J. Cryptology (1996) 9: 251–261

Journal of
CRYPTOLOGY

Pseudorandom Generators and the Frequency of Simplicity*

Yenjo Han and Lane A. Hemaspaandra
Department of Computer Science, University of Rochester,
Rochester, NY 14627, U.S.A.

Communicated by Andrew C.-C. Yao

Received 17 November 1994 and revised 23 June 1995

Abstract. Allender [2] showed that if there are dense P languages containing only a **finite** set of Kolmogorov-simple strings, then all pseudorandom generators are insecure. We extend this by proving that if there are dense P (or even BPP) languages containing only a **sparse** set of Kolmogorov-simple strings, then all pseudorandom generators are insecure.

Key words. Pseudorandom generator, Computational complexity, Kolmogorov complexity, Injectivity of pseudorandom generators.

1. Introduction

A pseudorandom generator is a deterministic polynomial-time algorithm that takes a short random seed and produces a long output. A pseudorandom generator is *secure* if the ensemble of its outputs is indistinguishable from a truly random distribution of strings. The existence of secure pseudorandom generators has been a subject of keen interest to many researchers ever since Yao [19] formally defined secure pseudorandom generators. Blum and Micali [4] showed that secure pseudorandom generators can be constructed under the assumption that the discrete logarithm problem is hard. Yao [19] showed how to construct a secure pseudorandom generator from any one-way permutation. Thereafter, many results on the relationship between pseudorandom generators and one-way functions showed that pseudorandom generators can be constructed from less restrictive one-way functions [15], [9], [13], [11]. This line of research culminated with results proving, in both the nonuniform model [13] and the uniform model [11], that the existence of secure pseudorandom generators is equivalent to the existence of one-way functions.

* This research was supported in part by Grants NSF-CCR-8957604, NSF-INT-9116781/JSPS-ENG-207, and NSF-CCR-9322513. The work of the second author was done in part while visiting the University of Electro-Communications, Tokyo.

In contrast, the study of the relationship between the existence of secure pseudorandom generators and the frequency of Kolmogorov-simple strings in dense languages was undertaken much later. Allender [2] showed that if there are dense P languages containing only a *finite* set of Kolmogorov-simple strings, then all pseudorandom generators are insecure. In this paper we continue this line of research by showing that if there are dense P (or even BPP) languages containing only a *sparse* set of Kolmogorov-simple strings, then all pseudorandom generators are insecure. We do this by first proving that the injectivity (degree of "many-to-one"-ness) of secure pseudorandom generators is bounded by an arbitrary polynomial fraction. We also prove similar bounds on the injectivity of secure pseudorandom extenders and the injectivity of pseudorandom string generators derived from any secure pseudorandom extender.

The rest of the paper consists of three sections. In Section 2 we introduce definitions and prove some useful lemmas. In Section 3 we prove results on the injectivity of secure pseudorandom generators and extenders. In Section 4 we study the relationship between the existence of secure pseudorandom generators and the frequency of simplicity in dense P (and BPP) languages.

2. Preliminaries

Throughout this paper we use the alphabet $\Sigma = \{0, 1\}$. For any set L and any integer n, $L^{=n}$ denotes the set of all length n strings in L. P denotes the class of languages that can be recognized by deterministic polynomial-time Turing machines (see, e.g., [12] for the definitions of Turing machines and other standard notions used in this paper). For any finite set A, we use $\|A\|$ to denote the cardinality of A. A set L is *dense* if there is a constant d such that, for infinitely many values of n, $\|L^{=n}\| \geq n^{-d}2^n$. A set L is *sparse* if there is a constant s such that, for all $n \geq 2$, $\|L^{=n}\| \leq n^s$.

A pseudorandom generator is a deterministic polynomial-time function that, for some constant $k > 1$ and for each n, has the property that, when given an input of length n, the function outputs a string of length n^k. A pseudorandom extender is a deterministic polynomial-time function that, when given an input of length n, outputs a string of length $n + 1$. A pseudorandom generator or extender is considered secure if the distribution of its outputs is indistinguishable from a truly random distribution. The formalization of security is done in terms of statistical tests [19]. A statistical test is a probabilistic polynomial-time decision algorithm. Let Π be a statistical test. $\mathrm{P}_\Pi(\Sigma^n)$ is the probability with which Π accepts an input string that is drawn uniformly from Σ^n. Let g be a pseudorandom generator or extender. $\mathrm{PS}_\Pi(g, \Sigma^n)$ is the probability with which Π accepts $g(x)$, when $g(x)$ is formed by drawing x uniformly from Σ^n.

Definition 2.1 (Secure Pseudorandom Generator) [19]. Let g be a pseudorandom generator that, for some constant $k > 1$ and every n, maps each input of length n to an output string of length n^k. Given a statistical test Π and a positive constant p, *g passes the statistical test Π with precision n^{-p}* if an integer n_0 exists such that, for all $n \geq n_0$, it holds that $|\mathrm{P}_\Pi(\Sigma^{n^k}) - \mathrm{PS}_\Pi(g, \Sigma^n)| < n^{-p}$. g is *secure* if, for all statistical tests Π and for all constants p, it passes the statistical test Π with precision n^{-p}.

The security of a pseudorandom extender is defined similarly [6]. It is known that secure pseudorandom generators exist if and only if secure pseudorandom extenders exist [6]. Since pseudorandom extenders are more convenient in proving our results, we, following the approach of Allender [2], consider only pseudorandom extenders in Section 4 of this paper. Nevertheless, the resulting theorems are valid even when stated in terms of pseudorandom generators.

Given a pseudorandom extender g, the following definition of g_b provides a method for generating pseudorandom strings of arbitrary length. Note that since g_b outputs only strings of one length, g_b does not qualify as a pseudorandom generator. Nonetheless, g_b generates perfectly good pseudorandom strings if g is secure and the length of the input is smaller than b. Henceforward, we refer to g_b as a "pseudorandom string generator" derived from the pseudorandom extender g.

Definition 2.2 [6]. Let g be a pseudorandom extender. For any string x and any $b \in \{0, 1, 2, \ldots\}$, $g_b(x)$ is defined as

$$g_b(x) = c_1(x)c_2(x)\cdots c_b(x),$$

where

$$c_i(x) = \text{head} \circ g(t_i(x))$$

and

$$t_i(x) = (\text{tail} \circ g)^{i-1}(x),$$

and head(x) is the first character of the string x and tail(x) is the remainder of x after the first character is taken off of it.

Note that the time complexity of g_b is $\mathcal{O}(bt_g(n))$ where $t_g(n)$ is the time complexity of g.

Obviously, a polynomial-time algorithm M_L that recognizes a language L can be considered a statistical test. Allender [2], via a generalization of a technique used in the earlier literature [6], [19], [5], showed that given a language L, there is a statistical test $T(L, g, b)$ that effectively "boosts" a test of a pseudorandom extender g to a test of the "generator" g_b. (See Lemma 2.4 below.) If b is sufficiently greater than the length of the input string, then, since g_b maps a short string to a much longer string, the output string has low Kolmogorov complexity, and, thus, it is easy to apply Kolmogorov complexity argumentation to the test of g_b. It turns out that the boosting effect remains unchanged even if we replace M_L with an arbitrary probabilistic polynomial-time decision algorithm A. Below, we introduce the statistical test $T(A, g, b)$ and the "boosting lemma" in a slightly generalized form that we will need later.

Definition 2.3 [2]. Let A, g, and b be a statistical test, a pseudorandom extender, and a positive integer, respectively. Given an input x of length $n + 1$, the statistical test

$T(A, g, b)$ performs the following algorithm:

```
Probabilistically choose $i \in \{0, \ldots, b-1\}$.
Probabilistically choose $z \in \Sigma^{b-i-1}$.
Let $y = g_i(\text{tail}(x))$, and let $c = \text{head}(x)$.
Accept iff $A$ accepts $zcy$.
```

Note that it is not necessarily true that i can be chosen from $\{0, \ldots, b-1\}$ with equal probability regardless of the value of b. For example, an algorithm with a two-sided die cannot simulate an equiprobability roll of a three-sided die if we always require an output. Nonetheless, it is easy and standard to design a polynomial-time algorithm that chooses i so close to equiprobably that the very small error is negligible for our purposes. Since the error does not affect the validity of our results in this paper, we ignore it and assume that i can be chosen from $\{0, \ldots, b-1\}$ with equal probability.

Lemma 2.4 (Boosting Lemma). *Let A and g be a statistical test and a pseudorandom extender, respectively. For all n and b, it holds that*

$$\mathrm{P}_{T(A,g,b)}(\Sigma^{n+1}) - \mathrm{PS}_{T(A,g,b)}(g, \Sigma^n) = \frac{1}{b}(\mathrm{P}_A(\Sigma^b) - \mathrm{PS}_A(g_b, \Sigma^n)).$$

The proof of Lemma 2.4 is a straightforward generalization of the proof of the corresponding result of Allender [2].

The following lemma shows that when a polynomial number of pinpoints are randomly dropped on the unit interval, any interval collection of "meaningful" size is hit by at least one of the pinpoints with very high probability. This lemma plays an important role in the proof of the injectivity results in Section 3.

Definition 2.5. Let n be a positive integer. Consider a collection of mutually disjoint intervals in $[0, 1]$, the total length of which is n^{-l}. Let $\gamma_l(n)$ denote the probability with which no interval in the collection is hit by any of n^{2l} pinpoints that are randomly dropped on $[0, 1]$.

Lemma 2.6 (Pin Dropping Lemma). *For all $l \in Z^+$, $\gamma_l(n) = \mathcal{O}(2^{-n^l})$.*

Proof. Clearly, $\gamma_l(n) = \left(1 - n^{-l}\right)^{n^{2l}}$. Since $\lim_{n\to\infty}(\gamma_l(n))^{n^{-l}} = e^{-1} < \frac{1}{2}$, it follows that $\gamma_l(n) = \mathcal{O}(2^{-n^l})$. □

Kolmogorov complexity (see [16]) provides an important tool to describe the complexity of each individual string. The Kolmogorov complexity of a given string x is the length of the shortest code that, as input to a (certain) universal Turing machine, yields output x. The short code, in a sense, plays the role of a *description* of the longer string x. In order to describe sets of strings with low Kolmogorov complexity, we adopt the time-bounded Kolmogorov complexity set notation of Hartmanis ([10], see also [18], and, generally, [16] for a discussion of the history). Let M_v be a Turing machine. $K_v[s(n), t(n)]$ is the set $\{x \in \Sigma^* \mid (\exists y \in \Sigma^*)[|y| \leq s(|x|) \wedge M_v(y)$ prints x in time $t(|x|)]\}$. Hartmanis [10] generalized the standard invariance theorem to this time-bounded case, noting that there is a *universal* Turing machine u such that, for all v, there is a constant c such that, for all s and t, $K_v[s(n), t(n)] \subseteq K_u[s(n) + c, ct(n) \log t(n) + c]$. In the rest of the paper we use $K[s(n), t(n)]$ to mean $K_u[s(n), t(n)]$.

3. Injectivity of Pseudorandom Generators

We prove an upper bound on the injectivity of secure pseudorandom generators and pseudorandom extenders. We also show that the same bound applies to the pseudorandom string generators that are derived from a pseudorandom extender using the method of Definition 2.2.

Theorem 3.1. *Let g be a secure pseudorandom generator. Then, for any integer l, an integer n_0 exists such that it holds that, for any integer $n \geq n_0$,*

$$\max_{x \in \Sigma^n} \|g^{-1}(g(x))\| \leq n^{-l} 2^n.$$

Proof. Since the proof is trivial for $l \leq 0$, we assume $l > 0$ for the rest of the proof. We prove the contrapositive. Let g be a pseudorandom generator such that an integer l exists such that, for infinitely many n, an $x \in \Sigma^n$ exists that satisfies

$$\|g^{-1}(g(x))\| > n^{-l} 2^n.$$

It suffices to show that g is not secure.

Let k be the integer such that $g(\Sigma^n) \subseteq \Sigma^{n^k}$. Consider the following statistical test A. Given a string $y \in \Sigma^{n^k}$, A executes the following algorithm:

```
Guess $n^{2l}$ strings of length $n$. Let $S'$ denote these strings.
If $y$ is in $g(S')$, then accept $y$;
otherwise, reject $y$.
```

Since A rejects all inputs from $\Sigma^{n^k} - g(\Sigma^n)$ with probability one, it holds that, for every length n,

$$\mathrm{P}_A(\Sigma^{n^k}) \leq \frac{\|g(\Sigma^n)\|}{\|\Sigma^{n^k}\|} \leq 2^{n-n^k}.$$

On the other hand, consider a length n for which there is a string x that satisfies

$$\|g^{-1}(g(x))\| > n^{-l} 2^n.$$

(Recall that there are infinitely many values of n that satisfy this.) Since the probability with which A accepts $g(x)$ is at least $1 - \gamma_l(n)$,

$$\mathrm{PS}_A(g, \Sigma^n) \geq \frac{\|g^{-1}(g(x))\|}{2^n}(1 - \gamma_l(n)) \geq n^{-l}(1 - \gamma_l(n)).$$

Note that $2^{n-n^k} \le n^{-l}/3$ for all sufficiently large n and that, from the result of Lemma 2.6, $\gamma_l(n) \le \frac{1}{3}$ for all sufficiently large n. Thus, it is easy to see that, for infinitely many values of n,

$$|\mathrm{P}_A(\Sigma^{n^k}) - \mathrm{PS}_A(g, \Sigma^n)| \ge n^{-l-1}.$$

Hence, g is not secure. □

Theorem 3.2. *Let g be a secure pseudorandom extender. Then, for any integer l, an integer n_0 exists such that it holds that, for any integer $n \ge n_0$,*

$$\max_{x \in \Sigma^n} \|g^{-1}(g(x))\| \le n^{-l} 2^n.$$

Proof. We prove the contrapositive. Let g be a pseudorandom extender such that an integer l exists such that, for infinitely many n, an $x \in \Sigma^n$ exists that satisfies

$$\|g^{-1}(g(x))\| > n^{-l} 2^n.$$

It suffices to show that g is not secure.

Let $k \ge 2$ be an integer constant. In order to use the result of Theorem 3.1, we define a pseudorandom generator h derived from g as

$$h(x) = g_{|x|^k}(x).$$

Consider a statistical test A_h that is identical to the test A of the proof of Theorem 3.1 except that A_h uses h instead of g. Since, for any x and y, $g(x) = g(y)$ implies $h(x) = h(y)$, it is clear that, for any n and x, $\|h^{-1}(h(x))\| \ge \|g^{-1}(g(x))\|$. From the assumption, it follows that, for infinitely many n, an $x \in \Sigma^n$ exists that satisfies

$$\|h^{-1}(h(x))\| > n^{-l} 2^n.$$

By the proof of Theorem 3.1, this implies that, for infinitely many values of n,

$$|\mathrm{P}_{A_h}(\Sigma^{n^k}) - \mathrm{PS}_{A_h}(h, \Sigma^n)| \ge n^{-l-1}.$$

By Lemma 2.4, this in turn implies that

$$|\mathrm{P}_{T(A_h, g, n^k)}(\Sigma^{n+1}) - \mathrm{PS}_{T(A_h, g, n^k)}(g, \Sigma^n)| \ge n^{-k-l-1}.$$

Hence, g is not secure. □

The following theorem on the injectivity of pseudorandom string generators derived from a pseudorandom extender will be useful in proving the results of Section 4.

Definition 3.3. Let g be a secure pseudorandom extender and let k be a positive integer. Let $\rho_{g,k}(n)$ denote the maximum injectivity of g_b in the range $n^k \le b < n^{k+1}$ for inputs of length n; that is,

$$\rho_{g,k}(n) = \max_{\substack{x \in \Sigma^n \\ n^k \le b < n^{k+1}}} \|(g_b^{-1}(g_b(x)))^{=n}\|.$$

Theorem 3.4. *Let g be a secure pseudorandom extender. Then, for any positive integers $k \geq 2$ and l, an integer n_0 exists such that it holds that, for any integer $n \geq n_0$,*

$$\rho_{g,k}(n) \leq n^{-l}2^n.$$

Proof. The proof of this theorem easily follows from the proof of Theorem 3.2 since, for any $x \in \Sigma^n$ and any b that satisfies $n^k \leq b < n^{k+1}$, it holds that

$$\|h^{-1}(h(x))\| \geq \|(g_b^{-1}(g_b(x)))^{=n}\|. \qquad \square$$

4. The Frequency of Simplicity

Allender [2] showed that if a secure pseudorandom extender exists, every dense language L in P has infinitely many easy strings. Allender stated this result using a Kolmogorov complexity notation of Levin ([14], see also [3]) that blurs together description length and time complexity. The following theorem restates his result using the now standard time-bounded Kolmogorov complexity notation [10].

Theorem 4.1 [2]. *If there are a dense set $L \in$ P and $\varepsilon > 0$ such that $L \cap K[n^\varepsilon, 2^{n^\varepsilon}]$ is finite, no pseudorandom extender is secure.*[1]

With the help of Lemma 2.4, the proof idea can easily be sketched as follows. Let g be a pseudorandom extender. Let M be a P machine that accepts L. We show that g is insecure by applying the statistical test M to g_b. Since L is dense, there is a number $d > 0$ such that $\mathrm{P}_M(\Sigma^b)$ (i.e., $\|L^{=b}\|/2^b$) is greater than b^{-d} for infinitely many b. Thus, in order to pass the statistical test M with precision n^{-p} for an arbitrary p, $\mathrm{PS}_M(g_b, \Sigma^n)$ must be close to $\mathrm{P}_M(\Sigma^b)$ and certainly greater than zero for infinitely many values of b that are within reach of an arbitrary polynomial in n. However, since $L \cap K[n^\varepsilon, 2^{n^\varepsilon}]$ is finite, for all but finitely many n and for $b \geq n^{1/\varepsilon}$, $L \cap g_b(\Sigma^n)$ is empty and, consequently, $\mathrm{PS}_M(g_b, \Sigma^n)$ is zero. Thus, g fails the statistical test M with precision n^{-p} for sufficiently large values of p. Therefore, g is not secure.

Starting from the same assumption as Allender's, we now draw a stronger conclusion: if a secure pseudorandom extender exists, every dense language in P has a nonsparse subset of easy strings. The proof below synthesizes the above proof idea and the injectivity result of Theorem 3.4.

Theorem 4.2. *If there are a dense set $L \in$ P, $\varepsilon > 0$, and $t > 1$ such that $L \cap K[n^\varepsilon, n^t]$ is sparse, then no pseudorandom extender is secure.*

[1] In an earlier conference paper [1], Allender originally made the following claim that is stronger than Theorem 4.1: if there are a dense set $L \in$ P, $\varepsilon > 0$, and $t > 1$ such that $L \cap K[n^\varepsilon, n^t]$ is finite, no pseudorandom extender is secure. However, the lemma that was used to establish this claim had an invalid proof. The proof overlooked the fact that the range of a function $r(n)$ with the integers as its domain may miss infinitely many integers in the case that $r(n) = \Omega(n)$. Nonetheless, Allender's original claim holds as it is a restricted version of Theorem 4.2.

Proof. Let g be a pseudorandom extender. We show that g is not secure by using the test Π defined below. Let j be a constant such that the running time of g is in $\mathcal{O}(n^j)$. Let h be an integer that satisfies $h > 1/\varepsilon$ and $1 + j/h < t$. Let $f(n) = n^h$. Since L is dense, a positive integer d exists such that $\|L^{=n}\| \geq n^{-d}2^n$ infinitely often. Let M be a polynomial-time decision algorithm that recognizes L. Given an input x of length $n + 1$, Π executes the following algorithm:

```
Probabilistically choose $b \in \{f(n), \ldots, f(n+1) - 1\}$.
Run $T(M, g, b)$ on $x$.
```

Note that as n spans all lengths, b covers all lengths, too. It is not hard to see that Π is a polynomial-time algorithm. Let

$$\Delta(n) = \mathrm{P}_\Pi(\Sigma^{n+1}) - \mathrm{PS}_\Pi(g, \Sigma^n).$$

To prove that g is not secure, it suffices to show that there is an integer p such that $|\Delta(n)|$ is greater than n^{-p} for infinitely many values of n. From the definition of Π, it is easy to see that

$$\Delta(n) = \frac{1}{f(n+1) - f(n)} \sum_{b=f(n)}^{f(n+1)-1} (\mathrm{P}_{T(M,g,b)}(\Sigma^{n+1}) - \mathrm{PS}_{T(M,g,b)}(g, \Sigma^n)).$$

Applying the Boosting Lemma (Lemma 2.4), we get

$$\Delta(n) = \frac{1}{f(n+1) - f(n)} \sum_{b=f(n)}^{f(n+1)-1} \frac{1}{b}(\mathrm{P}_M(\Sigma^b) - \mathrm{PS}_M(g_b, \Sigma^n)).$$

Since $\mathrm{P}_M(\Sigma^b)$ and $\mathrm{PS}_M(g_b, \Sigma^n)$ in the sum satisfy

$$\mathrm{P}_M(\Sigma^b) = \frac{\|L^{=b}\|}{2^b}$$

and

$$\mathrm{PS}_M(g_b, \Sigma^n) = \sum_{x \in \Sigma^n} \frac{\mathrm{Prob}(M(g_b(x)) \text{ accepts})}{2^n} \leq \sum_{y \in g_b(\Sigma^n)} \frac{\rho_{g,h}(n)\ \mathrm{Prob}(M(y) \text{ accepts})}{2^n},$$

we get the following after simple rewriting:

$$\Delta(n) \geq \frac{1}{(f(n+1))^2} \left[\max_{f(n) \leq b < f(n+1)} \frac{\|L^{=b}\|}{2^b} - \sum_{b=f(n)}^{f(n+1)-1} \frac{\rho_{g,h}(n)}{2^n} \|L \cap g_b(\Sigma^n)\| \right].$$

It is not hard to see that, for almost all n, if $b \geq n^h$, then $g_b(\Sigma^n) \subseteq K[n^\varepsilon, n^t]$. Since $L^{=n} \cap K[n^\varepsilon, n^t]$ is sparse, it follows that an s exists that satisfies $\|L \cap g_b(\Sigma^n)\| \leq b^s$ for all sufficiently large b. Recall that $\|L^{=b}\| \geq b^{-d}2^b$ for infinitely many values of b. Thus, choosing $l = (s + d + 1)h + 1$ in Theorem 3.4, it follows that, for infinitely many values of n,

$$\Delta(n) \geq \frac{1}{(n+1)^{2h}} \left[\frac{1}{n^{dh}} - \frac{(n+1)^{sh+1}}{n^l} \right] \geq n^{-(d+2)h-1}.$$

Hence, g is not secure. □

We note that Zimand [21], building on an analog of the contrapositive of the result of this paper and on Nisan's work [17], has obtained interesting results about properties of large sets in AC^0.

BPP [8] is the class of languages that can be recognized by probabilistic polynomial-time Turing machines whose accuracy is at least $\frac{1}{2} + \lambda$ for all inputs, where λ is any constant in the range $0 < \lambda < \frac{1}{2}$. That is, if L is a set in BPP, then there is a probabilistic polynomial-time Turing machine M such that, for each x,

$$x \in L \qquad \text{iff} \quad \mathrm{Prob}(M \text{ accepts } x) \geq \tfrac{1}{2} + \lambda,$$

and

$$x \notin L \qquad \text{iff} \quad \mathrm{Prob}(M \text{ rejects } x) \geq \tfrac{1}{2} + \lambda.$$

It is well known that the accuracy of a BPP set can be amplified greatly. Given a polynomial q, the accuracy of a BPP-machine can be made to exceed $1 - 2^{-q(|x|)}$ for all inputs x (see, e.g., [20]). With such a small error bound possible for BPP, it is not hard to see that a slight modification of the above proof leads to the following result.

Theorem 4.3. *If there are a dense set $L \in$ BPP, $\varepsilon > 0$, and $t > 1$ such that $L \cap K[n^{\varepsilon}, n^{t}]$ is sparse, then no pseudorandom extender is secure.*

It would be interesting if we could show the existence of easy strings at each length where the density of strings is high. Though such existence remains an open question, Allender showed a related positive result by employing (a somewhat idiosyncratic definition of) *a.e. dense* sets, instead of *dense* sets.

Definition 4.4 [2]. A set L is *a.e. dense* if L is infinite and, for some d and for all large n, $L^{=n} \neq \emptyset \Rightarrow \|L^{=n}\| \geq n^{-d}2^{n}$.

Theorem 4.5 [2]. *If secure pseudorandom extenders exist, then, for each a.e. dense* P *set L and for all k, there exists c such that, for almost all n, it holds that*

$$L^{=n^k} \neq \emptyset \quad \Rightarrow \quad \|L^{=n^k} \cap K[cn^{1/k}, 2^{cn^{1/k}}]\| > 0.$$

Using the proof technique developed in this paper, we can strengthen Theorem 4.5 as follows.

Theorem 4.6. *If secure pseudorandom extenders exist, then, for each a.e. dense* BPP *set L and for all k, there exists t such that, for all s and for almost all n, it holds that*

$$L^{=n^k} \neq \emptyset \quad \Rightarrow \quad \|L^{=n^k} \cap K[n^{1/k}, n^{t}]\| > n^{sk}.$$

All the results of this paper so far are with respect to security against probabilistic polynomial-time statistical tests. It might be asked whether this paper's generalizations also hold with respect to security against circuit-based nonuniform statistical tests in P/poly. It turns out that such a generalization is trivial. Consider the following theorem.

Theorem 4.7 [2]. *If pseudorandom extenders exist that are secure against statistical tests in* P/poly, *then, for each a.e. dense* P/poly *set L and for all $\varepsilon > 0$, there exists c such that, for almost all n, it holds that*

$$L^{=n} \neq \emptyset \quad \Rightarrow \quad \|L^{=n} \cap K[cn^{\varepsilon}, 2^{cn^{\varepsilon}}]\| > 0.$$

This theorem can be generalized as Theorem 4.8. The generalization is indeed trivial because the "generalization" from 0 to n^s can be handled by coding polynomially many strings into a circuit, and the generalization from P/poly to BPP/poly is trivialized by the fact[2] that P/poly = BPP/poly = $\bigcup_{S \in \mathrm{SPARSE}} \mathrm{BPP}^S$, where SPARSE denotes the class of sparse sets.

Theorem 4.8. *If pseudorandom extenders exist that are secure against statistical tests in* P/poly, *then, for each a.e. dense* BPP/poly *set L and for all $\varepsilon > 0$, there exists t such that, for all s and for almost all n, it holds that*

$$L^{=n} \neq \emptyset \quad \Rightarrow \quad \|L^{=n} \cap K[n^{\varepsilon}, n^{t}]\| > n^{s}.$$

Note that Theorem 4.8 subsumes the nonuniform versions of Theorems 4.2 and 4.6. It is an open question whether the uniform equivalent of Theorem 4.8 holds.

Acknowledgments

We are very grateful to Osamu Watanabe, Mitsunori Ogiwara, and Seinosuke Toda for hosting a visit to Tokyo during which this work was done in part, and to Eric Allender for literature pointers, helpful comments, and suggestions, and for a discussion that helped to strengthen Theorem 4.2 to its current form. We are also grateful to Marius Zimand for proofreading the paper.

References

[1] E. Allender. Some consequences of the existence of pseudorandom generators, preliminary version. *Proceedings of the 19th ACM Symposium on Theory of Computing*, pages 151–159, 1987.

[2] E. Allender. Some consequences of the existence of pseudorandom generators. *Journal of Computer and System Sciences*, 39:101–124, 1989.

[3] E. Allender. Applications of time-bounded Kolmogorov complexity in complexity theory. In O. Watanabe, editor, *Kolmogorov Complexity and Computational Complexity*, EATCS Monographs on Theoretical Computer Science, pages 4–22. Springer-Verlag, New York, 1992.

[4] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo-random bits. *Proceedings of the 23rd IEEE Symposium on Foundations of Computer Science*, pages 112–117, 1982. Final version appears as [5].

[5] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM Journal on Computing*, 13(4):850–864, 1984.

[2] The fact is clear as all these are in (P/poly)/poly = P/poly. Note that it is not in general the case that $\mathcal{C}$/poly = $\bigcup_{S \in \mathrm{SPARSE}} \mathcal{C}^S$, and, in fact, some "promise classes" may not meet this equality [7], though as we have just noted, the promise class BPP (also R, coR, and ZPP) does satisfy the equality.

[6] R. Boppana and R. Hirschfeld. Pseudorandom generators and complexity classes. In *Advances in Computing Research*, volume 5, pages 1–26. JAI Press, Greenwich, CT, 1989.

[7] R. Gavaldà and J. Balcázar. Strong and robustly strong polynomial-time reducibilities to sparse sets. *Theoretical Computer Science*, 88:1–14, 1991.

[8] J. Gill. Computational complexity of probabilistic Turing machines. *SIAM Journal on Computing*, 6(4):675–695, 1977.

[9] O. Goldreich, H. Krawczyk, and M. Luby. On the existence of pseudorandom generators. *SIAM Journal on Computing*, 22(6):1163–1175, 1993.

[10] J. Hartmanis. Generalized Kolmogorov complexity and the structure of feasible computations. *Proceedings of the* 24*th IEEE Symposium on Foundations of Computer Science*, pages 439–445, 1983.

[11] J. Håstad. Pseudo-random generators under uniform assumptions. *Proceedings of the* 22*nd ACM Symposium on Theory of Computing*, pages 395–404, 1990.

[12] J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, Reading, MA, 1979.

[13] R. Impagliazzo, L. Levin, and M. Luby. Pseudo-random generation from one-way functions. *Proceedings of the* 21*st ACM Symposium on Theory of Computing*, pages 12–24, 1989.

[14] L. Levin. Randomness conservation inequalities; information and independence in mathematical theories. *Information and Control*, 61:15–37, 1984.

[15] L. Levin. One way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.

[16] M. Li and P. Vitanyi. *An Introduction to Kolmogorov Complexity and Its Applications.* Springer-Verlag, New York, 1993.

[17] N. Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica*, 11(1):63–70, 1991.

[18] M. Sipser. A complexity theoretic approach to randomness. *Proceedings of the* 15*th ACM Symposium on Theory of Computing*, pages 330–335, 1983.

[19] A. Yao. Theory and applications of trapdoor functions. *Proceedings of the* 23*rd IEEE Symposium on Foundations of Computer Science*, pages 80–91, 1982.

[20] S. Zachos and H. Heller. A decisive characterization of BPP. *Information and Control*, 69:125–135, 1986.

[21] M. Zimand. Large sets in AC^0: a Kolmogorov complexity related property and some applications. *Proceedings of the* 7*th International Conference on Computing and Information*, to appear.