Pseudorandom Generators from Regular One-way Functions: New Constructions with Improved Parameters

Yu Yu*†

Abstract

We revisit the problem of basing pseudorandom generators on regular one-way functions, and present the following constructions:

- For any known-regular one-way function (on n-bit inputs) that is known to be ε-hard to invert, we give a neat (and tighter) proof for the folklore construction of pseudorandom generator of seed length $\Theta(n)$ by making a single call to the underlying one-way function.
- For any unknown-regular one-way function with known ε-hardness, we give a new construction with seed length $\Theta(n)$ and $O(n/\log(1/\varepsilon))$ calls. Here the number of calls is also optimal by matching the lower bounds of Holenstein and Sinha [FOCS 2012].

Both constructions require the knowledge about ε, but the dependency can be removed while keeping nearly the same parameters. In the latter case, we get a construction of pseudo-random generator from any unknown-regular one-way function using seed length $\tilde{O}(n)$ and $\tilde{O}(n/\log n)$ calls, where $\tilde{O}$ omits a factor that can be made arbitrarily close to constant (e.g. $\log\log\log n$ or even less). This improves the *randomized iterate* approach by Haitner, Harnik and Reingold [CRYPTO 2006] which requires seed length $O(n{\cdot}\log n)$ and $O(n/\log n)$ calls.

Keywords: Foundations, Pseudorandom Generators, One-way Functions, Randomized Iterate.

*Institute for Interdisciplinary Information Sciences, Tsinghua University. Email: yuyuathk@gmail.com.
†East China Normal University University.

1 Introduction

The seminal work of Håstad, Impagliazzo, Levin and Luby (HILL) [12] that one-way functions (OWFs) imply pseudorandom generators (PRGs) constitutes one of the centerpieces of modern cryptography. Technical tools and concepts (e.g. pseudo-entropy, leftover hash lemma) developed and introduced in [12] were found useful in many other contexts (such as leakage-resilient cryptography). Nevertheless, a major drawback of [12] is that the construction is quite involved and too inefficient to be of any practical use, namely, to obtain a PRG with comparable security to the underlying OWF on security parameter n, one needs a seed of length $O(n^8)$[1]. Research efforts (see [13, 11, 21], just to name a few) have been followed up towards simplifying and improving the constructions, and the current state-of-the-art construction [21] requires seed length $O(n^3)$. Let us mention all aforementioned approaches are characterized by a parallel construction, namely, they run sufficiently many independent copies of the underlying OWFs (rather than running a single trail and feeding its output back to the input iteratively) and there seems an inherent lower bound on the number of copies needed. This is recently formalized by Holenstein and Sinha [14], in particular, they showed that any black-box construction of a PRG from an arbitrary OWF f requires $\Omega(n/\log n)$ calls to f in general.[2]

PRGs from Special OWFs. Another line of research focuses on OWFs with special structures that give rise to more efficient PRGs. Blum, Micali [2] and Yao [23] independently introduced the notion of PRGs, and observed that PRGs can be efficiently constructed from one-way permutations (OWPs). That is, given a OWP f on input x and its hardcore function h_c (e.g. by Goldreich and Levin [8]), a single invocation of f already implies a PRG $g : x \mapsto (f(x), h_c(x))$ with a stretch[3] of $\Omega(\log n)$ bits and it extends to arbitrary stretch by repeated iterations (seen by a hybrid argument):

$$x \;\mapsto\; (\; h_c(x), h_c(f^1(x)), \ldots, h_c(f^\ell(x)), \ldots)$$

where $f^i(x) \stackrel{\mathsf{def}}{=} f(f^{i-1}(x))$ and $f^1(x) \stackrel{\mathsf{def}}{=} f(x)$. The above PRG, often referred to as the BMY generator, enjoys many advantages such as simplicity, optimal seed length, and minimal number of calls. Levin [17] observed that f is not necessarily a OWP, but it suffices to be one-way on its own iterate. Unfortunately, an arbitrary OWF doesn't have this property. Goldreich, Krawczyk, and Luby [7] assumed known-regular[4] OWFs and gave a construction of seed length $O(n^3)$ by iterating the underlying OWFs and applying k-wise independent hashing in between every two iterations. Later Goldreich showed a more efficient (and nearly optimal) construction from known-regular OWFs in his textbook [5], where in the concrete security setting the construction does only a single call to the underlying OWF (or $\omega(1)$ calls in general). The construction was also implicit in many HILL-style constructions (e.g. [13, 11]). Haitner, Harnik and Reingold [10] refined the technique used in [7] (which they called the *randomized iterate*) and adapted the construction to unknown regular OWFs with reduced seed length $O(n \cdot \log\ n)$. Informally, the randomized iterate follows the route of [7] and applies a random pairwise independent hash function h_i in between every two applications of f, i.e.

$$f^1(x) \stackrel{\mathsf{def}}{=} f(x); \text{for } i{\geq}2 \text{ let } f^i(x; h_1, \ldots, h_{i-1}) \stackrel{\mathsf{def}}{=} f(h_{i-1}(f^{i-1}(x; h_1, \cdots, h_{i-2})))$$

The key observation is "*the last iterate is hard-to-invert*" [9], more precisely, function f, when applied to $h_{i-1}(f^{i-1}; h_1, \cdots, h_{i-2})$, is hard-to-invert even if h_1, ..., h_{i-1} are made public. The generator follows by running the iterate $O(n/\log n)$ times, and outputting $\Omega(\log n)$ hardcore bits per iteration, which requires

[1] More precisely, the main construction of [12] requires seed length $O(n^{10})$, but [12] also sketches another construction of seed length $O(n^8)$, which was formalized and proven in [13].

[2] The lower bound of [14] also holds in the concrete security setting, namely, $\Omega(n/\log(1/\varepsilon))$ calls from any ε-hard OWF.

[3] The stretch of a PRG refers to the difference between output and input lengths (see Definition 2.4).

[4] A function $f(x)$ is regular if the every image has the same number (say α) of preimages, and it is known- (resp., unknown-) regular if α is efficiently computable (resp., inefficient to approximate) from the security parameter.

see length $O(n^2/\log n)$ and can be further pushed to $O(n \cdot \log n)$ using derandomization techniques (e.g., Nisan's bounded-space generator [18]). The randomized iterate matches the lower bound on the number of OWF calls[5], but it remains open if any efficient construction can achieve linear seed length and $O(n/\log n)$ OWF calls simultaneously.

Our Contributions. We contribute an alternative proof for the folklore construction of PRGs from known-regular OWFs via the notion of unpredictability pseudo-entropy, which significantly simplifies and tightens the proofs in [5]. We also give a new construction from any unknown-regular one-way function using seed length $\tilde{O}(n)$ and making $\tilde{O}(n/\log n)$ calls. Here both parameters are optimal up to an arbitrarily close to constant factor, and thus improves the results of the randomized iterate [9].

PRGs from Known-Regular OWFs. We start by assuming a (t,ε)-OWF f (see Definition 2.2) with known regularity 2^k (i.e., every image has 2^k preimages under f). The first key observation is that for uniform X (over $\{0,1\}^n$) we have X given $f(X)$ has $n + \log(1/\varepsilon)$ bits of pseudo-entropy (defined by the game below and formally in Definition 2.5). That is, no adversary A of running time t can win the following game against the challenger C with probability greater than $(2^{-k} \cdot \varepsilon)$. The

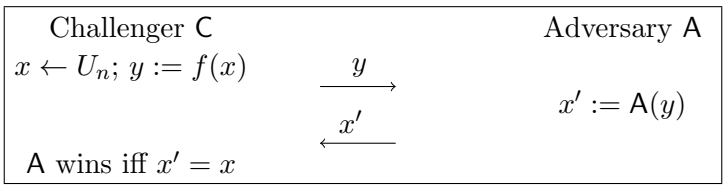

Figure 1: The interactive game between A and C that defines unpredictability pseudo-entropy, where $x \leftarrow U_n$ denotes sampling a random $x \in \{0,1\}^n$.

rationale is that conditioned on any $f(X) = y$ random variable X is uniformly distributed on set $f^{-1}(y) \stackrel{\mathsf{def}}{=} \{x : f(x) = y\}$ of size 2^k, and thus even if any deterministic (or probabilistic) A recovers a $x' \in f^{-1}(y)$, the probability that $X = x'$ is only 2^{-k}. Therefore, we obtain the following folklore construction (explicit in [5] and implicit in many HILL-style generators) using three extractions along with a three-line proof. In addition to simplicity, our technique can also be used to refine and tighten the proofs given in [5] (see Section 3.2 for details).

1. $f(X)$ has min-entropy $n - k$, and thus we can extract nearly $n - k$ statistically random bits.
2. X has min-entropy k given any $y = f(X)$, so we can extract another k statistically random bits.
3. The second extraction only reduces the unpredictability pseudo-entropy of X given $f(X)$ by no more than k (i.e., $\log(1/\varepsilon)$ bits remaining by the entropy chain rule), and hence we use Goldreich-Levin hardcore functions [8] to extract $O(\log(1/\varepsilon))$ bits.

The above construction is optimal (in seed length and the number of OWF calls), but requires the knowledge about parameter ε, more precisely, we need ε to decide entropy loss d such that the first extraction outputs $n - k - d$ bits with statistical error bounded by $2^{-d/2}$ (by the Leftover Hash Lemma [12]) and let the third extraction output more than d bits to achieve a positive stretch. It is unknown how to remove the dependency on ε for free (see also the discussions in [5]). Fortunately, there is a known repetition trick to solve the problem using seed length $\tilde{O}(n)$ and $\tilde{O}(1)$ OWF calls, where notation $\tilde{O}$ omits a factor of $q \in \omega(1)$ (i.e. q can be any factor arbitrarily close to constant such as $\log\log\log n$).

PRGs from Unknown-Regular OWFs. We also give a new construction oblivious of the regularity of f. The key idea is to transform any unknown regular OWF into another known regular OWF (over a

[5] As explicitly stated in [14], the lower bound of $\Omega(n/\log n)$ calls also applies to unknown regular OWFs.

special domain). That is, for a (length-preserving) unknown-regular (t,ε)-OWF $f:\{0,1\}^n\to\mathcal{Y}$ where $\mathcal{Y}\subseteq\{0,1\}^n$ denotes the range of f, define function $\bar{f}:\mathcal{Y}\times\{0,1\}^n\to\mathcal{Y}$ as $\bar{f}(y,r)\stackrel{\mathsf{def}}{=}f(y\oplus r)$ where "$\oplus$" denotes bitwise XOR. It is not hard to see that $\bar{f}$ has regularity 2^n (regardless of the regularity of f) and it preserves the hardness of f. Similar to that observed in the 1st construction, $\bar{f}(Y,R)$ hides $n+\log(1/\varepsilon)$ bits of pseudo-entropy about (Y,R), and thus we can extract $n+O(\log(1/\varepsilon))$ pseudorandom bits, namely, we get a PRG $\bar{g}$ that maps random elements over $\mathcal{Y}\times\{0,1\}^n$ to pseudorandom ones over $\mathcal{Y}\times\{0,1\}^{n+O(\log(1/\varepsilon))}$. Nevertheless, to use $\bar{g}$ we need to efficiently sample from $U_{\mathcal{Y}}=f(U_n)$ (i.e. uniform distribution over $\mathcal{Y}$), which costs n random bits despite that the entropy of $U_{\mathcal{Y}}$ may be far less than n. Therefore, the construction invests n bits (to sample a random $y\in\mathcal{Y}$) at initialization, runs $\bar{g}$ in iterations, and outputs $O(\log(1/\varepsilon))$ bits per iteration. The stretch becomes positive after $O(n/\log(1/\varepsilon))$ iterations, which matches the lower bounds of [14]. The seed length remains of order $\Theta(n)$ by reusing the coins for universal hash and G-L functions at every iteration, thanks to the hybrid argument. Similarly, in case that ε is unknown, we pay a penalty factor $\tilde{O}(1)$ for using the repetition trick. That is, we construct a PRG from any unknown-regular OWF using seed length $\tilde{O}(n)$ and $\tilde{O}(n/\log n)$ OWF calls.

2 Preliminaries

Notations and definitions. We use capital letters (e.g. X, Y, A) for random variables, standard letters (e.g. x, y, a) for values, and calligraphic letters (e.g. $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{S}$) for sets. $|\mathcal{S}|$ denotes the cardinality of set $\mathcal{S}$. For function f, we let $f(\mathcal{X})\stackrel{\mathsf{def}}{=}\{f(x):x\in\mathcal{X}\}$ be the set of images that are mapped from $\mathcal{X}$ under f, and denote by $f^{-1}(y)$ the set of y's preimages under f, i.e. $f^{-1}(y)\stackrel{\mathsf{def}}{=}\{x:f(x)=y\}$. We say that distribution X is flat if it is uniformly distributed over some set $\mathcal{X}$. We use $s\leftarrow S$ to denote sampling an element s according to distribution S, and let $s\leftarrow\mathcal{S}$ denote sampling s uniformly from set $\mathcal{S}$, and $y:=f(x)$ denote value assignment. We use U_n to denote the flat distribution over $\{0,1\}^n$ independent of the rest random variables in consideration, and let $f(U_n)$ be the distribution induced by applying function f to U_n. We use $\mathsf{CP}(X)$ to denote the collision probability of X, i.e., $\mathsf{CP}(X)\stackrel{\text{def}}{=}\sum_x\Pr[X=x]^2$, and collision entropy $\mathbf{H}_2(X)\stackrel{\text{def}}{=}-\log\mathsf{CP}(X)\geq\mathbf{H}_\infty(X)$. We also define average (aka conditional) collision entropy and average min-entropy of a random variable X conditioned on another random variable Z by

$$\mathbf{H}_2(X|Z)\stackrel{\text{def}}{=}-\log\big(\,\mathbb{E}_{z\leftarrow Z}\big[\,\textstyle\sum_x\Pr[X=x|Z=z]^2\,\big]\,\big)$$
$$\mathbf{H}_\infty(X|Z)\stackrel{\text{def}}{=}-\log\big(\,\mathbb{E}_{z\leftarrow Z}\big[\,\max_x\Pr[X=x|Z=z]\,\big]\,\big)$$

An entropy source refers to a random variable that has some non-trivial amount of entropy. A function $\mu:\mathbb{N}\to[0,1]$ is negligible if for every polynomial poly we have $\mu(n)<1/\mathsf{poly}(n)$ holds for all sufficiently large n. We define the *computational distance* between distribution ensembles $X\stackrel{\mathsf{def}}{=}\{X_n\}_{n\in\mathbb{N}}$ and $Y\stackrel{\mathsf{def}}{=}\{Y_n\}_{n\in\mathbb{N}}$ as follows: we say that X and Y are $(t(n),\varepsilon(n))$-close, denoted by $\mathsf{CD}_{t(n)}(X,Y)\leq\varepsilon(n)$, if for every probabilistic distinguisher D of running time up to $t(n)$ it holds that

$$|\ \Pr[\mathsf{D}(1^n,X_n)=1]-\Pr[\mathsf{D}(1^n,Y_n)=1]\ |\leq\varepsilon(n)\ .$$

The *statistical distance* between X and Y, denoted by $\mathsf{SD}(X,Y)$, is defined by

$$\mathsf{SD}(X,Y)\stackrel{\mathsf{def}}{=}\frac{1}{2}\sum_x|\Pr[X=x]-\Pr[Y=x]|\ =\ \ \mathsf{CD}_\infty(X,Y)$$

We use $\mathsf{SD}(X,Y|Z)$ (resp. $\mathsf{CD}_t(X,Y|Z)$) as shorthand for $\mathsf{SD}((X,Z),(Y,Z))$ (resp. $\mathsf{CD}_t((X,Z),(Y,Z))$).

SIMPLIFYING ASSUMPTIONS AND NOTATIONS. To simplify the presentation, we make the following assumptions without loss of generality. It is folklore that one-way functions can be assume to be length-preserving (see [10] for formal proofs). Throughout, most parameters are functions of the security parameter n (e.g., $t(n)$, $\varepsilon(n)$, $\alpha(n)$) and we often omit n when clear from the context (e.g., t, ε, α). Parameters (e.g. ε, α) are said to be known if they are known to be polynomial-time computable from n. By notation $f : \{0,1\}^n \to \{0,1\}^l$ we refer to the ensemble of functions $\{f_n : \{0,1\}^n \to \{0,1\}^{l(n)}\}_{n\in\mathbb{N}}$. As slight abuse of notion, poly might be referring to the set of all polynomials or a certain polynomial, and h might be either a function or its description, which will be clear from the context.

Definition 2.1 (universal hash functions [3]) *A family of functions $\mathcal{H} \stackrel{\mathsf{def}}{=} \{h : \{0,1\}^n \to \{0,1\}^l\}$ is called a* universal hash family*, if for any $x_1 \neq x_2 \in \{0,1\}^n$ we have $\Pr_{h\leftarrow\mathcal{H}}[h(x_1) = h(x_2)] \leq 2^{-l}$.*

Definition 2.2 (one-way functions) *A function $f : \{0,1\}^n \to \{0,1\}^{l(n)}$ is $(t(n),\varepsilon(n))$-one-way if f is polynomial-time computable and for any probabilistic algorithm A of running time $t(n)$*

$$\Pr_{y\leftarrow f(U_n)}[\mathsf{A}(1^n, y)\in f^{-1}(y)] \;\leq\; \varepsilon(n).$$

For $\varepsilon(n) = 1/t(n)$, we simply say that f is $\varepsilon(n)$-hard. f is a one-way function if it is $\varepsilon(n)$-hard for some negligible function $\varepsilon(n)$.

Definition 2.3 (regular functions) *A function f is α-regular if there exists an integer function α, called the regularity function, such that for every $n \in \mathbb{N}$ and $x \in \{0,1\}^n$ we have*

$$|f^{-1}(f(x))| = \alpha(n).$$

In particular, f is known-regular if α is polynomial-time computable, or is unknown-regular otherwise. Further, f is a (known-/unknown-) regular OWF if f is a OWF with (known/unknown) regularity.

Definition 2.4 (pseudorandom generators[2, 23]) *A function $g : \{0,1\}^n \to \{0,1\}^{l(n)}$ $(l(n) > n)$ is a $(t(n),\varepsilon(n))$-secure PRG if g is polynomial-time computable and*

$$\mathsf{CD}_{t(n)}(\; g(1^n, U_n) \;,\; U_{l(n)} \;) \;\leq\; \varepsilon(n).$$

where $(l(n) - n)$ is the stretch of g, and we often omit 1^n (security parameter in unary) from g's parameter list. We say that g is a pseudorandom generator if both $1/t(n)$ and $\varepsilon(n)$ are negligible.

Definition 2.5 (unpredictability pseudo-entropy[1, 15]) *Let $(X, Z) \stackrel{\mathsf{def}}{=} \{(X_n, Z_n)\}_{n\in\mathbb{N}}$ be a joint distribution ensemble, we say that X has $k(n)$ bits of pseudo-entropy conditioned on Z with respect to all $t(n)$-time adversaries, denoted by $\mathbf{H}_{t(n)}(X|Z) \geq k(n)$, if for any $n \in \mathbb{N}$ and any probabilistic adversary A of running time $t(n)$*

$$\Pr_{(x,z)\leftarrow(X_n,Z_n)}[\mathsf{A}(1^n, z) = x] \;\leq\; 2^{-k(n)}$$

Alternatively, we say that X is $2^{-k(n)}$-hard to predict given Z for all $t(n)$-time adversaries.

Unpredictability pseudo-entropy can be seen as a relaxed form of min-entropy by weakening adversary's running time from unbounded to parameter $t(n)$, which (presumably) characterizes the class of practical adversaries we care about. Note that the notion seems only meaningful in its conditional form as otherwise (when Z is empty) non-uniform attackers can simply hardwire the best guess about X, and thus $\mathbf{H}_{t(n)}$ collapses to $\mathbf{H}_\infty$. Let us mention the unpredictability pseudo-entropy is different from (and in fact, strictly weaker than [1, 15]) the HILL pseudo-entropy [12], which is another relaxed notion of min-entropy by considering its computationally indistinguishable analogues.

3 Pseudorandom Generators from Regular One-Way Functions

3.1 Technical Tools

The first technical tool we use is the leftover hash lemma. Informally, it states that when applying a random universal hash function to min-entropy (or Rényi entropy) source, one obtain random strings that are statistical close to uniform even conditioned on the description of hash function. The objects were later formalized as randomness extractors [19]. Universal hash functions are also good condensers (whose outputs have nearly maximal entropy) for a wider range of parameters than extractors.

Lemma 3.1 (Leftover Hash Lemma [12]) *For any integers $d<k\leq n$, there exists an (efficiently computable) universal hash function family $\mathcal{H} \stackrel{\mathsf{def}}{=} \{h : \{0,1\}^n \to \{0,1\}^{k-d}\}$ such that for any joint distribution (X,Z) where $X \in \{0,1\}^n$ and $\mathbf{H}_2(X|Z) \geq k$, we have*

$$\mathsf{SD}(H(X),\ U_{k-d} \mid H, Z)\ \leq\ 2^{-\frac{d}{2}}$$

where H is uniformly distributed over the members of $\mathcal{H}$, the description size of H is called seed length, and d is called entropy loss*, i.e., the difference between the entropy of X (given Z) and the number of bits that were extracted from X.*

Lemma 3.2 (Condensers from hash functions) *Let $\mathcal{H} \stackrel{\mathsf{def}}{=} \{h : \{0,1\}^n \to \{0,1\}^k\}$ be any universal hash function family and let (X,Z) be any random variable with $X \in \{0,1\}^n$ and $\mathbf{H}_2(X|Z) \geq k$. Then, for H uniform distributed over $\mathcal{H}$ we have $\mathbf{H}_2(H(X) \mid H, Z) \geq k-1$.*

Proof. Let X_1 and X_2 be i.i.d. to $X \mid Z=z$ (i.e. X conditioned on $Z=z$).

$$\begin{aligned} & 2^{-\mathbf{H}_2(H(X)|H,Z)} = \mathbb{E}_{h\leftarrow H, z\leftarrow Z}\left[\ \Pr[H(X_1)=H(X_2)|H=h,Z=z]\ \right] \\ \leq\ & \mathbb{E}_{z\leftarrow Z}\left[\ \Pr[X_1=X_2|Z=z]\ \right] + \mathbb{E}_{z\leftarrow Z}\left[\ \Pr[\ H(X_1)=H(X_2)\ \mid\ X_1\neq X_2, Z=z]\ \right] \\ \leq\ & 2^{-k}+2^{-k}\ =\ 2^{-(k-1)}\ . \end{aligned}$$

□

We refer to [20, 4, 16] for extremely efficient constructions of universal hash functions with short description (of length $\Theta(n)$), such as multiplications between matrices and vectors, or over finite fields.

Reconstructive extractors. We will also need objects that extract pseudorandomness from unpredictability pseudo-entropy sources. Unfortunately, the leftover hash lemma (and randomness extractors [19] in general) does not serve the purpose. Goldreich and Levin [8] showed that the inner product function is a reconstructive bit-extractor for unpredictability pseudo-entropy sources. Further, there are two ways to extend the inner product to multiple-bit extractors: (1) multiplication with a random matrix of length $O(n^2)$ and extracts almost all entropy (by a hybrid argument); (2) multiplication with a random Toeplitz matrix of length $\Theta(n)$ and extracts $O(\log(1/\varepsilon))$ bits (due to Vazirani's XOR lemma [22, 8]). We will use the latter multi-bit variant (as stated below) to keep the seed length linear. Interestingly, the Toeplitz matrix based functions also constitute pairwise independent and universal hash function families.

Theorem 3.1 (Goldreich-Levin [8]) *For distribution ensemble (X,Y) where X is uniform over $\{0,1\}^n$, and for any integer $m \leq n$, there exists[6] a function family $\mathcal{H}_C \stackrel{\mathsf{def}}{=} \{h_c : \{0,1\}^n \to \{0,1\}^m\}$ of description size $\Theta(n)$, such that*

[6] For example (see [8]), we can use an $m\times n$ Toeplitz matrix $a_{m,n}$ to describe the family of functions, i.e., $\mathcal{H}_C \stackrel{\mathsf{def}}{=} \{h_c(x) \stackrel{\mathsf{def}}{=} a_{m,n}\cdot x,\ \text{where } x\in\{0,1\}^n, a_{m,n}\in\{0,1\}^{m+n-1}\}$.

- *If $Y = f(X)$ for any (t,ε)-OWF f, then we have*

$$\mathsf{CD}_{t'}(\ H_C(X)\ ,\ U_m\ \mid\ Y, H_C)\ \in\ O(2^m \cdot \varepsilon)\ . \tag{1}$$

- *If X is ε-hard to predict given Y for all t-time adversaries, i.e. $\mathbf{H}_t(X|Y) \geq \log(1/\varepsilon)$, then we have*

$$\mathsf{CD}_{t'}(\ H_C(X)\ ,\ U_m\ \mid\ Y, H_C)\ \in\ O(2^m \cdot (n \cdot \varepsilon)^{\frac{1}{3}})\ . \tag{2}$$

where $t' = t \cdot (\varepsilon/n)^{O(1)}$ and function H_C is uniformly distributed over the members of $\mathcal{H}_C$.

Remark 3.1 *To see the difference between the different versions above, consider the interactive game in Figure 1, where by unpredictability A's prediction is successful only if $x = x'$, but in contrast A inverts OWF f as long as he finds any x' satisfying $f(x') = y$. Recall that the proof of the theorem can be seen an efficient local list decoding procedure for the Hadamard code, where in the former case the decoder returns a random member from the candidate list while in the latter case it goes through all candidates and outputs the one x' satisfying $f(x') = y$ (if exists). We refer to Goldreich's exposition [6] for further details.*

We recall two folklore facts below, namely the chain rule of unpredictability (pseudo-)entropy and the replacement inequality. Intuitively, any leakage $Y \in \{0,1\}^l$ decreases the unpredictability about secret X by a factor of no more than 2^l, which can be seen by a simple reduction (e.g., by replacing Y with a random string). The replacement inequality states that any information that is (efficiently) computable from the knowledge of the adversary does not help further reduce the unpredictability (pseudo-)entropy of the secret in consideration.

Fact 3.1 (chain rule of entropies) *For any joint distribution (X,Y,Z) where $Y \in \{0,1\}^l$, we have*

$$\mathbf{H}_\infty(X|Y,Z) \geq \mathbf{H}_\infty(X|Z) - l\ ,$$

$$\mathbf{H}_{t'}(X|Y,Z) \geq \mathbf{H}_t(X|Z) - l\ ,$$

where $t' \approx t$.

Fact 3.2 (replacement inequalities) *For any joint distribution (X,Y,Z) and any t_h-time computable function $h : \mathcal{Y} \to \{0,1\}^*$, we have*

$$\mathbf{H}_\infty(X|h(Y),h,Z) \geq \mathbf{H}_\infty(X|Y,Z)\ ,$$

$$\mathbf{H}_{t-t_h}(X|h(Y),h,Z) \geq \mathbf{H}_t(X|Y,Z)\ .$$

3.2 PRGs from OWFs with Known Regularity and Hardness

We state our motivating observation as the lemma below.

Lemma 3.3 *Let $f : \mathcal{X} \to \mathcal{Y}$ be a 2^k-regular (t,ε)-OWF. Then, we have*

$$\mathbf{H}_t(X\ \mid\ f(X)) \geq k + \log(1/\varepsilon)\ , \tag{3}$$

where X is uniform over $\mathcal{X}$.

Proof. The (t,ε)-one-wayness of f guarantees that for any deterministic adversary A of running time t

$$\Pr_{x\leftarrow\mathcal{X},y:=f(x)}[\ \mathsf{A}(y)\in f^{-1}(y)\ ]\le\varepsilon$$

which in turn implies (as conditioned on $f(X)=y$, X is uniform over $f^{-1}(y)$ of size 2^k):

$$\Pr_{x\leftarrow\mathcal{X},y:=f(x)}[\ \mathsf{A}(y)=x\ ]\le 2^{-k}\cdot\varepsilon$$

which is essentially Equation (3) by taking a negative logarithm. Note that the above argument extends to probabilistic t-time A as well, by considering $\mathsf{A}(y;r)$ on every fixing of his random coin r. □

The Construction for Known α and ε. For joint distribution $(X,f(X))$, the proposed PRG uses universal hash functions h_1, h_2 to extract nearly (up to entropy loss) $n-k$ and k bits from $f(X)$ and X respectively, and employs G-L function h_c to extract $\Theta(\log(1/\varepsilon n))$ bits of pseudo-entropy from X. For convenience, we assume without loss of generality that the regularity is a power of two, i.e., $\alpha=2^k$.

Theorem 3.2 (Preliminary Construction based on Known Regularity and Hardness) *Let $f:\{0,1\}^n\to\{0,1\}^n$ be a known 2^k-regular length-preserving (t,ε)-OWF, let d, s be any integer functions satisfying $9d+6s=2\log(1/\varepsilon n)$, let $\mathcal{H}_1\stackrel{\mathsf{def}}{=}\{h_1:\{0,1\}^n\to\{0,1\}^{n-k-d}\}$, $\mathcal{H}_2\stackrel{\mathsf{def}}{=}\{h_2:\{0,1\}^n\to\{0,1\}^k\}$ be universal hash function families, let $\mathcal{H}_C\stackrel{\mathsf{def}}{=}\{h_c:\{0,1\}^n\to\{0,1\}^{d+s}\}$ be a Goldreich-Levin function family, and let g be*

$$\begin{aligned}g:\{0,1\}^n\times\mathcal{H}_1\times\mathcal{H}_2\times\mathcal{H}_C&\to\{0,1\}^{n-k-d}\times\{0,1\}^k\times\{0,1\}^{d+s}\times\mathcal{H}_1\times\mathcal{H}_2\times\mathcal{H}_C\\(x,h_1,h_2,h_c)&\mapsto(h_1(f(x)),h_2(x),h_c(x),h_1,h_2,h_c)\end{aligned}$$

where $x\in\{0,1\}^n$, $h_1\in\mathcal{H}_2$, $h_2\in\mathcal{H}_2$, $h_c\in\mathcal{H}_C$. Then, g is a $(\ t\cdot(\varepsilon/n)^{O(1)},\ O((2^{3s}\cdot\varepsilon\cdot n)^{\frac{1}{9}})\ )$-secure PRG with stretch s.
We deal with the situation where $n-k-d\le 0$ by letting h_1 output a dummy string. Another special case $k=0$ (i.e., f is a OWP) is handled by letting h_1 and h_2 output the identity and dummy strings respectively.

Proof. The entropy conditions for the (pseudo)-randomness extractions are guaranteed by Lemma 3.4. We have by Equation (4), Equation (5) and the leftover hash lemma that the first $n-d$ bits extracted are statistically random, namely,

$$\begin{aligned}&\mathsf{SD}(\ (H_1(f(X)),H_2(X)),\ U_{n-d}\ \mid\ H_1,H_2)\\\le\ &\mathsf{SD}(\ H_1(f(X)),\ U_{n-k-d}\ \mid\ H_1)+\mathsf{SD}(\ H_2(X),\ U_k\ \mid\ H_1(f(X)),H_1,H_2)\\\le\ &2\cdot 2^{-\frac{d}{2}}=2\cdot 2^{\frac{s}{3}+\frac{1}{9}\log(\varepsilon n)}=\ O((2^{3s}\cdot\varepsilon\cdot n)^{\frac{1}{9}})\end{aligned}$$

Next, as stated in Equation (6), conditioned on the prefix of $n-d$ random bits (and the seeds used), X remains $(t-n^{O(1)},\varepsilon)$-hard to predict, and thus by Goldreich-Levin (Theorem 3.1)

$$\mathsf{CD}_{t'}(\ H_C(X)\ ,\ U_{d+s}\ \mid\ H_1(f(X)),H_2(X),H_1,H_2,H_C)=O(2^{d+s}\cdot(n\cdot\varepsilon)^{\frac{1}{3}})=O(2^{-\frac{d}{2}})=O((2^{3s}\cdot\varepsilon\cdot n)^{\frac{1}{9}})$$

holds for $t'=t\cdot(\varepsilon/n)^{O(1)}$. The conclusion follows by a triangle inequality. □

Lemma 3.4 (Entropy conditions) *Let f, $\mathcal{H}_1$, $\mathcal{H}_2$ be defined as in Theorem 3.2, we have*

$$\mathbf{H}_\infty(f(X))=n-k\ ,\tag{4}$$

$$\mathbf{H}_\infty(X\mid h_1(f(X)),h_1)\ge\mathbf{H}_\infty(X)-(n-k-d)\ =\ k+d\ ,\tag{5}$$

$$\mathbf{H}_{t-n^{O(1)}}(X\mid h_1(f(X)),h_2(X),h_1,h_2)\ \ge\ \mathbf{H}_t(X\mid f(X),h_2(X),h_2)\ \ge\ \log(1/\varepsilon)\ .\tag{6}$$

hold for every $h_1\in\mathcal{H}_1$, $h_2\in\mathcal{H}_2$, and X uniform over $\{0,1\}^n$.

Proof. Equation (4) follows from the regularity of f, i.e., every $y = f(x)$ has 2^k preimages, and thus $f(X)$ is uniformly distributed over a set of size 2^{n-k}. Equation (5) is due to the chain rule of min-entropy (see Fact 3.1). The first inequality of Equation (6) is the replacement inequality (see Fact 3.2), and the second one is obtained by applying the chain rule of unpredictability entropy to Equation (3), i.e., $\mathbf{H}_t(X \mid f(X), h_2(X), h_2) \geq \mathbf{H}_t(X|f(X)) - k = \log(1/\varepsilon)$. □

Therefore, we already complete the proof for the PRG with linear seed length by doing a single call to any 2^k-regular ε-hard OWF provided that ε and k are known. We provide an alternative (and simpler) proof to that given by Goldreich [5] for essentially the same construction via unpredictability pseudo-entropy.

On Tightening Security Bounds. Concretely, if the underlying OWF is $n^{-\log n}$- (resp., $2^{-\frac{n}{3}}$-) hard, then the outputs of the resulting PRG will be nearly $n^{-\frac{\log n}{9}}$- (resp., $2^{-\frac{n}{27}}$-) close to uniform (with respect to reasonably weakened adversaries than counterparts of the OWF). The main lossy step in the reduction is that we considered function $f'(x, h_2) \stackrel{\mathsf{def}}{=} (f(x), h_2(x), h_2)$, where by Equation (6) X is (ε, t)-hard to predict given $f'(X)$ and thus we directly applied Equation (2) to get the inferior bounds. However, a closer look at f' suggests that it is almost 1-to-1, which implies that f' is a OWF (stated as in Lemma 3.5), which allows us to use the tight version of Goldreich-Levin Theorem (see Equation (1)). This is actually the approach taken by [5], where however f' was only shown to be roughly $\varepsilon^{1/5}$-hard (by checking the proof of [5, Prop 3.5.9]). We give a refined analysis below to get the tighter $\sqrt{\varepsilon}$-hardness of f', and this eventually leads to the improved construction as in Theorem 3.3.

Lemma 3.5 *Let f and $\mathcal{H}_2$ be as defined in Theorem 3.2, then function $f'(x, h_2) \stackrel{\mathsf{def}}{=} (f(x), h_2(x), h_2)$ is a $(t, 3\sqrt{\varepsilon})$ one-way function.*

Proof. Suppose for contradiction there exists A of running time t such that

$$\Pr\left[\mathsf{A}(f'(X, H_2)) \in f'^{-1}(f'(X, H_2))\right] > 3\sqrt{\varepsilon}$$

Recall that $f(X)$ has min-entropy $n - k$ and conditioned on any $y = f(X)$ X has min-entropy k, and thus by the condensing property of universal hashing (see Lemma 3.2) we have $\mathsf{CP}(f(X), H_2(X) \mid H_2) \leq 2^{-(n-1)}$ and it follows from Claim 3.1 (setting $a = 2^{-n}/\sqrt{\varepsilon}$, $X_1 = (f(X), H_2(X))$, $Z_1 = H_2$) that $f'(X, H_2)$ hits set S (defined below) with negligible probability, i.e., $\Pr[f'(X, H_2) \in \mathcal{S}] \leq 2\sqrt{\varepsilon}$ where

$$\begin{aligned} \mathcal{S} &\stackrel{\mathsf{def}}{=} \{(y, w, h_2) : \Pr[(f(X), h_2(X)) = (y, w) \mid H_2 = h_2] \geq 2^{-n}/\sqrt{\varepsilon}\} \\ &= \{(y, w, h_2) : |f'^{-1}(y, w, h_2)| \geq 1/\sqrt{\varepsilon}\} \ . \end{aligned}$$

Then, let $\mathcal{E}$ be the event that A inverts f' on any image whose preimage size is bounded by $1/\sqrt{\varepsilon}$, i.e., $\mathcal{E} \stackrel{\mathsf{def}}{=} \mathsf{A}(f'(X, H_2)) \in f'^{-1}(f'(X, H_2)) \ \wedge \ f'(X, H_2) \notin \mathcal{S}$

$$\begin{aligned} \Pr\left[\mathsf{A}(f'(X, H_2)) = X\right] &\geq \Pr[\mathcal{E}] \cdot \Pr[\mathsf{A}(f'(X, H_2)) = X \mid \mathcal{E}] \\ &> (3\sqrt{\varepsilon} - 2\sqrt{\varepsilon}) \cdot \left(\frac{1}{1/\sqrt{\varepsilon}}\right) = \varepsilon \ , \end{aligned}$$

where the probability of hard-to-invertness is related to unpredictability by the maximal preimage size. The conclusion follows by reaching a contradiction to the (t, ε)-unpredictability of X given $f'(X, H_2)$ (as stated in Equation (6)). □

Claim 3.1 *Let (X_1, Z_1) be a random variable, for $a > 0$ define $\mathcal{S}_a \stackrel{\mathsf{def}}{=} \{(x, z) : \Pr[X_1 = x | Z_1 = z] \geq a\}$, it holds that $\Pr[(X_1, Z_1) \in \mathcal{S}_a] \leq \mathsf{CP}(X_1|Z_1)/a$.*

Proof. The proof is a typical Markov type argument.

$$\begin{aligned}
\mathsf{CP}(X_1|Z_1) &= \mathbb{E}_{z\leftarrow Z_1}\left[\sum_x \Pr[X_1=x|Z_1=z]^2\right] = \sum_{(x,z)} \Pr[(X_1,Z_1)=(x,z)]\cdot\Pr[X_1=x|Z_1=z] \\
&\geq \sum_{(x,z)\in\mathcal{S}_a} \Pr[(X_1,Z_1)=(x,z)]\cdot\Pr[X_1=x|Z_1=z] \\
&\geq a\cdot\Pr[(X_1,Z_1)\in\mathcal{S}_a] \ .
\end{aligned}$$

□

Theorem 3.3 (Improved Construction based on Known Regularity and Hardness) *For the same f, g, $\mathcal{H}_1$, $\mathcal{H}_2$, $\mathcal{H}_C$ as assumed in Theorem 3.2 except that d and s satisfy $3d+2s=\log(1/\varepsilon)$, we have that g is a $(\ t\cdot(\varepsilon/n)^{O(1)},\ O((2^{2s}\cdot\varepsilon)^{1/6})$-secure PRG with stretch s.*

Proof sketch. The proof is similar to Theorem 3.2. The first $n-d$ bits extracted are $2^{-d/2}$-statistically random, conditioned on which the next $d+s$ bits are $O(2^{d+s}\sqrt{\varepsilon})$-computationally random. It follows that the bound is $2^{-d/2}+O(2^{d+s}\sqrt{\varepsilon})=O(2^{-d/2})=O((2^{2s}\cdot\varepsilon)^{1/6})$. □

Three Extractions are Necessary. We argue that three extractions (using h_1, h_2 and h_c) seem necessary. One might think that the first two extractions (using h_1 and h_2) can be merged using a single universal hash function (that applies to the source $(X,f(X))$ and outputs $n-d$ bits). However, by doing so we cannot ensure the entropy condition (see Equation (6)) for the third extraction (using h_c). From another perspective, the merge would remove the dependency on the regularity and thus result in a generic construction that does a single call to any unknown regular OWFs, which is a contradiction to [14]. Furthermore, it seems necessary to extract from X at least twice, namely, using h_2 and h_c to get statistically and computationally random bits respectively.

3.3 PRGs from Any Known Regular OWFs: Removing the Dependency on ε

The parameterization of the aforementioned construction depends on ε, but sometimes ε is unknown or not polynomial-time computable. It is thus more desirable to have a construction based on any known-regular OWF regardless of parameter ε (as long as it is negligible). We observe that by setting entropy loss to zero (in which case hash functions are condensers) and letting G-L functions extract $O(\log n)$ bits the resulting generator is a generic (i.e. without relying on ε) pseudo-entropy generator (PEG) with a (collision) entropy stretch of $O(\log n)$ bits. Note however the output of the PEG is not indistinguishable from uniform but from some high collision entropy sources (with small constant entropy deficiency), which implies a PRG by running $q\in\omega(1)$ copies of the PEG and doing a single extraction from the concatenated outputs.

Definition 3.1 (pseudoentropy generators) *A function $g:\{0,1\}^n\to\{0,1\}^{l+e}$ $(l>n)$ is a (t,ε) $\mathbf{H}_2$-pseudoentropy generator (PEG) if g is polynomial-time computable and there exists a random variable $Y\in\{0,1\}^{l+e}$ with $\mathbf{H}_2(Y)\geq l$*

$$\mathsf{CD}_t(\ g(U_n)\ ,\ Y\ )\ \leq\ \varepsilon.$$

where $(l-n)$ is the stretch of g, and e is the entropy deficiency. We say that g is an $\mathbf{H}_2$-pseudoentropy generator if $1/\varepsilon$ and t are both super-polynomial.

Theorem 3.4 (PEGs from any known-regular OWFs) *For the same f, g, $\mathcal{H}_1$, $\mathcal{H}_2$, $\mathcal{H}_C$ as assumed in Theorem 3.2 except that $d=0$ and $s=2\log n+2$, we have that if f is a one-way function then g is a $\mathbf{H}_2$-pseudoentropy generator with stretch $2\log n$ and entropy deficiency 2.*

Proof sketch. It is not hard to see (using Lemma 3.2) that for $d = 0$ we have

$$\begin{aligned} & 2^{-\mathbf{H}_2(H_1(f(X)),H_2(X))} = \mathsf{CP}(H_1(f(X)), H_2(X)) \\ \leq & \Pr_{X_1,X_2\leftarrow U_n}[H_1(f(X_1)) = H_1(f(X_2))] \cdot \Pr[H_2(X_1) = H_2(X_2) \mid f(X_1) = f(X_2)\ ] \\ \leq & \ 2^{-(n-k-1)} \cdot 2^{-(k-1)} = 2^{-(n-2)}\ . \end{aligned}$$

And we have by Lemma 3.5 and Goldreich-Levin the $2\log n + 2$ hardcore bits are pseudo-random given $H_1(f(X))$ and $H_2(X)$, which completes the proof. □

Theorem 3.5 (PRGs from any known-regular OWFs) *For any known k, there exists a generic construction of pseudo-random generator with seed length $\tilde{O}(n)$ by making $\tilde{O}(1)$ calls to any (length-preserving) 2^k-regular one-way function.*

Proof sketch. The idea is to run $q \in \omega(1)$ independent copies of the PEGs as in Theorem 3.4 to get an entropy stretch of $2q\log n$ followed by a single randomness extraction with entropy loss $q\log n$. This yields a PRG with stretch $q\log n$ that is roughly $O(q{\cdot}n^2\sqrt{\varepsilon} + n^{-q})$ computationally indistinguishable from uniform randomness, where n^{-q} is negligible for any $q \in \omega(1)$. □

3.4 PRGs from Any Unknown Regular OWFs

The first attempt: a parallel construction. A straightforward way to adapt the construction to unknown regular OWFs is to pay a factor of $n/\log n$. That is, it is not hard to see the construction for known regularity $\alpha = 2^k$ remains secure even by using an approximated value $\tilde{\alpha} = 2^{\tilde{k}}$ with accuracy $|\tilde{k} - k| \leq \log n$. This immediately implies a parallel construction by running $n/\log n$ independent copies of our aforementioned construction, where each i^{th} copy assumes regularity $2^{i\cdot\log n}$. Therefore, at least one (unknown) copy will be a PRG and thus we simply XOR the outputs of all copies and produce it as the output. Unfortunately, similar to the HILL approach, the parallelism turns out an inherent barrier to linear seed length. We will avoid this route by giving a sequential construction.

Now we present the construction from any (length-preserving) unknown-regular OWF. We first transform it into a hardness-preserving equivalent with known regularity 2^n, as stated in Claim 3.2.

Claim 3.2 *For any length-preserving unknown-regular (t,ε)-OWF $f : \{0,1\}^n \to \{0,1\}^n$, define*

$$\begin{aligned} \bar{f} : \mathcal{Y} \times \{0,1\}^n &\to \mathcal{Y} \\ \bar{f}(y, r) &\stackrel{\mathsf{def}}{=} f(y \oplus r) \end{aligned} \tag{7}$$

where $\mathcal{Y} \stackrel{\mathsf{def}}{=} f(\{0,1\}^n) \subseteq \{0,1\}^n$, "$\oplus$" denotes bit-wise XOR. Then, $\bar{f}$ is a 2^n-regular $(t - O(n),\varepsilon)$-OWF.

Proof. On uniform (y,r) over $\mathcal{Y} \times \{0,1\}^n$, $y \oplus r$ is uniform over $\{0,1\}^n$. Thus, any algorithm inverts $\bar{f}$ to produce (y, r) with probability ε implies another algorithm that inverts f with the same probability by outputting $y \oplus r$. Let us assume that f is α-regular. Then, for any $y_1 = \bar{f}(y, r) = f(y{\oplus}r)$ we have $|f^{-1}(y_1)| = \alpha$, and for any $x \in f^{-1}(y_1)$ we have $|\{(y, r) \in \mathcal{Y} \times \{0,1\}^n : y \oplus r = x\}| = |\mathcal{Y}| = 2^n/\alpha$, which implies $|\bar{f}^{-1}(y_1)\}| = \alpha \cdot (2^n/\alpha) = 2^n$. □

Similarly to the known regular case, we first assume ε is known and then eliminate the dependency. Intuitively, the output of $\bar{f}$ hides n bits of min-entropy about its input (by the 2^n-regularity) plus another $\log(1/\varepsilon)$ bits of pseudo-entropy (due to the one-wayness), and thus one can extract $n + O(\log(1/\varepsilon))$ pseudorandom bits. This is formalized in Claim 3.3, where we build a generator $\bar{g}$ that expands random elements over $\mathcal{Y}{\times}\{0,1\}^n$ into pseudorandom ones over $\mathcal{Y}{\times}\{0,1\}^{n+O(\log(1/\varepsilon))}$. The proof of Claim 3.3 is

similar to that of Theorem 3.2, and we defer it to the appendix. Notice, however, generator $\bar{g}$ is NOT a practical PRG with positive stretch as the only black-box way to sample distribution $U_{\mathcal{Y}}$ is to compute $f(U_n)$, which costs n random bits (despite that $\mathbf{H}_\infty(U_{\mathcal{Y}})$ might be far less than n). Quite naturally and thanks to the hybrid argument, the our construction simply iterates $\bar{g}$, reuses the random seeds (in each iteration), and outputs $s = O(\log(1/\varepsilon))$ bits per iteration.

Claim 3.3 *Let f, $\bar{f}$ be defined as in Claim 3.2, for any integers d, s satisfying $7d + 6s = 2\log(1/\varepsilon n)$, let $\mathcal{H} \stackrel{\text{def}}{=} \{h : \{0,1\}^{2n} \to \{0,1\}^{n-d}\}$ be a universal hash function family, let $\mathcal{H}_C \stackrel{\text{def}}{=} \{h_c : \{0,1\}^{2n} \to \{0,1\}^{d+s}\}$ be a G-L function family, define $\bar{g}$ as*

$$\begin{aligned} \bar{g} : \mathcal{Y} \times \{0,1\}^n \times \mathcal{H} \times \mathcal{H}_C &\to \mathcal{Y} \times \{0,1\}^{n+s} \times \mathcal{H} \times \mathcal{H}_C \\ \bar{g}(y, r, h, h_c) &\stackrel{\text{def}}{=} (\ \bar{f}(y,r),\ (h(y,r), h_c(y,r)),\ h,\ h_c) \end{aligned} \tag{8}$$

Then, it holds that

$$\mathsf{CD}_{t\cdot(\varepsilon/n)^{O(1)}}(\ \bar{g}(Y,R,H,H_C)\ ,\ (U_{\mathcal{Y}}, U_{n+s}, H, H_C)\ ) \ =\ O((2^{3s}\cdot\varepsilon\cdot n)^{\frac{1}{7}})$$

where $U_{\mathcal{Y}} \stackrel{\text{def}}{=} f(U_n)$, (Y,R) is identically distributed to $U_{\mathcal{Y}} \times U_n$, and H, H_C are uniform over $\mathcal{H}$, $\mathcal{H}_C$ respectively.

Theorem 3.6 (PRGs from any unknown-regular OWFs with known hardness) *Let $f : \{0,1\}^n \to \{0,1\}^n$ be any (possibly unknown) regular length-preserving (t,ε)-OWF, define $\bar{f}$,$\bar{g}$, $\mathcal{H}$, $\mathcal{H}_C$, s as in Claim 3.3, and define g as*

$$\begin{aligned} g : \{0,1\}^n \times \{0,1\}^n \times \mathcal{H} \times \mathcal{H}_C &\to (\{0,1\}^s)^\ell \times \{0,1\}^n \times \mathcal{H} \times \mathcal{H}_C \\ g(x, r_0, h, h_c) &\stackrel{\text{def}}{=} (z_1, z_2, \ldots, z_\ell, r_\ell, h, h_c) \end{aligned}$$

where let $y_0 := f(x)$, and for $1 \le i \le \ell$ iteratively compute $(y_i, r_i, z_i, h, h_c) := \bar{g}(y_{i-1}, r_{i-1}, h, h_c)$, $h \in \mathcal{H}$, $h_c \in \mathcal{H}_C$, $(y_i, r_i) \in \mathcal{Y} \times \{0,1\}^n$ and $z_i \in \{0,1\}^s$. Then, for any $s \le \log(1/\varepsilon n)/3$, function g is a $(t\cdot(\varepsilon/n)^{O(1)} - \ell \cdot n^{O(1)},\ O(\ell\cdot(2^{3s}\cdot\varepsilon\cdot n)^{\frac{1}{7}}))$-secure PRG with stretch $\ell \cdot s - n$.

Proof. The proof follows from Claim 3.3 by a standard hybrid argument. □

Therefore, for any unknown-regular OWF with known hardness, we obtain a PRG with linear seed length, and by letting $s \in \Theta(\log(1/\varepsilon n))$ the number of calls $\ell \in \Theta(n/s) = \Theta(n/\log(1/\varepsilon n))$ matches the lower bound of [14]. This extends to the general case (where the hardness parameter is unknown) by repetition.

Theorem 3.7 (PRGs from any unknown-regular OWFs) *There exists a generic construction of pseudo-random generator with seed length $\tilde{O}(n)$ by making $\tilde{O}(n/\log n)$ calls to any (length-preserving) unknown-regular one-way function.*

Proof sketch. For any unknown-regular OWF f, define $\bar{g}$ as in Claim 3.3 except setting $d = 0$ and $s = 2\log n + 1$. It is not hard to see that the resulting $\bar{g}$ is a $\mathbf{H}_2$-pseudoentropy generator with stretch $2\log n$ and entropy deficiency 1 (proof similar to that in Theorem 3.4). Following the steps sketched in Theorem 3.5, for any $q \in \omega(1)$ run q independent copies of $\bar{g}$ followed by an extraction with entropy loss set to $q\log n$, we obtain a special pseudo-random generator $\bar{g}'$ over space

$$\bar{g}' : \mathcal{Y}^q \times \{0,1\}^{qn} \times \mathcal{H}^q \times \mathcal{H}_C^q \to \mathcal{Y}^q \times \{0,1\}^{q(n+\log n)} \times \mathcal{H}^q \times \mathcal{H}_C^q$$

Iterating $\bar{g}'$ for $\ell' = \lceil (qn+1)/q\log n \rceil \in O(n/\log n)$ rounds yields a PRG g' with stretch $s' \ge 1$, i.e.,

$$g' : \{0,1\}^{2qn} \times \mathcal{H}^q \times \mathcal{H}_C^q \to \{0,1\}^{2qn+s'} \times \mathcal{H}^q \times \mathcal{H}_C^q$$

which completes the proof. □

Acknowledgements: The author thanks Guang Yang and Colin Jia Zheng for very helpful comments. The author is also grateful to Thomas Holenstein for clarifying his lower bound results [14] at the very early stage of this work. This research work was supported by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61172085, 61061130540, 61073174, 61103221, 11061130539, 61021004 and 61133014.

References

[1] Boaz Barak, Ronen Shaltiel, and Avi Wigderson. Computational analogues of entropy. In *Proceedings of the 7th International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM 2003)*, pages 200–215, 2003.

[2] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo random bits. In *Proceedings of the 23rd IEEE Symposium on Foundation of Computer Science*, pages 112–117, 1982.

[3] J.L. Carter and M.N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18:143–154, 1979.

[4] Yevgeniy Dodis, Ariel Elbaz, Roberto Oliveira, and Ran Raz. Improved randomness extraction from two independent sources. In *APPROX-RANDOM*, pages 334–344, 2005.

[5] Oded Goldreich. *Foundations of Cryptography: Basic Tools*. Cambridge University Press, 2001.

[6] Oded Goldreich. Three XOR-lemmas - an exposition. In *Studies in Complexity and Cryptography*, pages 248–272. 2011.

[7] Oded Goldreich, Hugo Krawczyk, and Michael Luby. On the existence of pseudorandom generators. *SIAM Journal on Computing*, 22(6):1163–1175, 1993.

[8] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In D. S. Johnson, editor, *Proceedings of the Twenty First Annual ACM Symposium on Theory of Computing*, pages 25–32, Seattle, Washington, 15–17 May 1989.

[9] Iftach Haitner, Danny Harnik, and Omer Reingold. On the power of the randomized iterate. In *Proceedings of the 26th International Cryptology Conference (CRYPTO 2006)*, pages 22–40, 2006.

[10] Iftach Haitner, Danny Harnik, and Omer Reingold. On the power of the randomized iterate. *SIAM Journal on Computing*, 40(6):1486–1528, 2011.

[11] Iftach Haitner, Omer Reingold, and Salil P. Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. In *Proceedings of the 42nd ACM Symposium on the Theory of Computing*, pages 437–446, 2010.

[12] J. Håstad, R. Impagliazzo, L.A. Levin, and M. Luby. Construction of pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[13] Thomas Holenstein. Pseudorandom generators from one-way functions: A simple construction for any hardness. In *Proceedings of the 3rd Theory of Cryptography Conference (TCC 2006)*, 2006.

[14] Thomas Holenstein and Makrand Sinha. Constructing a pseudorandom generator requires an almost linear Number of calls. In *Proceedings of the 53rd IEEE Symposium on Foundation of Computer Science*, pages 698–707, 2012.

[15] Chun-Yuan Hsiao, Chi-Jen Lu, and Leonid Reyzin. Conditional computational entropy, or toward separating pseudoentropy from compressibility. In *Proceedings of the 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques (Eurocrypt 2007)*, pages 169–186, 2007.

[16] Chia-Jung Lee, Chi-Jen Lu, Shi-Chun Tsai, and Wen-Guey Tzeng. Extracting randomness from multiple independent sources. *IEEE Transactions on Information Theory*, 51(6):2224–2227, 2005.

[17] Leonid A. Levin. One-way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.

[18] Noam Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.

[19] Noam Nisan and David Zuckerman. Randomness is linear in space. *Journal of Computer and System Sciences*, 52(1):43–53, 1996.

[20] D. R. Stinson. Universal hash families and the leftover hash lemma, and applications to cryptography and computing. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 42:3–31, 2002. Available at http://www.cacr.math.uwaterloo.ca/~dstinson/publist.html.

[21] Salil P. Vadhan and Colin Jia Zheng. Characterizing pseudoentropy and simplifying pseudorandom generator constructions. In *Proceedings of the 44th ACM Symposium on the Theory of Computing*, pages 817–836, 2012.

[22] Umesh V. Vazirani and Vijay V. Vazirani. Efficient and secure pseudo-random number generation (extended abstract). In *Proceedings of the 25th IEEE Symposium on Foundation of Computer Science*, pages 458–463, 1984.

[23] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *Proceedings of the 23rd IEEE Symposium on Foundation of Computer Science*, pages 80–91, 1982.

A Proofs Omitted

Proof of Claim 3.3. Note that $\bar{f}(Y,R)$ is identically distributed to $U_{\mathcal{Y}}$, so it is equivalent to show

$$\mathsf{CD}_{t\cdot(\varepsilon/n)^{O(1)}}(\ (H(Y,R),H_C(Y,R))\ ,\ U_{n+s}\ \mid\ \bar{f}(Y,R),H,H_C\ )\ =\ O((2^{3s}\cdot\varepsilon\cdot n)^{\frac{1}{7}})\ .$$

It follows from the $(t-O(n),\varepsilon)$-one-way-ness of $\bar{f}$ (see Claim 3.3) and Lemma 3.3 that

$$\mathbf{H}_{t-O(n)}((Y,R)\mid\bar{f}(Y,R))\geq n+\log(1/\varepsilon)\ \ . \tag{9}$$

Then, similar to Lemma 3.4, we have the following entropy conditions

$$\mathbf{H}_{\infty}((Y,R)\mid\bar{f}(Y,R))=n\ \ ,$$

$$\mathbf{H}_{t-O(n)}((Y,R)\mid\bar{f}(Y,R),h(Y,R),h)\geq\mathbf{H}_{t-O(n)}((Y,R)\mid\bar{f}(Y,R))-(n-d)\ \geq\ d+\log(1/\varepsilon)\ \ ,$$

hold for any $h\in\mathcal{H}$, where the second inequality is by applying the chain rule to Equation (9). Therefore,

$$\begin{aligned}
&\mathsf{CD}_{t\cdot(\varepsilon/n)^{O(1)}}(\ (H(Y,R),H_C(Y,R))\ ,\ U_{n+s}\ \mid\ \bar{f}(Y,R),H,H_C\ )\\
\leq\ &\mathsf{SD}(H(Y,R),U_{n-d}\,|\bar{f}(Y,R),H)+\mathsf{CD}_{t\cdot(\varepsilon/n)^{O(1)}}(H_C(Y,R),U_{d+s}\mid\ \bar{f}(Y,R),H(Y,R),H,H_C)\\
\leq\ &2^{-\frac{d}{2}}+O(2^{d+s}\cdot(n\cdot\varepsilon\cdot2^{-d})^{\frac{1}{3}})=2^{-\frac{d}{2}}+O(2^{d+s}\cdot(2^{\frac{-(7d+6s)}{2}}\cdot2^{-d})^{\frac{1}{3}})\\
=\ &O(2^{-\frac{d}{2}})=O(2^{\frac{3s+\log(\varepsilon\cdot n)}{7}})=O((2^{3s}\cdot\varepsilon\cdot n)^{\frac{1}{7}})
\end{aligned}$$

where the first inequality is triangle, the statistical distance is due to the leftover hash lemma and the computational distance of the second inequality is by the Goldreich-Levin Theorem. □