}$, defined as

$$\bar{\rho}_{h,z_1,z_2}(x) = W^b \left(\textsc{Ind}(z_1) \oplus \textsc{Nis}(z_2)\right)_{h(x)}.$$

Recall the definition of a good restriction from the previous section. The following says that such a pseudo-random restriction is still likely to be good.

Lemma 8. $\Pr_{\bar{\rho}\in\bar{\mathcal{R}}}[\bar{\rho}$ *is not good*$] = o(1)$.

Due to the space limitation, we defer the proof to the journal version. The idea is similar to that of Lemma 4. Now we use the generators Ind and Nis, respectively, to guarantee that the two conditions of being good also fail with a small probability.

5.2 Proof of Theorem 3

Suppose there exists such a weakly black-box hardness amplification with $\varepsilon < \bar{\varepsilon}\cdot m/(\bar{m}\cdot \mathrm{poly}(\log n))$ and $\bar{\varepsilon} \geq 1/\mathrm{poly}(n)$. We will show how to obtain from it a hard function. The idea is the following. From Section 4, we know that for most ρ and f the function $\textsc{Amp}^{f\restriction_\rho}$ is hard (to invert), but we do not know which ρ and f give a hard function. Our first step is to replace the random restriction ρ by a pseudo-random one $\bar{\rho}$ so that the function $\textsc{Amp}^{f\restriction_{\bar{\rho}}}$ is still likely to be hard. Then we show that by replacing the random function f by a pseudo-random one $\bar{f}$, the resulting function $\textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}$ is likely to be close to $\textsc{Amp}^{f\restriction_{\bar{\rho}}}$. However, having $\textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}$ close to a hard function $\textsc{Amp}^{f\restriction_{\bar{\rho}}}$ does not seem sufficient to guarantee that $\textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}$ is hard. The problem is that on input $\textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}(\bar{x}) = \textsc{Amp}^{f\restriction_{\bar{\rho}}}(\bar{x})$, an inverter might output $\bar{x}'$ such that $\textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}(\bar{x}) = \textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}(\bar{x}') \neq \textsc{Amp}^{f\restriction_{\bar{\rho}}}(\bar{x}')$. Thus, one might succeed in inverting $\textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}$ but not $\textsc{Amp}^{f\restriction_{\bar{\rho}}}$ for many such $\bar{x}$'s. We will come up with a carefully designed function that avoids this problem.

First, similar to Lemma 5, we have the following. We omit the proof here due to space limitation.

Lemma 9. *For any good $\bar{\rho}\in\bar{\mathcal{R}}$,* $\Pr_f[\textsc{Amp}^{f\restriction_{\bar{\rho}}}$ *is not* $\sqrt{\bar{\varepsilon}}$*-hard*$] = o(\bar{\varepsilon})$.

Next, we want to replace the random function by the following pseudo-random one. Let $\bar{\mathcal{F}}$ be the class of functions $\bar{f}_{h,z_3}$, with $h \in \bar{\mathcal{H}}$ and $z_3 \in \{0,1\}^{r_3}$, defined as

$$\bar{f}_{h,z_3}(x) = \textsc{Nis}'(z_3)_{h(x)},$$

where $\textsc{Nis}' : \{0,1\}^{r_3} \to (\{0,1\}^m)^b$ is Nisan's $o(\bar{\varepsilon})$-PRG for AC^0, with $r_3 = \mathrm{poly}(\log n)$. One can show that it has a similar effect as the random one in the sense that for any $\bar{x}\in\{0,1\}^{\bar{n}}$, $\bar{\rho}\in\bar{\mathcal{R}}$, and $\bar{y}\in\{0,1\}^{\bar{m}}$,

$$\left| \Pr_{f\in\mathcal{F}}\left[\textsc{Amp}^{f\restriction_{\bar{\rho}}}(\bar{x}) = \bar{y}\right] - \Pr_{\bar{f}\in\bar{\mathcal{F}}}\left[\textsc{Amp}^{\bar{f}\restriction_{\bar{\rho}}}(\bar{x}) = \bar{y}\right] \right| = o(\bar{\varepsilon}). \tag{1}$$

This is because $\textsc{Nis}'$ can fool such a test.

For any good $\bar{\rho}\in\bar{\mathcal{R}}$, we know by definition that there is a large subset $B \subseteq \{0,1\}^{\bar{n}}$ of inputs such that for each input in B, the output of Amp is the same for most $f\in\mathcal{F}$, and by (1), for most $\bar{f}\in\bar{\mathcal{F}}$. We would like our function to output this corresponding value for each input in B, and to output a value different from all these values for inputs not in B. We use $\bar{f}^p = (\bar{f}_1, \ldots, \bar{f}_p) \in \bar{\mathcal{F}}^p$, with $p = n^c$ for some large enough constant c, to locate one such set of inputs. Let $\textsc{Maj}_{\bar{\rho},\bar{f}^p}(\bar{x})$ be the majority value in $\{\textsc{Amp}^{\bar{f}_1\restriction_{\bar{\rho}}}(\bar{x}), \ldots, \textsc{Amp}^{\bar{f}_p\restriction_{\bar{\rho}}}(\bar{x})\}$. Let

$$B_{\bar{\rho},\bar{f}^p} = \left\{ \bar{x} \in \{0,1\}^{\bar{n}} : \Pr_{i\in[p]}\left[\textsc{Amp}^{\bar{f}_i\restriction_{\bar{\rho}}}(\bar{x}) \neq \textsc{Maj}_{\bar{\rho},\bar{f}^p}(\bar{x})\right] < \sqrt{\bar{\varepsilon}} \right\}.$$

Now for $\bar{\rho} \in \bar{\mathcal{R}}$, $\bar{f}^p \in \bar{\mathcal{F}}^p$, and $\bar{y} \in \{0,1\}^{\bar{m}}$, define the function $\bar{A}_{\bar{\rho},\bar{f}^p,\bar{y}} : \{0,1\}^{\bar{n}} \to \{0,1\}^{\bar{m}}$ as

$$\bar{A}_{\bar{\rho},\bar{f}^p,\bar{y}}(\bar{x}) = \begin{cases} \textsc{Maj}_{\bar{\rho},\bar{f}^p}(\bar{x}) & \text{if } \bar{x} \in B_{\bar{\rho},\bar{f}^p}, \\ \bar{y} & \text{otherwise.} \end{cases}$$

Call $(\bar{\rho}, \bar{f}^p, \bar{y}) \in \bar{\mathcal{R}} \times \bar{\mathcal{F}}^p \times \{0,1\}^{\bar{m}}$ nice if $\bar{\rho}$ is good and the following three conditions all hold:

(a) $|B_{\bar{\rho},\bar{f}^p}| \geq (1 - o(\sqrt{\bar{\varepsilon}}))2^{\bar{n}}$.
(b) For any $\bar{x} \in B_{\bar{\rho},\bar{f}^p}$, $\Pr_{f\in\mathcal{F}}\left[\bar{A}_{\bar{\rho},\bar{f}^p,\bar{y}}(\bar{x}) \neq \textsc{Amp}^{f\restriction_{\bar{\rho}}}(\bar{x})\right] = o(\bar{\varepsilon})$.
(c) For any $\bar{x} \notin B_{\bar{\rho},\bar{f}^p}$ and $\bar{x}' \in B_{\bar{\rho},\bar{f}^p}$, $\bar{A}_{\bar{\rho},\bar{f}^p,\bar{y}}(\bar{x}) \neq \bar{A}_{\bar{\rho},\bar{f}^p,\bar{y}}(\bar{x}')$.

The following lemma says that a randomly chosen $(\bar{\rho}, \bar{f}^p, \bar{y})$ is likely to be nice. Due to the space limitation, we omit the proof here.

Lemma 10. $\Pr_{\bar{\rho}\in\bar{\mathcal{R}},\bar{f}^p\in\bar{\mathcal{F}}^p,\bar{y}\in\mathcal{U}_{\bar{m}}}[(\bar{\rho}, \bar{f}^p, \bar{y})$ *is not nice*$] = o(1)$.

The following shows that a nice $(\bar{\rho}, \bar{f}^p, \bar{y})$ gives a hard function.

Lemma 11. *For any nice $(\bar{\rho}, \bar{f}^p, \bar{y})$, the function $\bar{A}_{\bar{\rho},\bar{f}^p,y}$ is $(1 - o(1))\sqrt{\bar{\varepsilon}}$-hard.*

Proof. Fix any nice $(\bar{\rho}, \bar{f}^p, \bar{y})$. Consider any polynomial-size circuit $\bar{M}$ which tries to invert $\bar{A}_{\bar{\rho},\bar{f}^p,y}$. For notational convenience, let us write $\hat{A}$ for $\bar{A}_{\bar{\rho},\bar{f}^p,\bar{y}}$, A^f for $\textsc{Amp}^{f\restriction_{\bar{\rho}}}$, and B for $B_{\bar{\rho},\bar{f}^p}$. Suppose we sample $\bar{x}$ uniformly from $\{0,1\}^{\bar{n}}$ and f uniformly from $\mathcal{F}$. Let E be the event that $\bar{M}$ inverts $\hat{A}(\bar{x})$. Clearly, E is the union of the two events $E_1 : (\bar{M}$ inverts $\hat{A}(\bar{x})) \wedge (\hat{A}(\bar{x}') = A^f(\bar{x}'))$ and $E_2 : (\bar{M}$ inverts $\hat{A}(\bar{x})) \wedge (\hat{A}(\bar{x}') \neq A^f(\bar{x}'))$, where $\bar{x}' = \bar{M}(\hat{A}(\bar{x})))$.

First, note that the event E_1 is contained in the union of the two events $E_{1,1} : \hat{A}(\bar{x}) \neq A^f(\bar{x})$ and $E_{1,2} : \bar{M}$ inverts $A^f(\bar{x})$. From items (a) and (b), we have $\Pr_{\bar{x},f}[E_{1,1}] \leq \Pr_{\bar{x}}[\bar{x} \notin B] + \Pr_{\bar{x},f}[\hat{A}(\bar{x}) \neq A^f(\bar{x}) \mid \bar{x} \in B] = o(\sqrt{\bar{\varepsilon}})$. Then by Lemma 9, $\Pr_{\bar{x},f}[E_{1,2}]$ is at most

$$\Pr_f\left[A^f \text{ is not } \sqrt{\bar{\varepsilon}}\text{-hard}\right] + \Pr_{\bar{x},f}\left[\bar{M} \text{ inverts } A^f(\bar{x}) \mid A^f \text{ is } \sqrt{\bar{\varepsilon}}\text{-hard}\right] \leq o(\bar{\varepsilon}) + 1 - \sqrt{\bar{\varepsilon}}.$$

Next, note that the event E_2 is contained in the union of the two events $E_{2,1} : \bar{x} \notin B$ and $E_{2,2} : (\bar{x} \in B) \wedge (\bar{M}$ inverts $\hat{A}(\bar{x})) \wedge (\hat{A}(\bar{x}') \neq A^f(\bar{x}'))$. From item (a), $\Pr_{\bar{x}}[E_{2,1}] = o(\sqrt{\bar{\varepsilon}})$. Observe that the event $E_{2,2}$ implies that $(\bar{x}' \in B) \wedge (\hat{A}(\bar{x}') \neq A^f(\bar{x}'))$, so by item (b), $\Pr_{\bar{x},f}[E_{2,2}] = o(\bar{\varepsilon})$.

Combining these bounds together, we get $\Pr_{\bar{x},f}[E] \leq 1 - \sqrt{\bar{\varepsilon}} + o(\sqrt{\bar{\varepsilon}})$, which proves the lemma.

□

Finally, define the function $\bar{A} : \{0,1\}^{\bar{n}} \times \bar{\mathcal{R}} \times \bar{\mathcal{F}}^p \times \{0,1\}^{\bar{m}} \to \{0,1\}^{\bar{m}} \times \bar{\mathcal{R}} \times \bar{\mathcal{F}}^p \times \{0,1\}^{\bar{m}}$ as

$$\bar{A}(\bar{x}, \bar{\rho}, \bar{f}^p, \bar{y}) = (\bar{A}_{\bar{\rho}, \bar{f}^p, \bar{y}}(\bar{x}), \bar{\rho}, \bar{f}^p, \bar{y}).$$

Note that the input length of $\bar{A}$ is at most $\mathrm{poly}(n)$, since each $\bar{\rho} \in \bar{\mathcal{R}}$ can be specified by $\mathrm{poly}(\log n)$ bits and each $\bar{f}^p \in \bar{\mathcal{F}}^p$ can be specified by $\mathrm{poly}(n)$ bits.

Lemma 12. *The function $\bar{A}$ is $(1 - o(1))\bar{\varepsilon}$-hard.*

Proof. Consider any polynomial-size circuit $\bar{M}$ which attempts to invert $\bar{A}$. Then $\Pr_{\bar{x}, \bar{\rho}, \bar{f}^p, \bar{y}}[M \text{ fails to invert } \bar{A}(\bar{x}, \bar{\rho}, \bar{f}^p, \bar{y})]$ is at least

$$\Pr_{\bar{\rho}, \bar{f}^p, \bar{y}}\left[(\bar{\rho}, \bar{f}^p, \bar{y}) \text{ nice}\right] \cdot \Pr_{\bar{x}, \bar{\rho}, \bar{f}^p, \bar{y}}\left[M \text{ fails to invert } \bar{A}(\bar{x}, \bar{\rho}, \bar{f}^p, \bar{y}) \mid (\bar{\rho}, \bar{f}^p, \bar{y}) \text{ nice}\right],$$

which by Lemma 10 & 11 is at least $(1 - o(1)) \cdot (1 - o(1))\bar{\varepsilon} = (1 - o(1))\bar{\varepsilon}$. $\square$

Since Nisan's PRG, the generator IND, and functions in $\bar{\mathcal{H}}$ all can be computed in NC^1, the function $\bar{A}$ can be computed in NC^1 too. From [2], this yields a OWF in NC^0, which proves the theorem.

6 Black-Box Construction of PRG from OWF

In this section, we study the complexity-quality tradeoff for black-box constructions of pseudo-random generators from strongly one-way functions. Our result is the following.

Theorem 4. *No black-box construction of $(\bar{n}, \bar{m}, 1/5)$-PRGs from $(n, m, 1 - n^{-\log n})$-hard functions can be realized by $\mathsf{AC}(d, s)$ with $\bar{m} > \bar{n}(1 + (\log^{d+5} s)/m)$ and $s \leq 2^{m^{o(1/d)}}$. In particular, with $d = O(1)$, such construction of PRG can only have a sublinear stretch unless $s \geq 2^{m^{\Omega(1)}}$.*

Proof. Assume for the sake of contradiction that such a black-box construction realized by $\mathsf{AC}(d, s)$ exists with $\bar{m} \geq \bar{n}(1 + (\log^{d+5} s)/m)$ and $s \leq 2^{m^{o(1/d)}}$. We will show that this leads to a contradiction. The idea is similar to that in Section 4. First, we will show that for a random restriction ρ and a random function f, the function $f\lceil_\rho$ is weakly hard, and the function derived from it using direct product is strongly hard. On the other hand, suppose we have such a PRG construction. Then we will show that a random restriction can reduce the effect of a random function, and consequently there exists a distinguisher which breaks the PRG. This can then be used to invert the strongly-hard function well, and we reach a contradiction.

Let PRG be the encoding procedure and DEC the decoding procedure. Let $k = c_0 \log^{d+3} s$ for a large enough constant c_0, let $n' = n/k$ and $m' = m/k$. Note that $n', m' \geq \mathrm{poly}(n)$ since $s \leq 2^{n^{o(1/d)}}$ and $k \leq n^{o(1)}$. Now we replace the parameters n and m in the previous sections by n' and m', and consider sampling function $f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ and restriction $\rho : \{0,1\}^{n'} \to \{0,1,\star\}^{m'}$. Set the parameters:

$$\alpha = 1/\log^{d+1} s,\ \beta = (\log^2 s)/m',\ \text{and } b = t^2 m'.$$

Similar to Lemma 5, one can show that the function $f\lceil_\rho$ is $\Omega(\alpha)$-hard with high probability (using an almost identical proof). If $f\lceil_\rho$ is $\Omega(\alpha)$-hard, the function $f_\rho^k : \{0,1\}^{kn'} \to \{0,1\}^{km'}$ defined as $f_\rho^k(x_1, \ldots, x_k) = (f\lceil_\rho(x_1), \ldots, f\lceil_\rho(x_k))$ is $(1 - n^{-\log n})$-hard, according to [21]. Thus we have the following.

Lemma 13. *For most $\rho \in \mathcal{R}$, for any oracle algorithm M_ρ making at most* poly(n) *oracle queries, for most $f \in \mathcal{F}$,* $\Pr_{x \in \mathcal{U}_n}[M_\rho^{f_\rho^k}$ *inverts* $f_\rho^k(x)] \leq n^{-\log n}$.

For $x, x' \in \{0,1\}^n$, let $\triangle(x, x') = |\{i \in [n] : x_i \neq x'_i\}|/n$, their relative Hamming distance. Then as in Section 4, one can show that the random restriction can reduce the effect of the random function on Prg.

Lemma 14. *For most $\rho \in \mathcal{R}$, there exists a function $\bar{G}_\rho : \{0,1\}^{\bar{n}} \to \{0,1\}^{\bar{m}}$ such that for most $f \in \mathcal{F}$,* $\mathrm{E}_{\bar{x}}[\triangle(\bar{G}_\rho(\bar{x}), \mathrm{Prg}^{f_\rho^k}(\bar{x})] = \mu$ *for some* $\mu = O(1/m')$.

Form such a function $\bar{G}_\rho$, one can construct a distinguisher $\bar{D}_\rho : \{0,1\}^{\bar{m}} \to \{0,1\}$ for $\mathrm{Prg}^{f_\rho^k}$, defined by $\bar{D}_\rho(\bar{y}) = 1$ if and only if there exists some $\bar{y}'$ in the image of $\bar{G}_\rho$ such that $\triangle(\bar{y}, \bar{y}') \leq 5\mu$. The we have the following, whose proof is omitted due to space limitation.

Lemma 15. *For most $\rho \in \mathcal{R}$, there exists a distinguisher $\bar{D}_\rho : \{0,1\}^{\bar{m}} \to \{0,1\}$ such that for most $f \in \mathcal{F}$, $\bar{D}_\rho$ can $1/5$-distinguish* $\mathrm{Prg}^{f_\rho^k}$.

According to the lemma, for most $\rho \in \mathcal{R}$ and $f \in \mathcal{F}$, the function $M_\rho = \mathrm{Dec}^{\bar{D}_\rho}$ achieves $\Pr_x[M_\rho^{f_\rho^k}$ inverts $f_\rho^k(x)] > n^{-\log n}$. This contradicts Lemma 13, since Dec makes at most a polynomial number of queries to the oracle. Thus we have the theorem. □

References

1. Noga Alon, László Babai, Johan Håstad, and Rene Peralta. Some constructions of alomost k-wise independent random variables. *Random Structures and Algorithms*, 3(3), pages 289–304, 1992.
2. Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography in NC^0. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 166–175, 2004.
3. Ravi B. Boppana. The average sensitivity of bounded-depth circuits. *Information Processing Letters*, 63(5), pages 257–261, 1997.
4. Giovanni Di Crescenzo and Russell Impagliazzo. Security-preserving hardness-amplification for any regular one-way function. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pages 169–178, 1999.

5. Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1), pages 13–27, 1984.
6. Rosario Gennaro and Luca Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 305–313, 2000.
7. Oded Goldreich, Russell Impagliazzo, Leonid A. Levin, Ramarathnam Venkatesan, and David Zuckerman. Security preserving amplification of hardness. In *Proceedings of the 31st Annual IEEE Symposium on Foundations of Computer Science*, pages 318–326, 1990.
8. Johan Håstad. *Computational limitations for small depth circuits*. PhD thesis, MIT Press, 1986.
9. Johan Håstad, Russel Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4), pages 1364–1396, 1999.
10. Alexander Healy and Emanuele Viola. Constant-depth circuits for arithmetic in finite fields of characteristic two. *Electronic Colloquium on Computational Complexity*, TR05-087, 2005.
11. William Hesse, Eric Allender, and David A. M. Barrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *Journal of Computer and System Sciences*, 65(4), pages 695–716, 2002.
12. Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 44–61, 1989.
13. Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, Fourier transform, and learnability. *Journal of the ACM*, 40(3), pages 607–620, 1993.
14. Henry Lin, Luca Trevisan, and Hoeteck Wee. On hardness amplification of one-way functions. In *Proceedings of the 2nd Theory of Cryptography Conference*, pages 34–49, 2005.
15. Chi-Jen Lu, Shi-Chun Tsai, and Hsin-Lung Wu. On the complexity of hardness amplification. In *Proceedings of the 20th Annual IEEE Conference on Computational Complexity*, pages 170–182, 2005.
16. Joseph Naor and Moni Naor. Small-bias probability spaces: efficient constructions and applications. *SIAM Journal on Computing*, 22(4), pages 838–856, 1993.
17. Noam Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica*, 11(1), pages 63–70, 1991.
18. Christos Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.
19. Emanuele Viola. The complexity of constructing pseudorandom generators from hard functions. *Computational Complexity*, 13(3-4), pages 147–188, 2005.
20. Emanuele Viola. On constructing parallel pseudorandom generators from one-way functions. In *Proceedings of the 20th Annual IEEE Conference on Computational Complexity*, pages 183–197, 2005
21. Andrew Chi-Chih Yao. Theory and applications of trapdoor functions. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, pages 80–91, 1982.