

Incompressibility and Next-Block Pseudoentropy

Iftach Haitner ✉

The Blavatnik School of Computer Science at Tel-Aviv University, Israel

Noam Mazor ✉

The Blavatnik School of Computer Science at Tel-Aviv University, Israel

Jad Silbak ✉

The Blavatnik School of Computer Science at Tel-Aviv University, Israel

Abstract

A distribution is k -incompressible, Yao [FOCS '82], if no efficient compression scheme compresses it to less than k bits. While being a natural measure, its relation to other computational analogs of entropy such as *pseudoentropy*, Hastad, Impagliazzo, Levin, and Luby [SICOMP '99], and to other cryptographic hardness assumptions, was unclear.

We advance towards a better understating of this notion, showing that a k -incompressible distribution has $(k - 2)$ bits of *next-block pseudoentropy*, a refinement of pseudoentropy introduced by Haitner, Reingold, and Vadhan [SICOMP '13]. We deduce that a samplable distribution X that is $(H(X) + 2)$ -incompressible, implies the existence of one-way functions.

2012 ACM Subject Classification Theory of computation → Computational complexity and cryptography

Keywords and phrases incompressibility, next-block pseudoentropy, sparse languages

Digital Object Identifier 10.4230/LIPIcs.ITCS.2023.66

Related Version *Full Version*: <https://eccc.weizmann.ac.il/report/2022/032/> [7]

Funding Research supported by Israel Science Foundation grant 666/19.

Iftach Haitner: Member of the Check Point Institute for Information Security.

Jad Silbak: Research supported by Israel Science Foundation grant 1628/17.

Acknowledgements We thank Geoffroy Couteau, Ronen Shaltiel and Ofer Shayevitz for many useful discussions.

1 Introduction

Computational analogs of information-theoretic notions have given rise to some of the most interesting phenomena in the theory of computation. For example, *computational indistinguishability*, a computational analogue of statistical indistinguishability introduced by Goldwasser and Micali [5], enabled the bypassing of Shannon's impossibility results on perfectly secure encryption [15], and provided the basis for the computational theory of pseudorandomness [2, 21]. *Pseudoentropy*, a computational analogue of entropy introduced by Hastad, Impagliazzo, Levin, and Luby [10], was the key to their fundamental result that established the equivalence of pseudorandom generators and one-way functions and has become a basic concept in complexity theory and cryptography. *Next-block pseudoentropy*, a refinement of pseudoentropy introduced by Haitner, Reingold, and Vadhan [8] and Vadhan and Zheng [19], has led to simpler and more efficient constructions of pseudorandom generators based on one-way functions. An analogue of entropy for the realm of *unforgeability*, named *inaccessible entropy*, introduced by Haitner, Reingold, Vadhan, and Wee [9, 6], has led to simpler and more efficient constructions of *statistically hiding commitment* and *universal one-way hash functions* from one-way functions.



© Iftach Haitner, Noam Mazor, and Jad Silbak;

licensed under Creative Commons License CC-BY 4.0

14th Innovations in Theoretical Computer Science Conference (ITCS 2023).

Editor: Yael Tauman Kalai; Article No. 66; pp. 66:1–66:18

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In contrast to the above, *incompressibility*, a computational analogue of entropy introduced by Yao [21], was much less explored. Roughly, a random variable X is *k-incompressible*, if there exists no efficient (i.e., poly-time) compression scheme that compresses X to less than k bits. That is, there exists no efficient encoding scheme (Enc, Dec) , i.e., $\text{Dec}(\text{Enc}(x)) = x$ for every $x \in \text{Supp}(X)$, with

$$\mathbb{E}[|\text{Enc}(X)|] < k .$$

(Both X and k are functions of a “security parameter” n , which we omit throughout the introduction). It is immediate that a pseudorandom distribution of k bits is $(k - O(1))$ -incompressible.¹ More generally, Wee [20] proved that a distribution with k -bits of pseudoentropy, i.e., computationally indistinguishable from a distribution Y of k -bits of (real) Shannon entropy, is $(k - O(\log n))$ -incompressible. In contrast, the converse direction is less clear. Barak, Shaltiel, and Wigderson [1] showed how to extract $k - \omega(\log n)$ (close to uniform) bits from a *k-strongly-incompressible* source X , i.e., $\Pr[|\text{Enc}(X)| < k - t] \leq 2^{-t}$ for every t .² Other works showed that incompressibility is unlikely to imply pseudoentropy. Wee [20] showed that proving that incompressibility implies (similar amount of) pseudoentropy cannot be done using *black-box reductions*. Hsiao, Lu, and Reyzin [11] proved that under a certain cryptographic assumption, there exists a distribution whose *conditional* incompressibility is much larger than its conditional pseudoentropy, where conditional means that the compression and pseudoentropy are measures with respect to a randomly generated common reference string. Still, the most basic questions remained open:

Does k -incompressibility imply having a different, natural, type of “pseudoentropy”?
Does *non-trivial* incompressibility, i.e., sufficiently larger than the real entropy, imply the existence of one-way functions?

We give affirmative answers to the above questions, proving that a k -incompressible source has $(k - 2)$ bits of next-block pseudoentropy, and thus, a samplable source X that is $(H(X) + 2)$ -incompressible implies the existence of one-way functions. Before stating our results in more details, we recall the notion of next-block pseudoentropy Haitner et al. [8], focusing on its single-bit block variant, called *next-bit pseudoentropy*.

Next-bit pseudoentropy

Next-bit pseudoentropy measures the *bit-wise unpredictability* of X : how hard is it to predict X_i from $X_{<i} = (X_1, \dots, X_{i-1})$, for a uniform i . More formally, a random variable $X = (X_1, \dots, X_m)$ over $\{0, 1\}^m$ has next-bit pseudoentropy k , if there exists a set of random variables $\{Y_i\}_{i \in [m]}$, jointly distributed with X , such that

1. $\sum_{i \in [m]} H(Y_i | X_{<i}) \geq k$, for H being the Shannon entropy function, and
2. $(X_{<i}, X_i)$ is computationally indistinguishable from $(X_{<i}, Y_i)$, for every i .

That is, X has next-bit pseudoentropy k if predicting X_i from $X_{<i}$ is not easier than predicting Y_i from $X_{<i}$, where the bits of Y have k bits of (real) entropy given the past. It follows from [19] that k -bits of (standard) pseudoentropy implies k -bits of next-bit pseudoentropy,³ but

¹ The $O(1)$ loss is since even the (true) uniform distribution can be compressed by $\Theta(1)$ bits (using non prefix-free schemes) [17].

² I.e., not only that one cannot efficiently compress X to less than k bits, but it cannot compress, non-trivially, even *parts* of X .

³ Indeed, if X has k -bits of pseudoentropy, X is $k - H(X)$ KL-hard to estimate, and thus X_I is $(k - H(X))/m$ KL-hard to estimate given $X_{<I}$. The later implies that X_I has pseudoentropy k/m given $X_{<I}$.

the converse does not always hold.⁴ Yet, Haitner et al. [8] showed that an efficiently samplable source with non-trivial next-bit pseudoentropy, can be used to construct pseudorandom generators (and thus one-way functions).

1.1 Our Results

In this paper, following [1, 20], we focus on the *non-uniform settings* – the efficient algorithms get non-uniform polynomial-size advice per input length – and defer the uniform version of our results to the future version (see more details in Section 1.2). We state our results with respect to a weaker notion of incompressibility, where the source is only assumed to be incompressible by *prefix-free* schemes: no codeword is a prefix of another.

► **Lemma 1** (Incompressibility \rightarrow next-bit pseudoentropy). *A random variable that is k -incompressible by efficient prefix-free schemes, has next-bit pseudoentropy (at least) $k(n) - 2$.*

That is, incompressibility is a stronger measure of “pseudoentropy” than next-bit pseudoentropy. Since, incompressibility is weaker than pseudoentropy, we now have a rather good understanding about the computational hardness incompressibility induces. By Haitner et al. [8], Lemma 1 yields the following characterization.

► **Theorem 2** (Non-trivial incompressibility implies one-way functions). *Assume there exist an (efficiently) samplable random variable X that is $(H(X) + 2 + 1/p(n))$ -incompressible by efficient prefix-free schemes, for some $p \in \text{poly}$, then one-way functions exist.*

That is, if one-way functions do not exist, e.g., we live in “Pessiland” [13], then any samplable distribution can be compressed to its entropy plus two bits.

Theorem 2 improves upon previous results that require additional structure from the incompressible distribution. Wee [20] proved that if an m -bit *flat* X (i.e., uniformly distributed over its support) is $(H(X) + \Omega(\log m))$ -incompressible, then one-way functions exist. Where a simple application of Barak et al. [1], yields the same for $(H(X) + \omega(\log n))$ -*strongly*-incompressible X .

► **Remark 3** (Incompressibility and one-way functions). The common method for proving that a certain primitive implies (the existence of) one-way functions, is to show that the primitive induces a function that if invertible would contradict the security of the primitive ([14]). Interestingly, such an approach does not seem to work for samplable incompressible distributions; the efficient sampler of an incompressible X might use $\text{poly}(|X|) > H(X)$ random bits. Thus, even if we are able to invert it (assuming that one-way functions don’t exist) and output the (long) random string used for sampling, this alone does not contradict the incompressibility of X . Moreover, while the probability of every element in the support of X can be efficiently estimated (assuming that one-way functions do not exist, [14]), this also does not suffice for efficient compression.

So while our main result is establishing a connection between incompressibility and next-bit pseudoentropy, the fact that incompressibility implies one-way functions is an interesting corollary that to the best of our understanding does not immediately follow any standard technique.

⁴ Let g be a pseudorandom generator from n bits to $2n$ bits. Then $Z = (g(U_n), U_n)$ does not have pseudoentropy larger than n (Z is determined by its last n bits), but has $2n$ bits of next-bit pseudoentropy: let Y_1, \dots, Y_{2n} be uniform and independent bits, and $(Y_{2n+1}, \dots, Y_{3n}) = (Z_{2n+1}, \dots, Z_{3n})$.

Applications to sparse languages

A language \mathcal{L} is s -sparse if $|\mathcal{L} \cap \{0, 1\}^n| \leq 2^{s(n)}$. Theorem 2 yields the following characterization of sparse languages.

► **Theorem 4 (Informal).** *Let \mathcal{L} be an s -sparse language and let \mathcal{D} be a samplable distribution over \mathcal{L} (i.e., $\text{Supp}(\mathcal{D}_n) \cap \{0, 1\}^n \subseteq \mathcal{L} \cap \{0, 1\}^n$). If \mathcal{D} is $(s + 3)$ -incompressible, then one-way functions exist.*

► **Remark 5 (The two bits gap).** One might wonder whether the annoying two bits gap in Lemma 1 and Theorems 2 and 4 is unavoidable. We pay these two bits since in the proof of Lemma 1 we use the *arithmetic encoding* prefix-free compression scheme, which compresses a random variable X to $H(X) + 2$ bits. We use arithmetic encoding since it can be implemented efficiently given oracle access to the accumulated distribution function of X (with respect to the lexicographic order of elements). Indeed, the 2 bits gap can be reduced given a better encoding scheme that is efficient in these settings. While there are prefix-free compression schemes that get closer to $H(x)$, cf., Shannon [16], Fano [3], and Huffman [12], these schemes might not be efficient for large alphabets (in our case, the alphabet is $\{0, 1\}^m$). So as far as we know, there might be a random variable X that is not compressible to less than $H(x) + 2$ by an efficient prefix-free scheme. Since the existence of such a variable is unlikely to imply the existence of one-way functions, the 2 bits gap in our results might be unavoidable.

One might do better by asking for incompressibility by arbitrary (no prefix-free) efficient schemes (which might even compress to strictly less than $H(X)$ bits). But bearing in mind that compression is used for communicating many samples from the distribution, asking for prefix-freeness seems like the natural definition for incompressibility.

1.2 Our Technique

We explain here the high-level approach for proving Lemma 1 (incompressibility \rightarrow next-bit pseudoentropy). Let $X = (X_1, \dots, X_m)$ be a random variable over $\{0, 1\}^m$, and assume X does not have next-bit pseudoentropy $k - 2$. We prove that such X can be (efficiently) compressed into less than k bits, proving the lemma. Recall that next-bit pseudoentropy measures the bit-wise unpredictability of X : how hard is it to predict X_i from $X_{<i}$, for $i \leftarrow [m]$. So X not having $k - 2$ bits of next-bit pseudoentropy implies that X is “rather predictable” in an online fashion: predict X_1 , then use X_1 to predict X_2 , and so on. We use this characterization to design a prefix-free bits encoding of X , of average length less than k , more details below.

Let $I \leftarrow [m]$. Since X does not have $k - 2$ bits of next-bit pseudoentropy, the random variable $(X_{<I}, X_I)$ is distinguishable from $(X_{<I}, Y_I)$ for *every* set of random variable $\{Y_i\}_{i \in [m]}$ with $\sum_{i \in [m]} H(Y_i | X_{<i}) \geq (k - 2)$. That is, X_I has low entropy given $X_{<I}$, in the eyes of poly-time distinguisher. As discussed above, this implies that X_I is somewhat predictable given $X_{<I}$. Vadhan and Zheng [19] formalized this intuition and proved that for such an X , there exists a poly-time predictor P that predicts X_I from $X_{<I}$ within small KL-divergence. Specifically,

$$\text{KL}(X_{<I}, X_I | X_{<I}, P(X_{<I})) < (k - 2)/m - H(X_I | X_{<I}) \quad (1)$$

Let $Y = (Y_1, \dots, Y_m)$ be the random variable defined inductively by P as follows: $Y_1 = P(\epsilon)$, and $Y_i = P(Y_{<i})$. By Equation (1) and the chain-rules of KL-divergence, we deduce that

$$\text{KL}(X || Y) < k - 2 - H(X) \quad (2)$$

The above suggest the following method for compressing X . Use P for designing a good prefix-free encoding scheme for Y , and then apply Equation (2) to deduce that the scheme also compresses X well. The scheme we design for Y is the *arithmetic encoding* scheme. This scheme is useful for compressing any distribution \mathcal{D} for which we know how to compute the *accumulated probability function* $F^{\mathcal{D}}(x) := \sum_{x' \leq x} \Pr_{\mathcal{D}}[x']$. Since Y is defined according to P , it is not hard to see that the accumulated function of Y is efficiently computable, implying that the arithmetic encoding (Enc, \cdot) of Y can be computed efficiently. Furthermore, since (Enc, \cdot) is the arithmetic encoding of Y , it holds that

$$|\text{Enc}(y)| \leq -\log(\Pr[Y = y]) + 2 \quad (3)$$

for every $y \in \text{Supp}(Y)$. Using a well-known fact about using “wrong compression”, we deduce that

$$\mathbb{E}[|\text{Enc}(X)|] \leq H(X) + \text{KL}(X||Y) + 2 \quad (4)$$

Applying Equation (2), we conclude that $\mathbb{E}[|\text{Enc}(X)|] < k$.

► **Remark 6 (Uniform incompressibility).** In the above proof we assumed that the predictor P is deterministic (which can be assumed without loss of generality in the non-uniform setting). This assumption was crucial for the arithmetic encoding to work. Otherwise, the encoder and decoder will not agree on the same accumulated probability function. In the uniform setting, this obstacle can be overcome by letting the encoder and decoder have access to *shared randomness* (independent of the distribution to compress) which they can use as the random coins of P . We measure the compression of such shared randomness scheme by the expected encoding length over the shared randomness and the underlying distribution. All the results stated in this paper extend, with essentially the same parameters, to this uniform setting.

We note that, when compressing more than one sample from the distribution, the shared randomness needs to be sampled only once. Thus, the amortized cost of our compressing scheme is still equal to the entropy of the distribution, even if the decoder sends its randomness (instead of using shared randomness).

1.3 Related Work

Yao [21] used the term *effective entropy* to measure by how much a distribution can be compressed efficiently. So k -incompressibility is equivalent to having effective entropy at least k . Yao [21] did not require the compression scheme to be prefix-free, but only required that for every $t \in \text{poly}$, the sequence $\text{Enc}(x_1), \dots, \text{Enc}(x_t)$, where x_1, \dots, x_t are independent samples from the distribution, are decoded with high probability.

An interesting line of work considered efficient compression of of samplable distributions with (efficient) membership queries. Goldberg and Sipser [4] showed that any such distribution can be compressed by $\log n$ bits, and Trevisan, Vadhan, and Zuckerman [18] gave better schemes for flat distributions, and for distribution generated by log-space machines.

Paper Organization

Basic definitions and notations are given in Section 2. The formal definition of incompressibility and our results relating it to next-bit pseudoentropy, including some results that are not mentioned above, are given in Section 3.

2 Preliminaries

2.1 Notations

All logarithms are taken in base 2. We use calligraphic letters to denote sets and distributions, uppercase for random variables, and lowercase for values and functions. Let poly stand for the set of all polynomials. Let PPT stand for probabilistic poly-time, and n.u.-poly-time stand for non-uniform poly-time. An n.u.-poly-time algorithm A is equipped with a (fixed) poly-size advice string set $\{z_n\}_{n \in \mathbb{N}}$ (that we typically omit from the notation), and we let A_n stand for A equipped with the advice z_n (used for inputs of length n). Let neg stand for a negligible function. For a set $\mathcal{L} \subseteq \{0, 1\}^*$, let $\mathcal{L}_n := \mathcal{L} \cap \{0, 1\}^n$. A set \mathcal{S} is *prefix free*, if for no $x_1 \neq x_2 \in \mathcal{S}$ it holds that x_1 is a prefix of x_2 . Given a vector $v \in \Sigma^n$, let v_i denote its i^{th} entry, let $v_{<i} = (v_1, \dots, v_{i-1})$ and $v_{\leq i} = (v_1, \dots, v_i)$. Similarly, for a set $\mathcal{I} \subseteq [n]$, let $v_{\mathcal{I}}$ be the ordered sequence $(v_i)_{i \in \mathcal{I}}$.

2.2 Distributions and Random Variables

When unambiguous, we will naturally view a random variable as its marginal distribution. The support of a finite distribution \mathcal{P} is defined by $\text{Supp}(\mathcal{P}) := \{x : \Pr_{\mathcal{P}}[x] > 0\}$. For a (discrete) distribution \mathcal{P} , let $x \leftarrow \mathcal{P}$ denote that x was sampled according to \mathcal{P} . Similarly, for a set \mathcal{S} , let $x \leftarrow \mathcal{S}$ denote that x is drawn uniformly from \mathcal{S} . For random variable ensemble $B = \{B_n\}_{n \in \mathbb{N}}$ and $t: \mathbb{N} \mapsto \mathbb{N}$, let $B^t = \left\{ B_n^{t(n)} \right\}_{n \in \mathbb{N}}$ for $B_n^{t(n)}$ being $t(n)$ independent copies of B_n . For $m \in \mathbb{N}$, we use U_m to denote a uniform random variable over $\{0, 1\}^m$ (that is independent from other random variables in consideration). We use the following standard definitions:

► **Definition 7 (Indistinguishability).** *Distribution ensembles $\mathcal{P} = \{\mathcal{P}_n\}_{n \in \mathbb{N}}$ and $\mathcal{Q} = \{\mathcal{Q}_n\}_{n \in \mathbb{N}}$ are $\text{n.u.-poly-time-indistinguishable}$, if*

$$\Pr_{x \leftarrow \mathcal{P}_n}[\mathsf{D}(x) = 1] - \Pr_{x \leftarrow \mathcal{Q}_n}[\mathsf{D}(x) = 1] \leq \text{neg}(n)$$

for any n.u.-poly-time algorithm D .

► **Definition 8 (Samplability).** *A distribution ensemble $\mathcal{P} = \{\mathcal{P}_n\}$ is **samplable**, if there exists poly-time algorithm (sampler) S and poly-time computable function $m \in \text{poly}$, such that for every $n \in \mathbb{N}$, $\mathsf{S}(1^n; U_{m(n)})$ is distributed according to \mathcal{P}_n .*

2.2.1 Entropy and Distance Measures

The *Shannon entropy* of a distribution \mathcal{P} is defined by $H(\mathcal{P}) = \sum_{p \in \text{Supp}(\mathcal{P})} \Pr_{\mathcal{P}}[p] \cdot \log \frac{1}{\Pr_{\mathcal{P}}[p]}$. The conditional entropy of a random variable A given B , is defined as $H(A|B) = \mathbb{E}_{b \leftarrow B}[H(A|_{B=b})]$. We will use the following known facts:

► **Fact 9 (Chain rule for Shannon entropy).** *For a random variable $A = (A_1, \dots, A_n)$, it holds that $H(A_1, \dots, A_n) = \sum_{i=1}^n H(A_i | A_{<i})$.*

The *KL-divergence* (also known as, *Kullback-Leibler divergence*, and *relative entropy*) between distributions \mathcal{P} and \mathcal{Q} is defined by

$$\text{KL}(\mathcal{P} || \mathcal{Q}) = \sum_{a \in \text{Supp}(\mathcal{P})} \Pr_{\mathcal{P}}[a] \log \left(\frac{\Pr_{\mathcal{P}}[a]}{\Pr_{\mathcal{Q}}[a]} \right).$$

The KL-divergence also admits a chain rule.

► **Fact 10** (Chain rule for KL-divergence). For random variables $A = (A_1, \dots, A_n)$ and $B = (B_1, \dots, B_n)$, it holds that

$$\text{KL}(A||B) = \sum_{i \in [n]} \mathbb{E}_{a_1, \dots, a_n \leftarrow A} [\text{KL}(A_i |_{A_{<i} = a_{<i}} || B_i |_{B_{<i} = a_{<i}})].$$

Both Entropy and KL-divergence admit data processing inequalities.

► **Fact 11** (Data processing inequality). For every random variables A, B and function f , it holds that $\text{KL}(f(A)||f(B)) \leq \text{KL}(A||B)$ and $H(f(A)) \leq H(A)$.

We will use the following observation that bounds the KL-divergence between two distributions using their maximal ratio over a set.

► **Proposition 12** (Bounding KL-divergence). Let \mathcal{X} and \mathcal{Y} be finite distributions, and assume $\log \frac{\Pr_{\mathcal{X}}[s]}{\Pr_{\mathcal{Y}}[s]} \geq \alpha$ for every element s of a set \mathcal{S} . Then $\text{KL}(\mathcal{X}||\mathcal{Y}) > (\alpha - \log e) \cdot \Pr_{\mathcal{X}}[\mathcal{S}]$.

Proposition 12 is proved in the full version of this paper.

We will also use the following simple observation to bound the probability of an event in terms of Entropy.

► **Fact 13** (Bounding probability via entropy). Let C be a Boolean random variable. If $H(C) \geq d$ then $\Pr[C = 1] \geq (d/40)^2$.

Fact 13 is proved in the full version of this paper.

2.3 Encoding and Compression

We start with the definition of encoding schemes.

► **Definition 14** (Encoding schemes). A pair of algorithms (Enc, Dec) is an encoding for a distribution \mathcal{D} , if for every $x \in \text{Supp}(\mathcal{D})$ it holds that $\text{Dec}(\text{Enc}(x)) = x$.⁵ The pair is an encoding for a distribution ensemble $\mathcal{D} = \{\mathcal{D}_n\}_{n \in \mathbb{N}}$, if $(\text{Enc}(1^n, \cdot), \text{Dec}(1^n, \cdot))$ is an encoding scheme for \mathcal{D}_n for every n . The encoding is fixed-length, if $|\text{Enc}(1^n, x)| = \ell(n)$ for every n and $x \in \text{Supp}(\mathcal{D}_n)$, for some function ℓ , and is prefix-free if the set $\{\text{Enc}(1^n, x) : x \in \text{Supp}(\mathcal{D}_n)\}$ is prefix-free for every n .

► **Definition 15** (Compressing a distribution). An encoding scheme (Enc, \cdot) ℓ -compresses a distribution ensemble \mathcal{D} , if $\mathbb{E}_{x \leftarrow \mathcal{D}_n} [|\text{Enc}(1^n, x)|] \leq \ell(n)$ for every $n \in \mathbb{N}$.

We will refer to an encoding scheme that compresses the distribution as a *compressing scheme*. When clear from the context, we omit the parameter 1^n given to the encoder and the decoder.

Changing distributions

The following well-known observation bounds the price you pay by using the compressing scheme for the “wrong” distribution.

► **Proposition 16** (Changing distributions). Let \mathcal{P} and \mathcal{Q} be finite distributions with $\text{KL}(\mathcal{P}||\mathcal{Q}) < \infty$. Let (Enc, Dec) be a compression scheme for \mathcal{Q} , such that $|\text{Enc}(q)| \leq -\log(\Pr_{\mathcal{Q}}[q]) + c$ for every $q \in \text{Supp}(\mathcal{Q})$ for some $c > 0$. Then (Enc, Dec) is a compression scheme for \mathcal{P} with $\mathbb{E}_{p \leftarrow \mathcal{P}} [|\text{Enc}(p)|] \leq H(\mathcal{P}) + \text{KL}(\mathcal{P}||\mathcal{Q}) + c$.

Proposition 16 is proved in the full version of this paper.

⁵ Our results readily extend to encoding schemes with negligible decoding errors.

Arithmetic encoding

We use *Arithmetic encoding*, a well-known prefix-free encoding scheme.

► **Definition 17** (Arithmetic encoding). *Let X be a finite random variable over \mathcal{U} and let \prec be a total order over \mathcal{U} . Let $F : \mathcal{U} \rightarrow [0, 1]$ by $F(x) = (\sum_{a \prec x} \Pr[X = a]) + 1/2 \cdot \Pr[X = x]$. Define $\text{Enc}(x)$ as the first $(\lceil -\log \Pr[X = x] \rceil + 1)$ bits of $F(x)$.*

Arithmetic encoding enjoys the following properties:

► **Fact 18** (Properties of arithmetic encoding, lemma 2.8 in [18]). *For every random variable X and order \prec , the function Enc defined in Definition 17 is one-to-one and monotone, and the scheme $(\text{Enc}, \text{Enc}^{-1})$ is prefix-free and compresses X to $H(X) + 2$ bits.*

2.4 One-way Functions

► **Definition 19** (One-way functions). *A poly-time computable function $f : \{0, 1\}^n \mapsto \{0, 1\}^n$ is n.u.-one-way, if*

$$\Pr_{x \leftarrow \{0, 1\}^n} [\mathbf{A}_n(f(x)) \in f^{-1}(f(x))] \leq \text{neg}(n)$$

for any n.u.-poly-time \mathbf{A} .

2.5 Pseudoentropy and Next-bit Pseudoentropy

In this section we define pseudoentropy and next-bit pseudoentropy, a special case of next-block pseudoentropy defined at Haitner et al. [8].

Pseudoentropy

We start with recalling the standard notion of pseudoentropy [10].

► **Definition 20** (Pseudoentropy). *A random variable ensemble B has n.u.-pseudoentropy (at least) k , if for every $p \in \text{poly}$ there exists an ensemble $C = \{C_n\}_{n \in \mathbb{N}}$, such that:*

1. $H(C_n) \geq k(n)$, and
2. B and C are n.u.-poly-time-indistinguishable.

We also use a conditional version of the above definition.

► **Definition 21** (Conditional pseudoentropy). *Let $B = \{B_n\}_{n \in \mathbb{N}}$ be a random variable ensemble over $\{0, 1\}$ jointly distributed with $X = \{X_n\}_{n \in \mathbb{N}}$. We say that B has n.u.-conditional-pseudoentropy (at least) k given X , if for every $p \in \text{poly}$ there exists an ensemble $C = \{C_n\}_{n \in \mathbb{N}}$ over $\{0, 1\}$, jointly distributed with (X, B) , such that:*

1. $H(C_n | X_n) \geq k(n) - 1/p(n)$,⁶ and
2. (X, B) and (X, C) are n.u.-poly-time-indistinguishable.

► **Remark 22** (Order of quantifiers). The order of quantifiers in our definition of conditional pseudoentropy, Definition 21, is different from the ones appearing in Vadhan and Zheng [19]. Their definition requires that for every $p \in \text{poly}$ there exists an ensemble of random variable $\{C_n\}_{n \in \mathbb{N}}$ such that:

⁶ [8] do not have the $1/p(n)$ term in their definition of conditional pseudoentropy (which is implicit in their definition of next-bit pseudoentropy). Following [19], we add this term to slightly simplify the text.

1. $H(C_n|X_n) \geq k(n) - 1/p(n)$, and
2. (X, B) and (X, C) are $1/p(n)$ -indistinguishable for circuits of size $p(n)$.

Clearly Definition 21 implies that of [19], but it turns out that the converse also holds (making the definitions equivalent). Indeed, assuming that the definition of [19] holds, we show that there exists an ensemble $C = \{C_n\}_{n \in \mathbb{N}}$ such that (X, C) is n.u.-poly-time-indistinguishable from (X, B) (satisfying our definition of conditional pseudoentropy). For every n , let C_n be a random variable that fulfills Items 1 and 2 in the definition of [19] with respect to $p(n) = n^c$ for the largest possible value of $c \leq n$. It is not hard to see that $\{C_n\}_{n \in \mathbb{N}}$ satisfies Item 2 in Definition 21, and thus the definitions are indeed equivalent.

A similar difference exist between [19]'s and our definitions of (non conditional) pseudoentropy. Also in this case it can be shown that the two definitions are equivalent.

Next-bit pseudoentropy

We are now ready to define next-bit pseudoentropy. Intuitively, a random variable B over $\{0, 1\}^m$ has next-bit pseudoentropy k , if for a uniformly chosen $i \in [m]$, it holds that B_i has pseudoentropy k/m given $B_{<i}$. Formally, this is put using the above notation of conditional pseudoentropy.

► **Definition 23** (Next-bit pseudoentropy). *The random variables ensemble $B = \{B_n\}_{n \in \mathbb{N}}$ over $\{0, 1\}^{m(n)}$ has n.u.-next-bit-pseudoentropy (at least) k if the following holds: let $I = \{I_n\}_{n \in \mathbb{N}}$ be an ensemble of uniformly distributed random variables over $[m(n)]$, then $\{(B_n)_{I_n}\}_{n \in \mathbb{N}}$ has n.u.-conditional-pseudoentropy k/m given $\{(B_n)_{<I_n}\}_{n \in \mathbb{N}}$.*

In their construction of pseudorandom generator, [8] has proved that a generator whose next-bit pseudoentropy is larger than its input length, can be used to construct pseudorandom generators and thus one-way functions. The following is the non-uniform variant of their result.

► **Theorem 24** (Extending next-bit pseudoentropy implies one-way functions, [8]). *Assume there is a poly-time computable function $f: \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$ such that $\{f(U_n)\}_{n \in \mathbb{N}}$ has n.u.-next-bit-pseudoentropy $n + 1/p(n)$ for some $p \in \text{poly}$. Then there exists a n.u.-one-way functions.*

For our needs, we use the following corollary of the above.

► **Corollary 25** (Non-trivial next-bit pseudoentropy implies one-way functions). *Assume there is a poly-time computable function $f: \{0, 1\}^n \rightarrow \{0, 1\}^{m(n)}$ such that $\{f(U_n)\}_{n \in \mathbb{N}}$ has n.u.-next-bit-pseudoentropy $H(f(U_n)) + 1/p(n)$ for some $p \in \text{poly}$. Then there exists a n.u.-one-way functions.*

That is, it is enough to show that f has next-bit pseudoentropy larger than its image real entropy, rather than its input size. A proof sketch for Corollary 25 is given in the full version of this paper.

It is easy to see that next-bit pseudoentropy behaves nicely under direct product. It turns out that the converse is also true: if the direct product has kt bits of next-bit pseudoentropy, then a single copy has next-bit pseudoentropy (at least) k .

► **Proposition 26** (Direct product of next-bit pseudoentropy). *For any random variable ensemble B and $t \in \text{poly}$, if B^t has n.u.-next-bit pseudoentropy $(t \cdot k)$, then B has n.u.-next-bit pseudoentropy k .*

Proposition 26 is proved in the full version of this paper.

KL-hardness

In their “hashing free” construction of pseudorandom generators from one-way functions, Vadhan and Zheng [19] introduced the notion of *KL-hardness* of a distribution. Informally, it states that it is hard to approximate the distribution within a small KL divergence. This notion is formally defined using *KL-predictors*.

► **Definition 27** (KL-predictors). *Let (X, B) be a distribution over $\{0, 1\}^m \times \{0, 1\}$, and let $P: \{0, 1\}^m \times \{0, 1\} \mapsto (0, +\infty)$ be a deterministic function. We say that P is a δ -KL-predictor of B given X , if*

$$\text{KL}(X, B || X, C_P) \leq \delta,$$

for C_P being a random variable (jointly distributed with X) with $\Pr[C_P = b | X = x] = \frac{P(x,b)}{P(x,0)+P(x,1)}$.

A distribution is KL-hard if it possesses no efficient KL-predictor.

► **Definition 28** (KL-hardness). *Let (X, B) be a distribution ensemble over $\{0, 1\}^{m(n)} \times \{0, 1\}$. We say that B is δ -n.u.-KL-hard given X , if there exists no n.u.-poly-time P and $q \in \text{poly}$ such that P_n is a $(\delta - 1/q(n))$ -KL-predictor of B given X , for infinitely many n 's.*

We use the following result from [19].

► **Theorem 29** (KL-hardness imply pseudoentropy, [19] Corollary 3.9).

Let $(X, B) = \{(X_n, B_n)\}_{n \in \mathbb{N}}$ be a random variable ensemble over $\{0, 1\}^{m(n)} \times \{0, 1\}$. If B is δ -n.u.-KL-hard given X , then it has n.u.-conditional-pseudoentropy $H(B_n | X_n) + \delta(n)$ given X .

That is, KL-hard distribution has non-trivial next-bit pseudoentropy.⁷

3 Incompressibility and Next-bit Pseudoentropy

In this section, we define several notions of incompressibility and relate them to next-bit pseudoentropy (defined in Section 2.5). As said in the introduction, we focus on the non-uniform settings.

Incompressibility

We start with the standard notion of incompressibility that we define with respect to prefix-free compression schemes (it is immediate that a distribution that is incompressible with respect to arbitrary scheme is incompressible according to our definition).

► **Definition 30** (Incompressibility). *A distribution ensemble \mathcal{B} is k -incompressible, if for every n.u.-poly-time prefix-free compression scheme (Enc, \cdot) for \mathcal{B} , it holds that*

$$\mathbb{E}_{x \leftarrow \mathcal{B}_n}[|\text{Enc}(x)|] \geq k(n),$$

for all but finitely many n 's.

We will also address the following more fine-grain version of incompressibility.

⁷ [19] also proved that the converse direction holds.

► **Definition 31** (Local incompressibility).

A distribution ensemble \mathcal{B} is (α, β) -locally-incompressible, if for every n.u.-poly-time prefix-free compression scheme (Enc, \cdot) for \mathcal{B} , it holds that

$$\Pr_{x \leftarrow \mathcal{B}_n} \left[|\text{Enc}(x)| \geq \log \frac{1}{\Pr_{\mathcal{B}_n}[x]} + \alpha(n) \right] \geq \beta(n),$$

for all but finitely many n 's.

Local incompressibility gets handy when the gap between next-bit pseudoentropy and the real entropy is smaller than 2, settings in which our result for (non-local) incompressibility is not applicable.

We observe the following connections between incompressibility and local incompressibility, both proved in Section 3.6.

► **Proposition 32** (Incompressibility \rightarrow local incompressibility). *A k -incompressible distribution ensemble \mathcal{B} over $\{0, 1\}^{m(n)}$, with $m \in \text{poly}$, is $(k(n) - H(\mathcal{B}_n) - 2, \frac{1}{3m(n)})$ -locally-incompressible.*

► **Proposition 33** (local incompressibility \rightarrow incompressibility). *An (α, β) -locally incompressible distribution ensemble \mathcal{B} , is $H(\mathcal{B}_n) + \beta(n)(\alpha(n) - \log e)$ -incompressible.*

Relation to pseudoentropy

We recall the following two facts. The first states that incompressibility is not stronger than pseudoentropy.

► **Theorem 34** (Pseudoentropy \rightarrow incompressibility, [20]). *Let \mathcal{B} be a distributions ensemble over $\{0, 1\}^{m(n)}$ with n.u.-pseudoentropy k , then \mathcal{B} is $(k - 2 \log m)$ -incompressible.⁸*

While it is unknown if incompressibility is a weaker notion than pseudoentropy, there is an oracle separation between them, as stated in the next theorem.

► **Theorem 35** (Incompressibility $\not\rightarrow$ pseudoentropy, oracle separation, [20]). *There is an oracle \mathcal{O} and a distribution ensemble \mathcal{B} that relative to \mathcal{O} , \mathcal{B} is $(n - \omega(\log n))$ -incompressible but does not have pseudoentropy larger than $n/2$.*

3.1 Our Results

3.1.1 Incompressibility \rightarrow Next-Bit Pseudoentropy

Our main result states that incompressibility implies next-bit pseudoentropy.

► **Lemma 36** (Incompressibility \rightarrow next-bit pseudoentropy). *The following holds for every distribution ensemble \mathcal{B} over $\{0, 1\}^{m(n)}$ with $m \in \text{poly}$.*

1. \mathcal{B} is k -incompressible $\implies \mathcal{B}$ has n.u.-next-bit pseudoentropy $k(n) - 2$.
2. \mathcal{B} is (α, β) -locally-incompressible $\implies \mathcal{B}$ has n.u.-next-bit pseudoentropy $H(\mathcal{B}_n) + \beta(n)(\alpha(n) - 2 - \log e)$.

Lemma 36 is proved in Section 3.3. Combining its first part with Corollary 25, yields the following informative theorem.

► **Theorem 37** (Incompressibility \rightarrow one-way functions). *Assume there exists a samplable distribution ensemble \mathcal{B} over $\{0, 1\}^{m(n)}$ that is $(H(\mathcal{B}_n) + 2 + 1/p(n))$ -incompressible for some $p \in \text{poly}$, then n.u.-one-way functions exist.*

⁸ This result holds also for non-prefix free compressing schemes.

Amortization

When amortizing over several instances, Lemma 36 yields the following tighter characterization.

► **Lemma 38** (Incompressibility \rightarrow next-bit pseudoentropy, multiples copies). *Let \mathcal{B} be a distribution ensemble over $\{0, 1\}^{m(n)}$ with $m \in \text{poly}$, such that \mathcal{B}^t for some $t \in \text{poly}$, is $(t \cdot k)$ -incompressible. Then \mathcal{B} has n.u.-next-bit pseudoentropy $k(n) - 2/t(n)$.*

Proof. Since \mathcal{B}^t has $t \cdot k$ -incompressibility, by Lemma 36 it has n.u.-next-bit pseudoentropy $t \cdot k - 2$. Hence, Proposition 26 yields that \mathcal{B} has n.u.-next-bit pseudoentropy $k - 2/t$. ◀

3.1.2 Strong-Next-Bit Pseudoentropy

It is easy to see that next-bit pseudoentropy does not imply incompressibility.

► **Proposition 39** (Next-bit pseudoentropy $\not\rightarrow$ incompressibility). *Assuming n.u.-one-way function exists, then there exists a samplable distribution ensemble with n.u.-next-bit pseudoentropy $2n$, that is not $(n + 1)$ -incompressible.*

Proof sketch. Let $g: \{0, 1\}^n \mapsto \{0, 1\}^{2n}$ be a pseudorandom generator. The distribution ensemble $\mathcal{B} = \{(g(U_n), U_n)\}_{n \in \mathbb{N}}$ has $2n$ n.u.-next-bit pseudoentropy. But \mathcal{B} can be trivially compressed to n bits by $\text{Enc}(f(x), x) = x$. ◀

In contrast, the following variant of next-bit pseudoentropy does imply (and is equivalent to) incompressibility.

► **Definition 40** (Strong-next-bit pseudoentropy). *A random variable ensemble B has n.u.-strong-next-bit pseudoentropy k , if for every n.u.-poly-time, fixed-length encoding (Enc, \cdot) , the ensemble $\{\text{Enc}(B)\}_{n \in \mathbb{N}}$ has n.u.-next-bit pseudoentropy k .⁹*

That is, B has strong-next-bit pseudoentropy if every encoding of B has next-bit pseudoentropy. Lemma 36 easily extends to strong-next-bit pseudoentropy.

► **Lemma 41** (Incompressibility \rightarrow strong-next-bit pseudoentropy). *A k -incompressible distribution ensemble has n.u.-strong-next-bit pseudoentropy $k - 2$.*

Proof. Let \mathcal{B} be a k -incompressible distribution ensemble, and let B be a random variable ensemble distributed according to \mathcal{B} . It follows that for every n.u.-poly-time fixed-length encoding scheme (Enc, \cdot) , it holds that $\text{Enc}(B_n)$ is k -incompressible. Otherwise, one can efficiently compress \mathcal{B}_n by first encode it according to Enc , and then compress the output. Thus, by Lemma 36, $\text{Enc}(B_n)$ has n.u.-next-bit pseudoentropy at least $k - 2$. ◀

More interestingly, strong-next-bit pseudoentropy does imply incompressibility.

► **Lemma 42** (Strong-next-bit pseudoentropy \rightarrow incompressibility). *Let \mathcal{B} be a distribution ensemble with $\text{Supp}(\mathcal{B}_n) = \{0, 1\}^{m(n)}$. If \mathcal{B} has n.u.-strong-next-bit pseudoentropy $k + 1/p$ for some $p \in \text{poly}$, then \mathcal{B} is k -incompressible.*

Without requiring that $\text{Supp}(\mathcal{B}_n) = \{0, 1\}^{m(n)}$, we would only get that \mathcal{B} is incompressible by an encoding schemes that is prefix-free over $\{0, 1\}^{m(n)}$. Interestingly, the proof of Lemma 41 readily yields that this type of incompressibility is sufficient for next-bit pseudoentropy. Lemma 42 is proved in Section 3.5.

⁹ Note that in this definition, Enc is not necessarily a compressing encoding.

3.2 Applications to Sparse Languages

We use the following definition of sparse language. (Recall that for a set $\mathcal{L} \subseteq \{0, 1\}^*$, $\mathcal{L}_n := \mathcal{L} \cap \{0, 1\}^n$.)

► **Definition 43.** A language $\mathcal{L} \in \{0, 1\}^*$ is s -sparse if $|\mathcal{L}_n| \leq 2^{s(n)}$ for every $n \in \mathbb{N}$.

The results of Section 3 immediately yield that unless one-way functions exist, any samplable distribution over s -sparse language can be compressed to $s + 2$ bits.

► **Theorem 44.** For every samplable distribution ensemble $\mathcal{B} = \{\mathcal{B}_n\}_{n \in \mathbb{N}}$ and s -sparse language \mathcal{L} , such that $\text{Supp}(\mathcal{B}_n) \subseteq \mathcal{L}_n$ for every $n \in \mathbb{N}$, if \mathcal{B} is $(s + 2 + 1/p)$ -incompressible for some $p \in \text{poly}$, then n.u.-one-way functions exist.

Proof. Since \mathcal{B} is over \mathcal{L} , $s(n) \geq H(\mathcal{B}_n)$, and thus \mathcal{B} is $(H(\mathcal{B}_n) + 2 + 1/p)$ -incompressible. Thus by Theorem 37, n.u.-one-way functions exist. ◀

3.3 Proving Lemma 36 – Incompressibility implies Next-Bit Pseudoentropy

In this part we prove Lemma 36. We use the following lemma, proved in Section 3.4.

► **Lemma 45** (Next-bit predictor to compression). *There exists a pair of oracle-aided algorithms (Enc, Dec) such that the following holds: let $\mathsf{P}: \{0, 1\}^* \rightarrow (0, +\infty)$ be deterministic algorithm, and for $m \in \mathbb{N}$ let $\mathcal{D}_m^{\mathsf{P}}$ be the distribution over $\{0, 1\}^m$ defined by:*

$$\Pr_{\mathcal{D}_m^{\mathsf{P}}}[x] = \prod_{i \in [m]} \frac{\mathsf{P}(x_{<i}, x_i)}{\mathsf{P}(x_{<i}, 0) + \mathsf{P}(x_{<i}, 1)}$$

Then $(\text{Enc}^{\mathsf{P}}(1^m, \cdot), \text{Dec}^{\mathsf{P}}(1^m, \cdot))$ is a prefix-free compressing scheme for $\mathcal{D}_m^{\mathsf{P}}$ with $|\text{Enc}(x)| \leq \lceil -\log \Pr_{\mathcal{D}_m^{\mathsf{P}}}[x] \rceil + 1$ for every $x \in \{0, 1\}^m$. The running-time of $\text{Enc}^{\mathsf{P}}(1^m, \cdot)$ and $\text{Dec}^{\mathsf{P}}(1^m, \cdot)$ is polynomial in m and in the output length of P on inputs of length at most m .

Given Lemma 45, we are ready to prove Lemma 36.

Proof of Lemma 36. Let \mathcal{B} be as in Lemma 36, and assume it does not have next-bit pseudoentropy q (we will chose q later). We start by proving that there exists n.u.-poly-time algorithm P such that the distribution ensemble $\{\mathcal{D}_{m(n)}^{\mathsf{P}}\}_{n \in \mathbb{N}}$ is close in KL-divergence to \mathcal{B} , for $\mathcal{D}_m^{\mathsf{P}}$ being according to Lemma 45.

Let I_n be a uniform random variable over $[m(n)]$, and let B be an ensemble of random variables distributed according to \mathcal{B} . By assumption and Definition 23, $\{(B_n)_{I_n}\}_{n \in \mathbb{N}}$ has no q/m conditional pseudoentropy given $\{(B_n)_{<I_n}\}_{n \in \mathbb{N}}$. Thus, Theorem 29 implies that $\{(B_n)_{I_n}\}_{n \in \mathbb{N}}$ is not $\delta(n) := (q/m - H((B_n)_{I_n} | (B_n)_{<I_n}))$ KL-hard given $\{(B_n)_{<I_n}\}_{n \in \mathbb{N}}$. Namely, (see, Definition 28) there exists an infinite set $\mathcal{I} \subseteq \mathbb{N}$, $c > 0$ and a n.u.-poly-time algorithm $\mathsf{P}: \{0, 1\}^* \rightarrow (0, +\infty)$, such that P_n is a $(\delta - 1/n^c)$ -KL-predictor of $\{(B_n)_{I_n}\}_{n \in \mathbb{N}}$ given $\{(B_n)_{<I_n}\}_{n \in \mathbb{N}}$, for every $n \in \mathcal{I}$. Fix $n \in \mathcal{I}$, and omit n from the notation, and let D_m^{P} be a random variable distributed according to $\mathcal{D}_m^{\mathsf{P}}$. By definition of KL-predictor (Definition 27), it holds that:

$$\text{KL}(B_{<I}, B_I | B_{<I}, \mathsf{P}_n(B_{<I})) \leq \delta - 1/n^c < \delta = q/m - H(B_I | B_{<I}) \quad (5)$$

66:14 Incompressibility and Next-Block Pseudoentropy

It follows that

$$\begin{aligned}
q/m &> \mathbb{E}_{i \leftarrow I} [\text{KL}(B_{<i}, B_i \| B_{<i}, \mathbb{P}(B_{<i})) + H(B_i | B_{<i})] \\
&= 1/m \cdot \sum_{i \in [m]} (\text{KL}(B_{<i}, B_i \| B_{<i}, \mathbb{P}(B_{<i})) + H(B_i | B_{<i})) \\
&= 1/m \cdot \sum_{i \in [m]} (\text{KL}(B_{<i} \| B_{<i}) + \mathbb{E}_{b \leftarrow B} [\text{KL}(B_i |_{B_{<i}=b_{<i}} \| \mathbb{P}(B_{<i}) |_{B_{<i}=b_{<i}})] + H(B_i | B_{<i})) \\
&= 1/m \cdot \sum_{i \in [m]} (\mathbb{E}_{b \leftarrow B} [\text{KL}(B_i |_{B_{<i}=b_{<i}} \| (D_m^{\mathbb{P}})_i |_{(D_m^{\mathbb{P}})_{<i}=b_{<i}})] + H(B_i | B_{<i})) \\
&= 1/m \cdot (\text{KL}(B \| D_m^{\mathbb{P}}) + H(B)).
\end{aligned}$$

The second equality is due to chain-rule of KL-divergence. The third equality by the definition of $\mathcal{D}_m^{\mathbb{P}}$ and since $\text{KL}(X \| X) = 0$ for every random variable X . The last equality holds by chain-rule of KL-divergence and Shannon entropy. We deduce that

$$\text{KL}(B \| D_m^{\mathbb{P}}) < q - H(B) \tag{6}$$

Let (Enc, Dec) be the compressing scheme guaranteed by Lemma 45. Lemma 45 implies that

$$\left| \text{Enc}^{\mathbb{P}}(x) \right| \leq \lceil -\log \Pr_{\mathcal{D}_m^{\mathbb{P}}}[x] \rceil + 1 \leq -\log \Pr_{\mathcal{D}_m^{\mathbb{P}}}[x] + 2 \tag{7}$$

for every $x \in \{0, 1\}^m$. Given the above, we separately prove each part of the lemma.

\mathcal{B} is k -incompressible. Let $q(n) = k(n) - 2$. By Proposition 16 and Equations (6) and (7),

$$\mathbb{E}_{x \leftarrow \mathcal{B}_n} \left[\left| \text{Enc}^{\mathbb{P}}(x) \right| \right] < q(n) + 2 = k(n)$$

We conclude that \mathcal{B} is k -compressible, yielding a contradiction.

\mathcal{B} is (α, β) -locally-incompressible. By Equation (7),

$$\left| \text{Enc}^{\mathbb{P}}(x) \right| \leq -\log \Pr_{\mathcal{D}_m^{\mathbb{P}}}[x] + 2 \tag{8}$$

Let $\mathcal{S} = \left\{ x \in \{0, 1\}^m : \left| \text{Enc}^{\mathbb{P}}(x) \right| \geq -\log \Pr_{\mathcal{B}}[x] + \alpha \right\}$ and let $\eta = \Pr_{\mathcal{B}}[\mathcal{S}]$. Equation (8) yields that $-\log \Pr_{\mathcal{B}}[x] + \alpha \leq -\log \Pr_{\mathcal{D}_m^{\mathbb{P}}}[x] + 2$ for every $x \in \mathcal{S}$, implying that $\alpha - 2 \leq \log \frac{\Pr_{\mathcal{B}}[x]}{\Pr_{\mathcal{D}_m^{\mathbb{P}}}[x]}$ for every $x \in \mathcal{S}$. Applying Proposition 12 with respect to \mathcal{S} , yields that

$$\text{KL}(B \| D_m^{\mathbb{P}}) > \eta \cdot (\alpha - 2 - \log e)$$

Applying Equation (6) for $q = H(B) + \beta \cdot (\alpha - 2 - \log e)$, yields that

$$\text{KL}(B \| D_m^{\mathbb{P}}) < \beta \cdot (\alpha - 2 - \log e)$$

We deduce that that $\beta > \eta = \Pr_{\mathcal{B}}[\mathcal{S}]$, yielding that \mathcal{B} is not (α, β) -locally incompressible. \blacktriangleleft

3.4 Proving Lemma 45 – Next-bit Predictor to Compression

Proof. Let P, m and $\mathcal{D} = \mathcal{D}_m^P$ be according to the lemma statement. Our encoder defined below, encodes \mathcal{D} according to the arithmetic encoding, see Definition 17, with respect to the lexicographic order. Recall that on input x , the arithmetic encoding should output $e(x)$: the first $(\lceil \log 1/\Pr_{\mathcal{D}}[x] \rceil + 1)$ bits of $F(x) := (\sum_{y < x} \Pr_{\mathcal{D}}[y]) + 1/2 \cdot \Pr_{\mathcal{D}}[x]$.

► **Algorithm 46** (Enc).

Oracle: Predictor P .

Input: $x \in \{0, 1\}^m$.

Operation:

1. Let $p_{\text{eq}} = 1$ and $p_{\text{less}} = 0$.
2. For every $i \in [m]$:
 - a. If $x_i = 1$: $p_{\text{less}} = p_{\text{less}} + p_{\text{eq}} \cdot \frac{P(x_{<i}, 0)}{P(x_{<i}, 0) + P(x_{<i}, 1)}$.
 - b. $p_{\text{eq}} = p_{\text{eq}} \cdot \frac{P(x_{<i}, x_i)}{P(x_{<i}, 0) + P(x_{<i}, 1)}$.
3. Output the first $(\lceil -\log p_{\text{eq}} \rceil + 1)$ bits of $p_{\text{less}} + p_{\text{eq}}/2$.

By induction, at the end of the i^{th} iteration of Enc it holds that $p_{\text{less}} = \Pr_{y \leftarrow \mathcal{D}}[y_{\leq i} < x_{\leq i}]$, and $p_{\text{eq}} = \Pr_{y \leftarrow \mathcal{D}}[y_{\leq i} = x_{\leq i}]$. Hence, when Enc reaches Step 3., it holds that $p_{\text{less}} = \Pr_{y \leftarrow \mathcal{D}}[y < x]$, and $p_{\text{eq}} = \Pr_{\mathcal{D}}[x]$, stipulating that $\text{Enc}^P(x) = e(x)$. Thus by Fact 18, we deduce that Enc^P is a prefix-free compressing scheme for \mathcal{D} with $|\text{Enc}(x)| = \lceil \log 1/\Pr_{\mathcal{D}}[x] \rceil + 1$, for every $x \in \{0, 1\}^m$.

Regarding efficiency, since P only outputs positive numbers, the running time of Enc^P is polynomial time in m and output size of P on inputs of length at most m . In addition, a decoding procedure $\text{Dec}^P(1^m, \cdot)$ for $\text{Enc}^P(1^m, \cdot)$ can be implemented with the same efficiently using a straightforward binary search over $\{0, 1\}^m$. ◀

3.5 Proving Lemma 42 – Strong-Next-Bit Pseudoentropy implies Incompressibility

Proof. Assume that \mathcal{B} is not k -incompressible and let (Enc, Dec) be the n.u.-poly-time compressing scheme that k' -compresses \mathcal{B} , with $k'(n) < k(n)$ for infinite many n 's. Let $q(n)$ for $q \in \text{poly}$ be a bound on the output length of Enc on inputs of length $m(n)$, and let $\text{Enc}'(1^n, x) = (\text{Enc}(x), 0^{q(n) - |\text{Enc}(x)|})$.

Let B be a random variable ensemble distributed according to \mathcal{B} . We claim that $\text{Enc}'(1^n, B_n)$ does not have next-bit pseudoentropy $k(n) + 1/p(n)$. Indeed, consider the following distinguisher $D = \{D_n\}_{n \in \mathbb{N}}$:

► **Algorithm 47** (D_n).

Input: $y \in \{0, 1\}^*$, $b \in \{0, 1\}$.

Operation:

1. If there is no $i \leq |y|$ such that $\text{Dec}(y_{\leq i}) \in \{0, 1\}^{m(n)}$ and $\text{Enc}(\text{Dec}(y_{\leq i})) = y_{\leq i}$, output 0.
2. Otherwise, output b .

We now show that D contradicts the $k(n) + 1/p(n)$ -next-bit pseudoentropy of $\{\text{Enc}'(B_n)\}_{n \in \mathbb{N}}$ (Definition 23). It is easy to see that if (y, b) is a prefix of $\text{Enc}'(x)$ for some $x \in \{0, 1\}^{m(n)}$, then D_n outputs 0. Hence, we conclude the proof by showing that D_n outputs 1 with noticeable probability over $(\text{Enc}'(B_n)_{< I_n}, C_n)$, for $I_n \leftarrow [q(n)]$ and any random variable C_n with

66:16 Incompressibility and Next-Block Pseudoentropy

$$H(C_n \mid \text{Enc}'(B_n)_{<I_n}) \geq \frac{k(n) + 1/p(n)}{q(n)} - \frac{1}{2p(n)q(n)} = \frac{k(n)}{q(n)} + \frac{1}{2p(n)q(n)}.$$

Indeed, fix $n \in \mathbb{N}$ with $k'(n) < k(n)$, and let C_n be such random variable and let $\delta(n) := \frac{1}{2p(n)q(n)} \geq 1/\text{poly}(n)$. Let W be the indicator for the event $I_n > |\text{Enc}(B_n)|$ (that is, $W = 1$ if $I_n > |\text{Enc}(B_n)|$ and $W = 0$ otherwise). Compute,

$$\begin{aligned} & k(n)/q(n) + \delta(n) \\ & \leq H(C_n \mid \text{Enc}'(B_n)_{<I_n}) \\ & = H(C_n \mid \text{Enc}'(B_n)_{<I_n}, W) \\ & = \Pr[W = 1] \cdot H(C_n \mid \text{Enc}'(B_n)_{<I_n}, W = 1) \\ & \quad + \Pr[W = 0] \cdot H(C_n \mid \text{Enc}'(B_n)_{<I_n}, W = 0) \\ & \leq \Pr[W = 1] \cdot H(C_n \mid \text{Enc}'(B_n)_{<I_n}, W = 1) + \Pr[W = 0] \\ & = \Pr[I_n > |\text{Enc}(B_n)|] \cdot H(C_n \mid \text{Enc}'(B_n)_{<I_n}, I > |\text{Enc}(B_n)|) + \Pr[I_n \leq |\text{Enc}(B_n)|] \\ & \leq \Pr[I_n > |\text{Enc}(B_n)|] \cdot H(C_n \mid \text{Enc}'(B_n)_{<I_n}, I > |\text{Enc}(B_n)|) + k(n)/q(n). \end{aligned}$$

The first equality holds since $\text{Enc}'(B_n)_{<I_n}$ determines the value of W . The second inequality holds since $H(C_n) \leq 1$. It follows that

$$\Pr[I_n > |\text{Enc}(B_n)|] \cdot H(C_n \mid \text{Enc}'(B_n)_{<I_n}, I > |\text{Enc}(B_n)|) \geq \delta(n) \quad (9)$$

In particular,

$$\Pr[I_n > |\text{Enc}(B_n)|] \geq \delta(n) \quad (10)$$

and

$$H(C_n \mid I_n > |\text{Enc}(B_n)|) \geq H(C_n \mid \text{Enc}'(B_n)_{<I_n}, I_n > |\text{Enc}(B_n)|) \geq \delta(n) \quad (11)$$

Hence, Fact 13 yields that

$$\Pr[C_n = 1 \mid I_n > |\text{Enc}(B_n)|] \geq (\delta(n)/40)^2 \quad (12)$$

Since D_n outputs 1 when $C_n = 1$ and $I_n > |\text{Enc}(B_n)|$, we deduce that

$$\Pr[D_n(\text{Enc}'(B_n)_{<I_n}, C_n) = 1 \mid I_n > |\text{Enc}(B_n)|] \geq (\delta(n)/40)^2 \quad (13)$$

Combining the above with Equation (10), yields that,

$$\begin{aligned} & \Pr[D_n(\text{Enc}'(B_n)_{<I_n}, C_n) = 1] \\ & = \Pr[D_n(\text{Enc}'(B_n)_{<I_n}, C_n) = 1 \mid I_n > |\text{Enc}(B_n)|] \cdot \Pr[I_n > |\text{Enc}(B_n)|] \\ & \geq (\delta(n)/40)^3. \end{aligned}$$

It follows that D_n distinguishes C_n from $\text{Enc}'(B_n)_{I_n}$ given $\text{Enc}'(B_n)_{<I_n}$ with probability $(\delta(n)/40)^3 \geq 1/\text{poly}(n)$. \blacktriangleleft

3.6 Additional Missing Proofs

Proposition 32

Proof of Proposition 32. Assume toward contradiction that \mathcal{B} is not (α, β) -locally-incompressible, for $\alpha(n) = k(n) - H(\mathcal{B}_n) - 2$ and $\beta = 1/3m(n)$.

Let (Enc, \cdot) be a compression scheme that violates the local-incompressibility of \mathcal{B} , and consider the following encoder Enc' :

$$\text{Enc}'(1^n, x) = \begin{cases} 0, \text{Enc}(x) & |\text{Enc}(x)| \leq m(n) \\ 1, x & \text{o/w.} \end{cases}$$

It follows that $|\text{Enc}'(1^n, x)| \leq m(n) + 1$ for every $x \in \{0, 1\}^{m(n)}$, and by (local) compressibility

$$\Pr_{x \leftarrow \mathcal{B}_n} \left[|\text{Enc}'(1^n, x)| \geq \log \frac{1}{\Pr_{\mathcal{B}_n}[x]} + \alpha(n) + 1 \right] < \beta(n) \quad (14)$$

It follows that,

$$\begin{aligned} \mathbb{E}_{x \leftarrow \mathcal{B}_n} [|\text{Enc}'(1^n, x)|] &\leq \mathbb{E}_{x \leftarrow \mathcal{B}_n} [-\log \Pr_{\mathcal{B}_n}[x] + \alpha(n) + 1] + \beta(n)(m(n) + 1) \\ &\leq H(\mathcal{B}_n) + \alpha(n) + 1 + \beta(n)(m(n) + 1) \\ &= k(n) - 1 + \beta(n)(m(n) + 1). \\ &< k(n) \end{aligned} \quad (15)$$

The first equation holds by Equation (14), the equality holds by our choice of α and the last inequality follows by our choice of β . This concludes that proof since by Equation (15), \mathcal{B}_n is not k -incompressible. \blacktriangleleft

Proposition 33

Proof of Proposition 33. Let (Enc, \cdot) be a prefix-free compression scheme for \mathcal{B} , and let $\mathcal{D} = \{\mathcal{D}_n\}_{n \in \mathbb{N}}$ be the distribution ensemble over $\text{Supp}(\mathcal{B}_n) \cup \{\perp\}$, defined by $\Pr_{\mathcal{D}_n}[x] = 2^{-|\text{Enc}(x)|}$ for $x \in \text{Supp}(\mathcal{B}_n)$, $\Pr_{\mathcal{D}_n}[\perp] = 1 - \sum_{x \in \text{Supp}(\mathcal{B}_n)} 2^{-|\text{Enc}(x)|}$.

Since \mathcal{B} is locally-incompressible, it holds that $\Pr_{x \leftarrow \mathcal{B}_n} \left[\log \frac{\Pr_{\mathcal{B}_n}[x]}{\Pr_{\mathcal{D}_n}[x]} \geq \alpha(n) \right] \geq \beta(n)$. Thus, Proposition 12 yields that $\text{KL}(\mathcal{B}_n \| \mathcal{D}_n) \geq \beta(n)(\alpha(n) - \log e)$, and therefore

$$\begin{aligned} \mathbb{E}_{x \leftarrow \mathcal{B}_n} [|\text{Enc}(x)|] &= \mathbb{E}_{x \leftarrow \mathcal{B}_n} [-\log \Pr_{\mathcal{D}_n}[x]] \\ &= \mathbb{E}_{x \leftarrow \mathcal{B}_n} \left[-\log \Pr_{\mathcal{B}_n}[x] + \log \frac{\Pr_{\mathcal{B}_n}[x]}{\Pr_{\mathcal{D}_n}[x]} \right] \\ &= H(\mathcal{B}_n) + \text{KL}(\mathcal{B}_n \| \mathcal{D}_n) \\ &\geq H(\mathcal{B}_n) + \beta(n)(\alpha(n) - \log e). \end{aligned} \quad \blacktriangleleft$$

References

- 1 Boaz Barak, Ronen Shaltiel, and Avi Wigderson. Computational analogues of entropy. In *Approximation, Randomization, and Combinatorial Optimization.. Algorithms and Techniques (APPROX)*, pages 200–215. Springer, 2003.
- 2 Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo random bits. In *Annual Symposium on Foundations of Computer Science (FOCS)*, pages 112–117, 1982.
- 3 Robert M Fano. *The transmission of information*. Massachusetts Institute of Technology, Research Laboratory of Electronics, 1949.
- 4 Andrew Goldberg and Michael Sipser. Compression and ranking. In *Annual ACM Symposium on Theory of Computing (STOC)*, pages 440–448, 1985.

66:18 Incompressibility and Next-Block Pseudoentropy

- 5 Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, pages 270–299, 1984.
- 6 Iftach Haitner, Thomas Holenstein, Omer Reingold, Salil P. Vadhan, and Hoeteck Wee. Inaccessible entropy II: IE functions and universal one-way hashing. *Theory of Computing*, 2020. Preliminary version in Eurocrypt '10.
- 7 Iftach Haitner, Noam Mazor, and Jad Silbak. Incompressibility and next-block pseudoentropy. Electronic Colloquium on Computational Complexity, TR22-032, 2022. URL: <https://eccc.weizmann.ac.il/report/2022/032/>.
- 8 Iftach Haitner, Omer Reingold, and Salil Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. *SIAM Journal on Computing*, 42(3):1405–1430, 2013.
- 9 Iftach Haitner, Omer Reingold, Salil Vadhan, and Hoeteck Wee. Inaccessible entropy i: Inaccessible entropy generators and statistically hiding commitments from one-way functions. Technical Report 2010.05586, arXiv, 2019. Preliminary version in STOC '09.
- 10 Johan Hastad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, pages 1364–1396, 1999.
- 11 Chun-Yuan Hsiao, Chi-Jen Lu, and Leonid Reyzin. Conditional computational entropy, or toward separating pseudoentropy from compressibility. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pages 169–186, 2007.
- 12 David A Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952.
- 13 Russell Impagliazzo. A personal view of average-case complexity. In *Proceedings of the Tenth Annual Structure in Complexity Theory Conference*, pages 134–147. IEEE Computer Society, 1995.
- 14 Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography. In *Annual Symposium on Foundations of Computer Science (FOCS)*, pages 230–235, 1989.
- 15 Claude Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, pages 656–715, 1949.
- 16 Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- 17 Wojciech Szpankowski and Sergio Verdú. Minimum expected length of fixed-to-variable lossless compression without prefix constraints. *IEEE Transactions on Information Theory*, 57(7):4017–4025, 2011.
- 18 Luca Trevisan, Salil Vadhan, and David Zuckerman. Compression of samplable sources. *Computational Complexity*, 14(3):186–227, 2005.
- 19 Salil Vadhan and Colin Jia Zheng. Characterizing pseudoentropy and simplifying pseudorandom generator constructions. In *Annual ACM Symposium on Theory of Computing (STOC)*, pages 817–836, 2012.
- 20 Hoeteck Wee. On pseudoentropy versus compressibility. In *Annual IEEE Conference on Computational Complexity (COMPLEXITY)*, pages 29–41, 2004.
- 21 Andrew C. Yao. Theory and applications of trapdoor functions. In *Annual Symposium on Foundations of Computer Science (FOCS)*, pages 80–91, 1982.