Input Locality and Hardness Amplification

Andrej Bogdanov[1,*] and Alon Rosen[2,**]

[1] Dept. of CSE and ITCSC, Chinese Univ. of Hong Kong
andrejb@cse.cuhk.edu.hk
[2] Efi Arazi School of Computer Science, IDC Herzliya
alon.rosen@idc.ac.il

Abstract. We establish new hardness amplification results for one-way functions in which each input bit influences only a small number of output bits (a.k.a. input-local functions). Our transformations differ from previous ones in that they approximately preserve input locality and at the same time retain the input size of the original function.

Let $f\colon \{0,1\}^n \to \{0,1\}^m$ be a one-way function with input locality d, and suppose that f cannot be inverted in time $\exp(\tilde{O}(\sqrt{n} \cdot d))$ on an ε-fraction of inputs. Our main results can be summarized as follows:

- If f is injective then it is equally hard to invert f on a $(1-\varepsilon)$-fraction of inputs.
- If f is regular then there is a function $g\colon \{0,1\}^n \to \{0,1\}^{m+O(n)}$ that is $d + O(\log^3 n)$ input local and is equally hard to invert on a $(1-\varepsilon)$-fraction of inputs.

A natural candidate for a function with small input locality and for which no sub-exponential time attacks are known is Goldreich's one-way function. To make our results applicable to this function, we prove that when its input locality is set to be $d = O(\log n)$ certain variants of the function are (almost) regular with high probability.

In some cases, our techniques are applicable even when the input locality is not small. We demonstrate this by extending our first main result to one-way functions of the "parity with noise" type.

Keywords: one-way function, input locality, hardness amplification, parity with noise.

1 Introduction

In this paper we are interested in amplifying the hardness of inverting a one-way function. Our goal is to do so without significantly deteriorating the function's parallel complexity and/or efficiency. To the best of our knowledge, these objectives are not simultaneously achieved by any of the previous methods for amplifying hardness.

[*] Research supported by RGC GRF grant 2150617.

[**] Research supported by BSF grant 2006317 and ISF grant 334/08.

Y. Ishai (Ed.): TCC 2011, LNCS 6597, pp. 1–18, 2011.

Our results assume the function is regular, and sub-exponentially hard to invert. They crucially rely on it being *input-local*, meaning that each input bit affects only a small number of output bits. Under these assumptions we show how to amplify hardness while preserving the function's input length and input locality. In some cases we achieve this without modifying the function altogether.

1.1 Hardness Amplification

The problem of hardness amplification can be described as follows: given a one-way function $f(x)$, construct a function, $g(y)$, so that if $f(x)$ is hard to invert on an ε fraction of inputs, then $g(y)$ is hard to invert on some $1-\delta > \varepsilon$ fraction of inputs. Amplification of hardness is established by exhibiting a reduction from the task of inverting f to the task of inverting g. The overall quality of the amplification is determined by: (1) the complexity of the construction (in particular, the relationship between $|x|$ and $|y|$), (2) the complexity of the reduction, and (3) the exact asymptotic relationship between ε and $1-\delta$.

The most basic method for amplifying hardness is due to Yao [16]. It consists of independently evaluating the function $f(x)$ many times in parallel. Using this transformation, it is essentially possible to obtain an arbitrary level of amplification. However, this comes at the cost of significantly blowing up the input size. For instance, if we wish to amplify from error $\varepsilon > 0$ to error $1-\delta > \varepsilon$, evaluating $g(y)$ will involve applying $f(x)$ to $O((1/\varepsilon)\log(1/\delta))$ small pieces of y, each of size $|x|$ (resulting in $|y| = O(|x| \cdot (1/\varepsilon)\log(1/\delta))$).

A better tradeoff between security and efficiency is achieved by Goldreich et al (GILVZ), for the special case of regular one-way functions [9]. In their construction, the evaluation of $g(y)$ consists of repeatedly applying f in sequence, where every two successive applications are interleaved with a randomly chosen step on an expander graph. The starting point of g's evaluation is an input x to f, and intermediate steps on the graph are determined by an auxiliary random string whose total length is $O((1/\varepsilon)\log(1/\delta))$. This results in $|y| = |x| + O((1/\varepsilon)\log(1/\delta))$, but renders the evaluation of $g(y)$ inherently sequential.

A related transformation was analyzed by Haitner et al (HHR), also for the case of regular functions [11,10]. Their transformation sequentially iterates the function with intermediate applications of a hash function, and has the advantage of not requiring knowledge of the regularity of f. Similarly to the GILVZ transformation, it is sequential in nature.

One last category of amplification results relies on *random self-reducibility*. It applies to functions that allow an efficient randomized mapping from $f(x)$ to $f(y)$, where y is a random value from which one can efficiently retrieve x. When satisfied, random self-reducibility enables very simple worst-case to average-case hardness amplification, without having to modify the original function. However, it is not known to be satisfied by one-way functions in general.

1.2 Highly Parallelizable One-Way Functions

Applebaum, Ishai and Kushilevitz (AIK) give strong evidence for the existence of one-way functions that can be evaluated in as little as constant parallel time.

They first present one-way functions with constant *output locality*, meaning that each output bit depends on a most a constant number of input bits [3]. These functions are constructed using *randomized encodings*, a tool that allows them to transform well known candidate one-way functions that have low (but not necessarily constant) parallel complexity into ones with constant output locality. They then go on and show that, in some specific cases, the functions resulting from their randomized encodings also satisfy constant input locality [4].

An alternative source for candidate one-way functions with small input and output locality is given by Goldreich [8]. These candidates are arguably more natural than the ones resulting from the AIK transformations. They also seem to offer a more attractive tradeoff between input length and security (as in many cases randomized encodings necessitate a significant blow up in the input size of the original function). Goldreich's constructions are quite general, and allow flexibility in the choice of the function, both in terms of the way in which inputs are connected to outputs, as well as in the choice of the predicates used to compute the function's output bits. To date, no sub-exponential time inversion algorithm is known for any variant of his functions.

Known hardness amplification methods are not well suited for functions of the above sort. Being inherently sequential, the GILVZ and HHR transformations do not preserve parallelism. Yao's transformation, on the other hand, does not increase parallel time, but it does incur a significant loss in efficiency (cf. Lin et al. [15]). This presents us with the challenge of coming up with efficient hardness amplification methods that are well suited for parallelizable functions. Our approach to the problem will be to utilize properties implied by the highly parallel structure of the function, and specifically small input-locality.

1.3 Main Results

Let $f\colon \{0,1\}^n \to \{0,1\}^m$ be a one-way function with input locality d, and suppose that f cannot be inverted in time $\exp(\tilde{O}(\sqrt{n}\cdot d))$ on an ε-fraction of inputs. Our first main result falls into the category of *self-amplification*, meaning that the hardness amplification does not require modifying the underlying function.

Theorem 1 (Self-amplification for injective functions): *Suppose that f is injective. Then, f cannot be inverted in time* $\exp(\tilde{O}(\sqrt{n}\cdot d))$ *on a $(1-\varepsilon)$-fraction of inputs.*

Based on the ideas used in the proof Theorem 1, we prove an analogous theorem for functions of the "parity with noise" type. Specifically, consider a family, $\{M_n\}$, of $m(n)\times n$ matrices with entries in $\{0,1\}$ and let $p \in [0,1]$ be a parameter. Define a function family $f_n\colon \{0,1\}^n \to \{0,1\}^m$ as $f_n(x,e) = M_n x + e \pmod 2$, where x is a vector chosen uniformly at random from $\{0,1\}^n$, and $e \in \{0,1\}^m$ is a vector of hamming weight at most $2pm$ chosen from the following distribution: Each entry of e is chosen independently from a p-biased distribution, conditioned on e having hamming weight at most $2pm$.

We assume that $\{f_n\}$ is one-way against randomized time $\exp(\tilde{O}(\sqrt{m}))$ on some ε fraction of inputs. We also require that the functions f_n are 1-1. This happens when M_n is a generator matrix of a code of minimum distance $4pm$. In such a case, the input locality of f_n will be as large as $\Omega(n)$. Nevertheless, we can prove the following analogue of Theorem 1.

Theorem 2 (Self-amplification for parity with noise): *Suppose that* $\{f_n\}$ *is injective. Then, (under appropriate constraints on parameters)* $\{f_n\}$ *cannot be inverted in randomized time* $\exp(\tilde{O}(\sqrt{m}))$ *on a* $(1-\varepsilon)$*-fraction of inputs.*

To make our results applicable to a wider class of functions, we also consider a generalization of Theorem 1 to the case where the function we wish to amplify is regular (every output has the same number of preimages). As before, we assume that the function $f\colon \{0,1\}^n \to \{0,1\}^m$ has input locality d, and that f cannot be inverted in time $\exp(\tilde{O}(\sqrt{n}\cdot d))$ on an ε-fraction of inputs. This time, however, we are not able to prove self-amplification and settle for some increase in output length and input locality, while still preserving input length.

Theorem 3 (Amplification for regular functions): *Suppose that* f *is regular. Then, there is a function* $g\colon \{0,1\}^n \to \{0,1\}^{m+O(n)}$ *that is* $d + O(\log^3 n)$ *input local and that cannot be inverted in time* $\exp(\tilde{O}(\sqrt{n}\cdot d))$ *on a* $(1-\varepsilon)$*-fraction of inputs.*

A natural candidate for a function with small input locality and for which no subexponential time attacks are known is Goldreich's one-way function [8]. Given a bipartite graph G with n vertices on the left, m vertices on the right, and regular right-degree d_{out} and given a predicate $P\colon \{0,1\}^{d_{\mathsf{out}}} \to \{0,1\}$, the function $f_{G,P}\colon \{0,1\}^n \to \{0,1\}^m$ is defined by setting the i^{th} bit of $f_{G,P}(x)$ to be equal to $P(x_{\Gamma(i,1)}, \ldots, x_{\Gamma(i,d_{\mathsf{out}})})$, where $\Gamma_{(i,j)}$ is the j^{th} neighbor of right vertex i of G. Goldreich proposed setting $m = n$ and considered d_{out} ranging from a constant to $O(\log n)$. He conjectured that when G is a good expander graph and P is randomly chosen, with high probability $f_{G,P}$ is one-way when n is sufficiently large.

We consider instantiations of Goldreich's functions with a certain class of balanced predicates, which we call d_{out}*-parity-blowup* predicates, and assume that G is chosen at random. Relying on Theorem 3 we can prove the following.

Theorem 5 (Amplification for Goldreich's function): *Suppose that for at least half the graphs* G*, the function* $f_{G,P}$ *is hard to invert on an* ε*-fraction of inputs for circuits of size* $\exp(\tilde{O}(\sqrt{n}))$*. Then there exists a function* $g\colon \{0,1\}^n \to \{0,1\}^{n+O(\log(1/\varepsilon))}$ *of circuit size* $O(n \log n)$ *that is hard to invert on a* $(1-\varepsilon)$*-fraction of inputs by circuits of size* $\exp(\tilde{O}(\sqrt{n}))$*.*

By observing that parity-blowup predicates can be represented by constant degree polynomials over $GF(2)$ we can apply the randomized encodings of AIK [3], and obtain a function with constant output locality and slightly longer input and output length.

Finally, we state a result that applies in the setting where d_{out} is constant and $m \geq Dn$, where $D = D(d_{\mathsf{out}})$ is a sufficiently large constant. Invoking a recent result of Bogdanov and Qiao [6], we prove that for any P and with high probability over the choice of G if $f_{G,P}$ is hard to invert on an ε fraction of inputs in time $\exp(\tilde{O}(\sqrt{n}))$, then $f_{G,P}$ is hard to invert on a $1-\varepsilon$ fraction of inputs in time $\exp(\tilde{O}(\sqrt{n}))$.

1.4 Applicability

Generally speaking, our results are not applicable to functions that are obtained via the randomized encodings of AIK. This is because these encodings typically incur at least a quadratic blow up in the input size. Thus, even if the original function is exponentially hard to invert, we cannot hope to prove that the resulting function is more than $\exp(O(\sqrt{n}))$ hard to invert (at least not based on the hardness of the original function).

It is conceivable that in some specific cases the randomized encodings can be performed in a way that does not significantly increase the input length of the original function. However, even if such cases exist, we are currently not aware of any natural candidate one-way function that would potentially satisfy Theorem 1's hypothesis. While AIK give several injective functions with constant output locality, none of these seems to have small input locality, and moreover they are all known to be invertible in time less than $\exp(\tilde{O}(\sqrt{n}))$ (e.g., ones that are based on the hardness of factoring and of finding discrete-logarithms). Other, presumably harder to invert, candidates are not known to be injective (though they may be regular, making Theorem 3 applicable).

Nevertheless, we feel that Theorem 1 is worth stating and proving. First of all, the fact that we could not think of any appropriate example does not mean that such does not exist. Secondly, the proof of the theorem contains the core ideas behind our reductions, and gives us the opportunity to present them without any irrelevant complications. Finally, and most importantly, using the main ideas of the theorem, we are able to prove an analogous result for functions of the "parity with noise" type, which are generally not known to be invertible in less than $\exp(O(n/\log n))$ time [5].

As we mentioned above, there is no known sub-exponential time algorithm that succeeds in inverting Goldreich's function on a non-negligible fraction of inputs. Applebaum, Barak, and Wigderson [2] prove that, when based on d-parity blowup predicates, the output of Goldreich's function is pseudorandom against linear functions, low-degree polynomials, and constant-depth circuits. In light of this, it currently seems reasonable to conjecture that no algorithm can invert such variants of the function on a small $\varepsilon = \varepsilon(n)$ fraction of inputs in time $\exp(\tilde{O}(\sqrt{n} \cdot d))$. Under this assumption, we obtain a function with polylogarithmic input locality and constant output locality that cannot be inverted by algorithms with comparable running time on a significantly larger, $(1-\varepsilon)$, fraction of inputs.

Even though not stated explicitly in Section 1.3, our reductions offer a concrete tradeoff between the running time of the reduction and the error $\varepsilon = \varepsilon(n)$ we are able to amplify from. The actual overhead incurred by the reduction is $\exp(O(\sqrt{n \cdot \log(1/\varepsilon)} \cdot d \cdot \log n))$. Thus, assuming that the original function is hard to invert in roughly this time, we can amplify starting from errors as small as say $\varepsilon(n) = 2^{-n^{O(1)}}$. Note that previous amplification methods are not applicable for such ranges of parameters, even if we assume sub-exponential hardness. This is because the input lengths of the functions resulting from their transformations grows proportionally to $\tilde{O}(1/\varepsilon)$.

1.5 Ideas and Techniques

Our self-amplification result is based on the following simple idea. Suppose f is a 1-1 function with input locality d and x and x' are two inputs that differ in one coordinate. Suppose we can invert $f(x)$. Then with a little bit more work we can invert $f(x')$: By input locality, $f(x)$ and $f(x')$ can differ in at most d coordinates. We change d coordinates of $f(x')$ until we find $f(x)$, recover x, and change x in one coordinate to recover x'.

By repeating this argument r times, we can invert $f(x')$ where x and x' are within distance r using $O(n^{dr})$ invocations to the original inverter. So if we can invert f at x, we can also invert f at any x' within distance r of x. Therefore, assuming f is easy to invert on some set that covers an ε-fraction of $\{0,1\}^n$, we can also invert f at any input within distance r of this set. By setting $r = O(\sqrt{n})$, we obtain Theorem 1, the self-amplification result for 1-1 functions.

Amplifying regular functions. The assumption that f is 1-1 is important in this argument. If f was not 1-1, the inverter could return some other preimage which is very far from x and therefore also far from x'. In Theorem 3 we show that if the function $f\colon \{0,1\}^n \to \{0,1\}^m$ is not 1-1 but regular (i.e. K-to-1 for some K), then there exists a new function $f'\colon \{0,1\}^n \to \{0,1\}^{m'}$, $m' = m+O(n)$ such that if f is hard on an small fraction of inputs, then f' is hard on almost all of its inputs.

The transformation from f to f' effectively isolates inputs by applying an appropriate hash function. Hashing is a standard way to reduce a regular function to a 1-1 function [13,12]. However, applying a pairwise-independent hash increases input locality by $\Omega(\log K)$ (see Section 5.1) and makes Theorem 1 inapplicable when K is large. In Claim 1 we describe a new construction of a hash function which increases input locality only by $O((\log n)^3)$ and maps most preimages of f to unique values. Combining this hash with Theorem 1, we obtain Theorem 3, our amplification result for regular input-local functions.

Parity with noise. In Section 4 we apply our ideas to show self-amplification for functions of the parity with noise type. Although these functions do not have low-input locality, we are able to apply our techniques. The reason is that these functions consists of two parts: A linear component, which is randomly self reducible, and the noise component, which is input-local. By combining an application of Theorem 1 to the noise component with a random self-reduction on the linear component, we prove Theorem 2.

Goldreich's function. As we explain in Section 6, Goldreich's function is unlikely to be 1-1 (except in special cases which are easy to invert), so Theorem 1 does not apply directly. However, we show that when m/n is a sufficiently large constant, if $f(x_1) = f(x_2)$, then x_1 and x_2 must be substantially correlated. Assuming f can be inverted on an ε-fraction of inputs, using our self-reduction from Theorem 1, for most x' we can invert $f(x)$ at some x that is close to x'. The inverse we obtain may not be equal to x, but it will be substantially correlated to x'. Using a result of Bogdanov and Qiao [6], we then recover an inverse for $f(x')$.

Our second application concerns functions $f\colon \{0,1\}^n \to \{0,1\}^m$ where $m = n$, but the degree is $O(\log n)$ and the predicate that f is based on is a "parity blowup" predicate (see Section 6). First we prove that such functions are likely to be at most K-to-1 for some constant K. Using the hash from Claim 1, we obtain a new function f' that is almost 1-1 and is almost as hard to invert. Finally, using the randomized encodings of Applebaum et al. [3], we can transform f' into a function with constant *output* locality at a polylogarithmic cost in the input and output length.

1.6 Open Questions

We believe it is interesting to investigate if our methods apply to a wider class of candidate one-way functions. In Section 6 we show that our amplification methods apply to variants of Goldreich's function where either (1) the degree is constant but the output to input length ratio is sufficiently large, or (2) the function is length-preserving, but the degree is logarithmic (so the function is not output-local) and the predicate is of a special form.

It would be interesting to investigate the range of parameters where the function is length-preserving and the degree is constant. We conjecture that when the predicate is balanced, such functions are "almost 2^{cn}-to-1" for some constant c, in the sense that for most x, $f(x)$ has $2^{cn \pm o(n)}$ preimages. If this was the case, we could apply Theorem 3 (and Corollary 1) to obtain very hard to invert functions with better locality parameters.

1.7 Paper Organization

Section 2 contains the most basic definitions relating to input locality, output locality, and regularity. The proof of Theorem 1, which holds the key ideas for our results, as well as the proof of Theorem 3, which deals with the regular case and involves the construction of a new input local hash function, are included in the main body of the paper. Other results are stated in corresponding sections. Due to lack of space, their proofs are deferred to the full version.

2 Definitions

Let $f\colon \{0,1\}^n \to \{0,1\}^m$ be a function. We say that the ith output $f(x)_i$ *depends* on the jth input x_j if there exists a setting of the inputs $x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n$ such that $f(x_1, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_n)_i \neq f(x_1, \ldots, x_{j-1}, 1, x_{j+1}, \ldots, x_n)_i$. We define the *degree* of the jth input to be the number of outputs that depend on the jth input. We say f has *input locality* d if the degree of every input is at most d. We define the *degree* of an output as the number of inputs it depends on and the *output locality* as the maximum degree of an output.

We say that f is *K-to-1* if for every $x \in \{0,1\}^n$, there exist exactly K inputs $x' \in \{0,1\}^n$ such that $f(x') = f(x)$. We say f is *regular* if it is K-to-1 for some K. We say f is *at most K-to-1* (resp., *at least K-to-1*) if for every x there are at most K (resp., at least K) x' such that $f(x') = f(x)$. We say f is *ε-close to K-to-1* if for at least a $(1-\varepsilon)$ fraction of the inputs $x \in \{0,1\}^n$, there exist exactly K inputs $x' \in \{0,1\}^n$ such that $f(x') = f(x)$.

In this work we consider both uniform and non-uniform constructions of one-way functions. The security of such functions can be defined against deterministic, randomized, and non-uniform inverters. We do not attempt to state our results in the most general setting. Instead, we use the definition that is most natural for the proof, in order to avoid distracting technical issues. For our purposes, it will be sufficient to define non-uniform one-way functions against non-uniform adversaries and uniform one-way functions against uniform (possibly randomized) adversaries.

In the non-uniform setting, we say $f\colon \{0,1\}^n \to \{0,1\}^m$ is *hard against circuits of size s on an α-fraction of inputs* if for every circuit C of size at most s, $f(C(f(x))) = f(x)$ for at most $\alpha \cdot 2^n$ inputs $x \in \{0,1\}^n$.

In the uniform setting, a function family $f = \{f_n\colon \{0,1\}^n \to \{0,1\}^{m(n)}\}$ is *one-way against (randomized) time $t(n)$ on an $\alpha(n)$-fraction of inputs* if (1) f is computable in deterministic polynomial time and (2) for every (randomized) algorithm A that runs in time $t(n)$ and every sufficiently large n, $f_n(A(1^n, f_n(x))) = f_n(x)$ for at most an $\alpha(n) \cdot 2^n$ fraction of inputs $x \in \{0,1\}^n$ (and with probability at most $1/2$ over the coin tosses of A). (To simplify notation, we will omit the length parameter 1^n as an input to the inverter in our proofs.)

3 Self-amplification for 1-1 Functions

Let $f\colon \{0,1\}^n \to \{0,1\}^m$ be any 1-1 function, and let d_j be the degree of the jth input. Set $\Delta = \sum_{j=1}^n d_j^2$.

Theorem 1. *Let $f = \{f_n\colon \{0,1\}^n \to \{0,1\}^{m(n)}\}$ be a 1-1 function family. Suppose f is one-way against time* $\exp(O(\sqrt{r\Delta}\log n))$ *on a e^{-r}-fraction of inputs ($r = r(n)$). Then f is one-way against time* $\exp(O(\sqrt{r\Delta}\log n))$ *on a* $(1-e^{-r})$-*fraction of inputs.*[1]

When f is a 1-1 function with input locality d, we get that if f is one-way against time $\exp(O(\sqrt{rn} \cdot d\log n))$ for a e^{-r}-fraction of inputs, then the same function is also one-way for a $(1-e^{-r})$-fraction of inputs.

[1] Usually, hardness amplification results are stated in terms of two parameters, the initial hardness ε and the "derived hardness" $(1-\delta)$. Since the complexity of our inverter is dictated by the minimum of ε and δ, without loss of generality we state our results for the special case $\varepsilon = \delta = e^{-r}$.

The proof is based on the following idea. For simplicity let us consider the case where the degree of every input is at most d. Assume that f can be inverted in time $\exp(O(\sqrt{r\varDelta}\log n))$ on an e^{-r}-fraction of inputs and let S' be the set of inputs on which this inversion algorithm succeeds. Let us consider all inputs x that are within hamming distance $\sqrt{2rn}$ from S'. By a standard probabilistic argument (Lemma 1, based on Theorem 7.5.3 in [1]) it follows that at least $1-e^{-r}$ fraction of inputs x have this property. Now if x and $x' \in S'$ differ in at most $\sqrt{2rn}$ coordinates, then $y = f(x)$ and $y' = f(x')$ will differ in at most $\sqrt{2rnd}$ coordinates. Therefore we can invert f at $y = f(x)$ by flipping the given set of $\sqrt{2rnd}$ coordinates on which y and y' differ, inverting f at y' to obtain x', and then moving back from x' to x by changing at most $\sqrt{2rn}$ coordinates.

We first state and prove the probabilistic inequality which is the technical heart of our argument. We prove the inequality in slightly more general form than is needed to prove Theorem 1 for later applications.

Lemma 1. *Consider the space $\{0,1\}^n$ with the p-biased distribution (i.e., each coordinate takes value 1 independently at random with probability p) for some $p \in [0,1]$. Let $X \subseteq \{0,1\}^n$ be any set of measure e^{-r} and let $d_1, \ldots, d_n$ be positive numbers. Let*

$$Z = \Big\{z\colon \textstyle\sum_{j\in[n]\colon x_j\neq z_j} d_j \leq \sqrt{2r\varDelta} \textit{ for some } x \textit{ in } X\Big\}.$$

where $\varDelta = \sum_{i=1}^n d_i^2$. Then Z has measure at least $1-e^{-r}$.

Proof. Define $d(z) = \min_{x\in S}\sum_{j\in[n]\colon x_j\neq z_j} d_j$. Then any change in z_j changes $d(z)$ by at most d_j. By Azuma's inequality, we have

$$\Pr[d(z) \leq \mathrm{E}[d(z)] - t] < e^{-2t^2/\varDelta} \quad \text{and} \quad \Pr[d(z) \geq \mathrm{E}[d(z)] + t] < e^{-2t^2/\varDelta}$$

Setting $t = \mathrm{E}[d(z)]$, from the first inequality we get $e^{-2t^2/\varDelta} > e^{-r}$, and therefore $t < \sqrt{r\varDelta/2}$. From the second one, $\Pr[z \notin Z] = \Pr[d(z) \geq \sqrt{2r\varDelta}] < e^{-r}$. □

Alternatively, Lemma 1 follows from a simple application of Talagrand's inequality.

Proof (of Theorem 1). Let $\varepsilon = e^{-r}$. We prove the contrapositive. Assume A inverts f_n on an ε-fraction of inputs in time $\exp(O(\sqrt{r\varDelta}\log m))$. We construct an algorithm B that inverts f_n on a $(1-\varepsilon)$-fraction of inputs as follows: On input y, perform the following procedure: For any set of at most $\sqrt{2r\varDelta}$ coordinates of $[m]$, flip the value of y in these coordinates to obtain y', compute $x' = A(y')$, then flip any set of $\sqrt{2r\varDelta}$ coordinates of x' to obtain x. If $f_n(x) = y$, output x. The running time of B is

$$\binom{m}{\sqrt{2r\varDelta}} \cdot (\text{running time of } A) \cdot \binom{n}{\sqrt{2r\varDelta}} \cdot (\text{eval. time of } f_n) = \exp(O(\sqrt{r\varDelta}\log n)).$$

We now argue that B inverts f on a $(1-\varepsilon)$-fraction of inputs. Let S' be the set of those x' such that $A(f(x')) = x'$. For each $j \in [n]$, let d_j denote the degree of the jth input. Now let

$$S = \Big\{x\colon \textstyle\sum_{j\in[n]\colon x_j\neq x'_j} d_j \leq \sqrt{2r\varDelta} \text{ for some } x' \text{ in } S'\Big\}.$$

If x' is in S' and x is its closest element in S, then $f(x)$ and $f(x')$ differ in at most $\sqrt{2r\Delta}$ coordinates. Moreover, x and x' can also differ in at most this many coordinates. It follows that if x is in S, then B successfully inverts $f(x')$. By Lemma 1, S contains at least a $1 - \varepsilon$ fraction of inputs. □

Remark 1. The proof of Theorem 1 easily generalizes to function families that are $e^{-r}/2$-close to 1-1. A non-uniform version, where "running time" is replaced by "circuit size", is also straightforward. We will use these extensions in our applications in Sections 5 and 6.

Theorem 1 gives a non-trivial result only when the sum of the squares of the input degrees D is at most $o(n^2/\log n)$. This assumption could be violated even if there is a single input of f whose degree is $\Omega(n)$. It is natural to ask if the self-amplification argument could be modified so as to allow for a small number of inputs that have unusually large degree.

We argue that this is unlikely to be the case: In the full version of the paper, we give an example showing that if non-trivial self-amplification can be achieved for functions where all but one of their inputs have degree at most $d + 1$, then every function of input locality d has a non-trivial inversion algorithm.

4 Linear Functions with Noise

We now state a self-amplification result for functions of the "parity with noise" type. We consider the following type of function. Let $\{M_n\}$ be a family of $m(n)$ by n matrices with entries in $\{0, 1\}$ and $p \in [0, 1]$ be a parameter. We define the function family $f_n \colon \{0, 1\}^n \to \{0, 1\}^{m(n)}$ as follows:

$$f_n(x, e) = M_n x + e$$

where x is a vector chosen uniformly at random from $\{0, 1\}^n$, and $e \in \{0, 1\}^m$ is a vector of hamming weight at most $2pm$ chosen from the following distribution: Each entry of e is chosen independently from a p-biased distribution, conditioned on r having hamming weight at most $2pm$. The matrix multiplication and vector addition are performed modulo two.

We will consider functions f_n that are 1-1. This happens when M_n is a generator matrix of a code of minimum distance $4pm$. In such a case, the input locality of f_n will be as large as $\Omega(n)$. Nevertheless, we can prove an analogue of Theorem 1 in this setting. One difference is that our self-amplification argument here is randomized, so we require that the function family is hard to invert even for randomized adversaries.

Theorem 2. *Suppose the function family $\{f_n \colon f_n(x, e) = M_n x + e\}$ is 1-1 and one-way against randomized time $\exp(O(\sqrt{rm}\log m))$ on a e^{-r} fraction of inputs. Assume $r < pm/10$. Then $\{f_n\}$ is one-way against randomized time $\exp(O(\sqrt{rm}\log m))$ on a $1 - e^{-r}$ fraction of inputs.*

The proof of Theorem 2 is given in the full version of this paper.

5 Hardness Amplification for Regular Functions

Theorem 1 shows how to achieve self-amplification for functions with small input locality that are 1-1. The assumption that the function is 1-1 was crucial in the argument for the following reason. Suppose f is a 1-1 function with input locality d and x and x' are two inputs that differ in exactly one coordinate. Suppose we can invert $f(x)$. Then with a little bit more work we can invert $f(x')$: Since $f(x)$ and $f(x')$ can differ in at most d coordinates, we change d coordinates of $f(x')$ until we find $f(x)$, recover x, and move back from x to x'.

An important point in this argument is that because f is 1-1, the inversion algorithm is guaranteed to return x and not some other preimage for $f(x)$. If f were not 1-1, the inverter could return some other preimage which is very far from x and therefore also far from x'. So in general we do not know how to achieve self-amplification for input-local functions that are not 1-1.

We now argue that if $f\colon \{0,1\}^n \to \{0,1\}^m$ is not 1-1 but regular, then there exists a new function $f'\colon \{0,1\}^n \to \{0,1\}^{m'}$, $m' = m + O(n)$ such that if f is hard on an small fraction of inputs, then f' is hard on almost all of its inputs. Moreover, the input locality of f' is not much larger than the input locality of f.

To simplify notation, let $\alpha(d, r, \log n) = (d + r + (\log n)^3) \cdot (\log n)$.

Theorem 3. *Suppose there exists a K-to-1 function $f\colon \{0,1\}^n \to \{0,1\}^m$ with input locality d which is hard against circuits of size $\exp(O(\sqrt{rn} \cdot \alpha(d, r, \log n))$ on a e^{-r}-fraction of inputs. Then there exists $f'\colon \{0,1\}^n \to \{0,1\}^{m+\log_2 K+O(r)}$, with input locality $d + O(r + (\log n)^3)$ which is hard against circuits of size $\exp(O(\sqrt{rn} \cdot \alpha(d, r, \log n)))$ on a $(1 - e^{-r})$-fraction of inputs. Moreover, if f is computable by a circuit of size s, then f' is computable by a circuit of size $s + O(n(\log n)^3)$.*

The construction of f' from f is non-uniform. In fact, our proof provides a randomized construction but for simplicity we present the argument in the non-uniform setting. We follow the standard approach of turning a general function into an almost 1-1 function via hashing [13,12]. The function f' will have the form $f'(x) = (f(x), h(x))$, where h is a suitably chosen hash function that does not increase input locality by much. If f is regular, then f' will be almost 1-1 in the sense that for most x, $f(x)$ has a unique preimage. Moreover, if f has input locality d, then f' will have input locality $d + O(r + (\log n)^3)$. We then amplify the hardness of f using Theorem 1 (and Remark 1).

Theorem 3 can be combined with the randomized encodings of Applebaum et al. [3,4] to obtain a hardness amplification result that preserves *output locality,* at the expense of increasing the input length by logarithmic factors.

Corollary 1. *Suppose there exists a regular function $f\colon \{0,1\}^n \to \{0,1\}^m$ with input locality d_{in} and output locality $d_{\mathsf{out}} \geq 3$ that is hard against circuits of size $\exp(O(\sqrt{rn}\cdot\alpha(d_{\mathsf{in}}, r, \log n)))$ on a e^{-r}-fraction of inputs. Then there is a function $f'\colon \{0,1\}^{n'} \to \{0,1\}^{m'}$, where $n' = O(n(\log n)^3)$ and $m' = m+O(n(\log n)^3)$ with output locality d_{out} that is hard against circuits of size $\exp(O(\sqrt{rn}\cdot\alpha(d_{\mathsf{in}}, r, \log n)))$ on a $(1 - e^{-r})$-fraction of inputs. If f is computable by a circuit of size s, then f' is computable by a circuit of size $s + O(n(\log n)^3)$.*

5.1 A Hash with Small Input Locality

A standard way to reduce a K-to-1 one-way function to a 1-1 one-way function is by hashing. Namely, we would like to define $f'(x) = (f(x), h(x))$, where $h\colon \{0,1\}^n \to \{0,1\}^{\log_2 K+O(1)}$ is a pairwise independent hash function. However, known constructions of pairwise independent hash functions have input locality as large as $\Omega(\log_2 K)$. This is in fact necessary: Mansour et al [14] showed that pairwise independent hash functions have *average sensitivity* $\Omega(n)$. By averaging, it follows that the input locality of such functions must be $\Omega(\log_2 K)$.

We need to construct a function f' from f which preserves not only the hardness of f but also its small input locality. Our function f' will also have the form $f'(x) = (f(x), h(x))$, where h is a suitably chosen hash function. However, our hash function h will only be approximately pairwise independent, chosen in a manner to have small input locality.

We note that Appelbaum et al. [4] (Appendix C in the journal version) give a different construction of an "almost pairwise-independent" hash function. However, the almost pairwise independence property they establish for their construction, while sufficient for their application, appears too weak to derive Claim 1.

Claim 1. *Suppose $f\colon \{0,1\}^n \to \{0,1\}^m$ is at most K-to-1, where $2^{k-1} \le K < 2^k$. Then there exists a function $h\colon \{0,1\}^n \to \{0,1\}^{k+3r+3}$ such that the function $f'(x) = (f(x), h(x))$ is $e^{-r}/2$-close to 1-1. Moreover, h is a linear function over $\{0,1\}^n$ with input locality $O(r) + \min\{k, O((\log n)^3)\}$.*

We now prove Claim 1. The construction of h will be probabilistic.

Construction of h. Assume that f is at most K-to-1, where $2^{k-1} \le K < 2^k$. The function h has the form $h(x) = (h_a(x), h_b(x))$ where

$$h_a(x) = (a_k \cdot x + a'_k, a_{k-1} \cdot x + a'_{k-1}, \dots, a_{k_0+1} \cdot x + a'_{k_0+1})$$
$$h_b(x) = (b_1 \cdot x + b'_1, b_2 \cdot x + b'_2, \dots, b_{3r+k_0+3} \cdot x + b'_{3r+k_0+3}).$$

and $k_0 = \min\{8(\log n)^2, k\}$. (In particular, if $k < 8(\log n)^2$, h only consists of the h_b part.)

To generate a random h, we choose the vectors $a_i, b_i \in \{0,1\}^n$ from the following distributions: Each a_i is chosen independently at random from the p_i-biased distribution over $\{0,1\}^n$, where $p_i = 4(\log n)^2/i < 1/2$. Each b_i is chosen independently at random from the uniform distribution over $\{0,1\}^n$, and a'_i, b'_i are uniformly random bits.

We now argue that if f is regular, then with probability at least $1/2$ over the choice of h, f' is regular over all but an $e^{-r}/2$ fraction of its inputs.

The proof will have two stages. In the first stage, we argue that for all but an $e^{-r}/8$ fraction of inputs x, there are at most 2^{r+k_0} inputs x' such that $(f(x), h_a(x)) = (f(x'), h_a(x'))$. In the second stage, we finish the proof by showing that h_b hashes all but an $e^{-r}/8$ fraction of those xs uniquely.

The first stage. If $k_0 = k$, the conclusion is trivial, so let us assume that $k_0 < k$. Let us fix an input x and let $S = \{x' \colon f(x) = f(x')\}$. Without loss of generality, we may assume that $2^{k-1} \le |S| < 2^k$. (If $|S|$ is smaller, we disregard the effect of the first few hashes in h_a.) We consider the following sequence of random sets defined recursively: $S_k = S$, $T_i = \{x' \in S_i \colon a_i \cdot x' = a_i \cdot x)\}$ and

$$S_{i-1} = \begin{cases} T_i, & \text{if } (1-1/n)|S_i|/2 \le |T_{i-1}| \le (1+1/n)|S_i|/2 \\ S_i, & \text{otherwise.} \end{cases}$$

Here is the intuition for this definition: We want to think of the ith hash a_i as "successful" if it decreases the size of siblings of x by roughly a factor of two (not much more and not much less). If all but r of the hashes are successful, then the size of S_0 can not be much more than 2^r, and so x will not have more than 2^r siblings that map to $(f(x), h_a(x))$. It is sufficient to show that the probability that more than r of the hashes fail to be successful is quite small.

Notice that by definition of the sets S_i, it must be that $S_i \ge \prod_{j=i}^{k}(1-1/n) \cdot 2^{i-1} \ge 2^{i-2}$. So we are left with the following question: Given a set S of size at least 2^{i-2}, how likely is it to be split successfully at stage i?

Lemma 2. *Assume $|R| \ge 2^{i-2}$. Let $a \sim \{0,1\}_p^n, b \sim \{0,1\}_{1/2}$, where $p = 4(\log n)^2/i < 1/2$. Then for n sufficiently large and any $\varepsilon > 0$,*

$$\Pr\big[\#\{y \in R \colon a \cdot y + b = 0\} \notin (1 \pm \varepsilon)|R|/2\big] \le \frac{1}{n^4 \varepsilon^2}.$$

Applying this lemma with $R = S_i$ and $\varepsilon = 1/n$, we have that each hash is successful with probability at least $1 - 1/n^2$, and the events are independent of one another. By a union bound, the probability of having more than r unsuccessful splits is at most $\binom{n}{r} \cdot (1/n^2)^r \le n^{-r} < e^{-r}/8$. So for any $x \in \{0,1\}^n$,

$$\Pr\big[|S_{k_0}| \ge 2^{k_0+r}\big] \le e^{-r}/8.$$

Proof. Let $X = \sum_{y \in R}(-1)^{a \cdot y + b}$. Then $\mathrm{E}[X] = 0$ and

$$\begin{aligned} \mathrm{E}[X^2] &= \sum_{y,z \in R} \mathrm{E}[(-1)^{a\cdot(y+z)}] \\ &\le |R| \max_z \sum_{y \in R} \mathrm{E}[(-1)^{a\cdot(y+z)}] \\ &= |R| \max_z \sum_{y \in R_z} \mathrm{E}[(-1)^{a \cdot y}] \\ &= |R| \max_z \sum_{y \in R_z} (1-2p)^{|y|} \end{aligned}$$

where $R_z = \{y + z \colon y \in T\}$, and $|y|$ denotes the hamming weight of y. Notice that the summation is maximized when R_z is a threshold set T – the set of all strings of hamming weight up to $k-1$ and possibly some of hamming weight k. Then we have

$$\mathrm{E}[X^2] \le |R| \cdot \sum_{y \in T} (1-2p)^{|y|} \le |R| \cdot \sum_{w=0}^{k} \binom{n}{w} \cdot (1-2p)^w.$$

We look at two cases: If $i \ge n$, then

$$\mathrm{E}[X^2] \le |R| \sum_{w=0}^{n} \binom{n}{w} \cdot (1-2p)^w = |R| \cdot 2^n \cdot (1-p)^n \le 4|R|^2 \cdot e^{-pn} \le |R|^2/n^4,$$

for n sufficiently large. If $i < n$, the ratio of consecutive terms in the summation is

$$\binom{n}{w+1}(1-2p)^{w+1} \Big/ \binom{n}{w}(1-2p)^w = (1-2p) \cdot \frac{n-w}{w+1} > 1$$

for every $w \le k$, and so

$$\mathrm{E}[X^2] \le |R| \cdot k \cdot \binom{n}{k} \cdot (1-2p)^k \le |R|^2 \cdot n \cdot (1-2p)^k \le |R|^2 \cdot n \cdot e^{-2pk}.$$

Since $k \ge i/\log n$, we get that $\mathrm{E}[X^2] \le |R|^2/n^4$ in this case also. By Chebyshev's inequality, it follows that

$$\Pr\big[\#\{y \in R\colon a \cdot y + b = 0\} \notin (1 \pm \varepsilon)|R|/2\big] = \Pr\big[|X| > \varepsilon|R|\big] \le 1/n^4\varepsilon^2. \qquad \square$$

The second stage and conclusion. Now fix an x such that $|S_{k_0}| < 2^{k_0+r}$. We now argue that by the end of the second stage, x is very likely to have a unique hash:

$$\Pr[\exists x' \in S_{k_0} - \{x\}\colon h(x') = h(x)] \le \sum_{x' \in S_{k_0} - \{x\}} \Pr[h(x') = h(x)] < e^{-r}/8.$$

Putting the analysis of both stages together, it follows that by the end of stage 2, for any specific x,

$$\Pr_h[\exists x'\colon f'(x') = f'(x)] \le e^{-r}/4.$$

Averaging over x and applying Markov's inequality, we get that for at least half the functions h,

$$\Pr_x[\exists x'\colon f'(x') = f'(x)] \le e^{-r}/2.$$

Now let us calculate the locality of a typical function h. For any fixed input bit, say x_1, let Y_a and Y_b be the number of occurrences of x_1 in h_a and h_b respectively. Then $\mathrm{E}[Y_a] = \sum_{i=k_0}^{k} 4(\log n)^2/i \le 4(\log n)^3$ and $\mathrm{E}[Y_b] = (3r + k_0 + 3)/2$, so $\mathrm{E}[Y_a + Y_b] = O((\log n)^3 + r)$. By Chernoff bounds and a union bound, we get that with probability at least 3/4, no input bit has more than $O((\log n)^3 + r)$ occurrences in h.

Therefore, there exists a hash function h that has input locality $O((\log n)^3 + r)$ and such that f' is 1-1 on all but $e^{-r}/2$ fraction of its inputs.

Remark 2. The conclusion of Claim 1 also holds under the weaker assumption that the function h is $e^{-r}/4$-close to at most K-to-1. We will use this generalization in Section 6.

5.2 Proof of Theorem 3

We now prove Theorem 3. To do so, first we show that the transformation from f to f' is hardness-preserving in the following sense: If f is hard to invert on an e^{-r}-fraction of inputs, then f' is hard to invert on an $\Omega(e^{-r})$-fraction of inputs. Since f' is almost 1-1, we can apply self-amplification to conclude that f' is in fact hard on a $1 - e^{-r}$ fraction of inputs.

Claim 2. *Assume $f\colon \{0,1\}^n \to \{0,1\}^m$ is K-to-1 where $2^{k-1} \leq K < 2^k$. Let f' and h be as in Claim 1. Assume that f' can be inverted on an $(1 - e^{-r}/400)$-fraction of inputs by a circuit of size s. Then f can be inverted on a $(1 - e^{-r})$-fraction of inputs by a circuit of size $O(s \cdot r \cdot 2^{3r})$.*

The proof of Claim 2 can be found in the full version of this paper.

Proof (of Theorem 3). Suppose $f\colon \{0,1\}^n \to \{0,1\}^m$ is a regular function with input locality d which is hard against circuits of size $\exp(O(\sqrt{rn} \cdot \alpha(d, r, \log n)))$ on a e^{-r}-fraction of inputs. Let $f'(x) = (f(x), h(x))$, where h is chosen as in Claim 1. It is easy to check that f' has the desired input locality and circuit complexity.

Now suppose f' can be inverted by a circuit of size $\exp(O(\sqrt{rn} \cdot \alpha(d, r, \log n)))$ on a e^{-r} fraction of its inputs. By Claim 1, f' is $e^{-r}/2$-close to 1-1. By Theorem 1 and Remark 1, f' can be inverted on a $(1-e^{-r}/400)$-fraction of inputs by a circuit of size $s = \exp(O(\sqrt{rn} \cdot \alpha(d, r, \log n)))$. By Claim 2, f can then be inverted on a $(1 - e^{-r})$ fraction of inputs by circuits of size $\exp(O(\sqrt{rn} \cdot \alpha(d, r, \log n)))$. □

Proof (of Corollary 1). Since $h(x)$ is a linear function, we can apply the randomized encoding of Applebaum et al. to reduce its output locality at the cost of increasing the input and output length of f'. Specifically, we perform the following transformation on f' to obtain a new function f''. Suppose the ith output $h(x)_i$ has the form

$$h(x)_i = x_{i1} + x_{i2} + \cdots + x_{ik_i}.$$

We introduce new inputs $r_{i1}, r_{i2}, \ldots, r_{i(k_i-1)}$ and replace the output $h(x)_i$ by the sequence of outputs:

$$(x_{i1} + r_{i1}, r_{i1} + x_{i2} + r_{i2}, \ldots, r_{i(k_i-1)} + x_{ik_i}).$$

It is easy to check that f'' has the desired input and output length, and its output locality is $\max\{d_{\mathsf{out}}, 3\}$.

Applebaum et al. [3,4] show that if f'' can be inverted on an ε-fraction of inputs by a circuit of size s, then f' can be inverted on a $\Omega(\varepsilon)$-fraction of inputs by a circuit of size $O(s/\varepsilon)$. Plugging in $\varepsilon = e^{-r}$ and $s = \exp(O(\sqrt{rn} \cdot \alpha(d_{\mathsf{in}}, r, \log n)))$, the corollary follows. □

6 Goldreich's Function on a Random Graph

We now consider two applications of our techniques to the candidate one-way function proposed by Goldreich [8]. Given a bipartite graph G with n vertices on the left, m vertices on the right, and regular right-degree d_{out} and a predicate $P\colon \{0,1\}^{d_{\mathsf{out}}} \to \{0,1\}$, the function $f_{G,P}$ from $\{0,1\}^n$ to $\{0,1\}^m$ is defined by

$$f_{G,P}(x)_i = \text{the } i\text{th bit of } f(x) = P(x_{\Gamma(i,1)}, \ldots, x_{\Gamma(i,d_{\mathsf{out}})})$$

where $\Gamma_{(i,j)}$ is the jth neighbor of right vertex i of G.

Goldreich [8] considered such constructions for the setting of parameters $m = n$ and d_{out} ranges from a constant to $O(\log n)$. He conjectured that when G is a good expander graph and P is a randomly chosen predicate, with high probability $f_{G,P}$ is one-way.

Cook et al. [7] showed that when G is random and P is suitably chosen, $f_{G,P}$ is secure against adversaries that implement *myopic algorithms*. Bogdanov and Qiao [6] studied a variant of Goldreich's function in the setting where G is random, d is constant, and $m = Dn$, where $D = D(d_{\mathsf{out}})$ is a sufficiently large constant. They showed that for a large class of predicates P (those that correlate with one or a pair of their inputs) and for most G, $f_{G,P}$ can be inverted on most inputs. It is conceivable that $f_{G,P}$ could be one-way for all predicates P that are not linear and do not belong to the class ruled out by Bogdanov and Qiao.

We establish two results regarding local hardness amplification of Goldreich's function. Informally, we show that

1. In the setting where d is constant and $m \geq Dn$, where $D = D(d_{\mathsf{out}})$ is a sufficiently large constant, for any P and with high probability over the choice of G if $f_{G,P}$ is hard to invert on an e^{-r} fraction of inputs in time $\exp(O(\sqrt{rn} \cdot d_{\mathsf{out}} \cdot \log n))$, then $f_{G,P}$ is hard to invert on a $1 - e^{-r}$ fraction of inputs in time $\exp(O(\sqrt{rn} \cdot d_{\mathsf{out}} \cdot \log n))$.
2. When $d_{\mathsf{out}} = O(\log n)$ and $m = n$, for a certain class of predicates P and with high probability over G, if $f_{G,P}$ is hard to invert on a e^{-r} fraction of inputs, then there exists a function $f'\colon \{0,1\}^{n'} \to \{0,1\}^{m'}$, where $n', m' = n \cdot \text{polylog} n$, of similar complexity to f and constant output locality that is hard to invert on a $1 - e^{-r}$ fraction of inputs.

   Our result applies to all $O(\log n)$*-parity-blowup* predicates, which we define as follows. Let $P_c\colon \{0,1\}^c \to \{0,1\}$ be any balanced predicate, where c is some constant. The d_{out}-parity-blowup of P_c is the predicate $P\colon \{0,1\}^{d_{\mathsf{out}}} \to \{0,1\}$ which is obtained by replacing each of the variables in P_c by a parity of $\lfloor d_{\mathsf{out}}/c \rfloor$ inputs, where all the inputs are distinct. Applebaum, Barak, and Wigderson [2] showed that the output of Goldreich's function based on such predicates is pseudorandom against linear functions, low-degree polynomials, and constant-depth circuits.

The random graph G is chosen from the following distribution: For each of the m right vertices of G, choose all of its d_{out} neighbors independently at random among the n left vertices of G. We will call such graphs (n, m, d_{out}) *random graphs*.

6.1 Self-reducibility for Functions with Long Output

Theorem 4. *Let $D \geq 2^{K d_{\mathsf{out}}}$ where K is a sufficiently large constant, and $P\colon \{0,1\}^{d_{\mathsf{out}}} \to \{0,1\}$ be any predicate. Let G be an (n, m, d_{out}) random graph.*

With probability at least $1 - o(1)$ *over the choice of* G*, if* $f_{G,P}$ *is hard for circuits of size* $\exp(O(\sqrt{rn} \cdot Dd_{\mathsf{out}} \log n))$ *on an* e^{-r}*-fraction of inputs, then* $f_{G,P}$ *is hard for circuits of size* $\exp(O(\sqrt{rn} \cdot Dd_{\mathsf{out}} \log n))$ *on a* $1 - e^{-r}$*-fraction of inputs.*

We prove Theorem 4 (in the full version of this paper) by an argument similar to the one used in the proof of Theorem 1. The principal obstacle to applying Theorem 1 here is that with high probability, the function $f_{G,P}$ is not 1-1. There are several reasons for this. One reason is that $f_{G,P}$ is likely to contain inputs that do not appear in any output. A more important reason is that unless the predicate P is linear, for most inputs x, it is likely that there is a linear number of coordinates i such that the ith coordinate does appear in the output, but changing the value of x_i does not change the value of $f_{G,P}(x)$.

We show that although $f_{G,P}$ is unlikely 1-1, with high probability every pair of inputs that map to the same output is highly correlated (or anticorrelated), that is they agree (or disagree) in value on most of the coordinates. Using the argument from the proof of Theorem 1, we show that if $f_{G,P}$ can be inverted on an ε-fraction on inputs by a circuit of suitable size, then for a $1 - \varepsilon$ fraction of inputs x, it is possible to find an x' such that x and x' are highly correlated. We then use a result of Bogdanov and Qiao [6] which says that for most inputs x, given x' that is correlated with x, then we can invert $f_{G,P}(x)$.

6.2 Amplification for Certain Length-Preserving Functions

Let $P\colon \{0,1\}^c \to \{0,1\}$ be a balanced predicate. The d_{out}*-parity-blowup* of P is the predicate obtained by replacing each variable in P by $\lfloor d_{\mathsf{out}}/c \rfloor$ variables, where all the new variables are disjoint.

Theorem 5. *Let* $c \geq 3$ *and* $d_{\mathsf{out}} = \max\{130c \log n, 4c^2\}$*. Let* P *be the* d_{out}*-parity-blowup of some balanced predicate on* c *bits and let* G *be an* (n, n, d_{out})*-random graph. Suppose that for at least half the graphs* G*,* $f_{G,P}\colon \{0,1\}^n \to \{0,1\}^n$ *is hard to invert on an* e^{-r}*-fraction of inputs against circuits of size* $\exp(O(r^{3/2}\sqrt{n} \log n))$*. Then there exists*

1. *A function* $f'\colon \{0,1\}^n \to \{0,1\}^{n+O(r)}$ *of circuit size* $O(n \log n + r)$ *that is hard on a* $1 - e^{-r}$*-fraction of inputs for circuits of size* $\exp(O(r^{3/2}\sqrt{n} \log n))$*.*
2. *A function* $f''\colon \{0,1\}^{n'} \to \{0,1\}^{m'}$*, where* $n', m' = O(n \cdot d(n)^c)$*,* f'' *is hard on a* $1 - e^{-r}$*-fraction of inputs for circuits of size* $\exp(O(r^{3/2}\sqrt{n} \log n))$ *and where every output of* f'' *depends on at most* $c + 1$ *inputs.*

Theorem 5 is proved in the full version of this paper. To prove part 1, we first show that the function $f_{G,P}$ is likely to be $e^{-r}/4$-close to $O(e^r)$-to-1. Using Claim 1, we then transform $f_{G,P}$ into a function $f'\colon \{0,1\}^n \to \{0,1\}^{n+O(r)}$ which is $e^{-r}/2$-close to 1-1. Using the self-reduction of Theorem 1, we then argue that if f' is easy to invert on an e^{-r} fraction of inputs, then it is also easy to invert on a $1 - e^{-r}$-fraction of inputs, and so $f_{G,P}$ is also easy to invert on the same fraction of inputs by circuits of similar size. To prove part 2, we observe that parity-blowup predicates can be represented constant degree polynomials over

$GF(2)$. Applying the randomized encodings of Applebaum et al. [3] to these polynomials, we obtain a function with constant output locality and slightly longer input and output length.

Acknowledgements. We thank the TCC'11 reviewers for useful comments.

References

1. Alon, N., Spencer, J.: The Probabilistic Method. John Wiley, Chichester (1992)
2. Applebaum, B., Barak, B., Wigderson, A.: Public-key cryptography from different assumptions. In: STOC 2010, pp. 171–180 (2010)
3. Applebaum, B., Ishai, Y., Kushilevitz, E.: Cryptography in NC0. In: FOCS 2004, pp. 166–175 (2004)
4. Applebaum, B., Ishai, Y., Kushilevitz, E.: Cryptography with Constant Input Locality. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 92–110. Springer, Heidelberg (2007); Journal version in J. Cryptology 22(4), 429-469 (2009)
5. Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. In: STOC 2000, pp. 435–440 (2000)
6. Bogdanov, A., Qiao, Y.: On the security of goldreich's one-way function. In: Dinur, I., Jansen, K., Naor, J., Rolim, J. (eds.) APPROX 2009. LNCS, vol. 5687, pp. 392–405. Springer, Heidelberg (2009)
7. Cook, J., Etesami, O., Miller, R., Trevisan, L.: Goldreich's One-Way Function Candidate and Myopic Backtracking Algorithms. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 521–538. Springer, Heidelberg (2009)
8. Goldreich, O.: Candidate One-Way Functions Based on Expander Graphs. Electronic Colloquium on Computational Complexity (ECCC) 7(90) (2000)
9. Goldreich, O., Impagliazzo, R., Levin, L.A., Venkatesan, R., Zuckerman, D.: Security Preserving Amplification of Hardness. In: FOCS 1990, pp. 318–326 (1990)
10. Goldreich, O., Krawczyk, H., Luby, M.: On the Existence of Pseudorandom Generators. SIAM J. Comput. 22(6), 1163–1175 (1993)
11. Haitner, I., Harnik, D., Reingold, O.: On the Power of the Randomized Iterate. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 22–40. Springer, Heidelberg (2006)
12. Impagliazzo, R., Levin, L., Luby, M.: Pseudo-random Generation from One-way Functions. In: STOC 1989, pp. 12–24 (1989)
13. Impagliazzo, R., Luby, M.: One-way Functions are Essential for Complexity Based Cryptography. In: FOCS 1989, pp. 230–235 (1989)
14. Mansour, Y., Nisan, N., Tiwari, P.: The Computational Complexity of Universal Hashing. Theor. Comput. Sci. 107(1), 121–133 (1993)
15. Lin, H.C., Trevisan, L., Wee, H.: On Hardness Amplification of One-Way Functions. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 34–49. Springer, Heidelberg (2005)
16. Yao, A.C.-C.: Theory and Applications of Trapdoor Functions (Extended Abstract). In: FOCS 1982, pp. 80–91 (1982)