Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 201 (2024)

On one-way functions and the average time complexity of almost-optimal compression

Marius Zimand*

Abstract

We show that one-way functions exist if and only if there exists an efficiently-samplable distribution relative to which almost-optimal compression is hard on average. The result is obtained by combining a theorem of Ilango, Ren, and Santhanam [IRS21, IRS22] and one by Bauwens and Zimand [BZ23].

1 Introduction

Several recent papers show that the existence of one-way functions (OWF) is equivalent to the hardness of certain problems in meta-complexity [LP20, LP21, RS21, IRS21, IRS22, LP23a, LP23b, HLO24, LS24]. The motivation for this research line comes primarily from cryptography, where one-way functions play a central role[1]. Ilango, Ren and Santhanam [IRS21, IRS22] have obtained a result of this type involving standard (unbounded) Kolmogorov complexity. Informally speaking, they have shown that one-way functions exist if and only if "finding good approximations of Kolmogorov complexity" is hard on average with respect to some polynomial-time samplable distribution. Bauwens and Zimand [BZ23] have shown that given a good approximation of the Kolmogorov complexity of a string x, one can compress x in probabilistic polynomial time to a string of length close to its complexity (so, x is almost-optimally compressed). The combination of these 2 results yields the following theorem.

Theorem 1 (Informal statement)**.** *The following two assertions are equivalent:*

1. *There exists a one-way function.*
2. *Almost optimal compression is hard on average with respect to some polynomial-time samplable distribution.*

The result of Ilango et. al. [IRS21, IRS22] is not exactly stated in the form that we mentioned above. For this reason, we prefer to give a proof which does not directly invoke [IRS21, IRS22], but which closely follows their method. In one direction, it is based on results of Impagliazzo, Levin and Luby[IL90, IL89] connecting the existence of OWFs to the hardness of approximating poly-time samplable distributions, and, in the other direction, it is based on the connection between OWFs and pseudo-random generators established by Håstad, Impagliazzo, Levin and Luby [HILL99].

2 Definitions, and technical tools

Kolmogorov complexity. We fix an optimal universal Turing machine U with prefix-free domain. A program for string x is a string p such that $U(p) = x$. The prefix-free Kolmogorov complexity $K(x)$ of the string x is the length of a shortest program for x.[2]

*Department of Computer and Information Sciences, Towson University, Baltimore, MD. Partially supported by a grant from the School of Emerging Technologies at Towson University.

[1] See [HLO24] and [LS24] for a discussion of some of these and related works.

[2] The prefix-free Kolmogorov complexity $K(x)$ is a little more convenient for the proof than the plain complexity $C(x)$. The difference $K(x) - C(x)$ is bounded by $2 \log |x|$ and, therefore, the result is valid for $C(x)$ as well.

ISSN 1433-8092

Distributions. We consider ensembles of distributions. An ensemble has the form $D = (D_n)_{n \in \mathbb{N}}$, where each D_n is a distribution on $\{0,1\}^n$. The ensemble D is *samplable* if there exists a probabilistic algorithm Samp, such that for every n and every $x \in \{0,1\}^n$,

$$\text{Prob}\,[\mathsf{Samp}(1^n) = x] = D_n(x)$$

(the probability is over the randomness of Samp).

D is said to be *P-samplable*, in case Samp runs in polynomial time.

Some notation: For every x, we denote $D(x) = D_{|x|}(x)$. For every m, U_m denotes the uniform distribution over m-bit strings.

Lemma 1. *If D is samplable, then for every x in its support,*

$$K(x) \leq \log \frac{1}{D(x)} + 3\log(|x|) + O(1).$$

Proof. Fix a binary string x and let n be its length. Given n and the code of Samp, one can compute $D_n(y)$ for all strings y of length n and then list all these strings in descending order of their $D_n(\cdot)$ probability (with ties broken, say, lexicographically). The string x is described by its rank t in this list. Since the D_n-probability of the first t strings in the order is at most 1 and at least $t \cdot D_n(x)$, it follows that $t \leq \lceil 1/D_n(x) \rceil$. An overhead of $2\log(|x|) + O(1)$ bits is added to obtain a self-delimited description in the standard way. □

Lemma 2. *For every distribution D, and every $\Delta \geq 0$,*

$$\text{Prob}_{x \leftarrow D}\ [K(x) \geq \log \frac{1}{D(x)} - \Delta] \geq 1 - 2^{-\Delta}.$$

Proof. The complement of the event in the probability is $E = \{x \mid D(x) \leq 2^{-\Delta} \cdot 2^{-K(x)}\}$. We have

$$D(E) = \sum_{x \in E} D(x) \leq \sum_{x \in E} 2^{-\Delta} \cdot 2^{-K(x)} \leq 2^{-\Delta} \sum_{x \in \{0,1\}^*} 2^{-K(x)} \leq 2^{-\Delta} \cdot 1 = 2^{-\Delta}.$$

In the penultimate transition, we have used the Kraft inequality, which is legitimate because $K(\cdot)$ represents the lengths of a prefix-free code. □

Formal statement of Theorem 1. The following 2 assertions are equivalent:

Assertion (1): The hypothesis "∃ OWF": There exists a polynomial-time computable $f : \{0,1\}^* \to \{0,1\}^*$ with the following property: For every probabilistic polynomial-time algorithm $\mathsf{Inverter}$, every $q \in \mathbb{N}$ and almost every length $n \in \mathbb{N}$,

$$\text{Prob}_{x \leftarrow U_n, \mathsf{Inverter}}[\mathsf{Inverter}(1^n, f(x)) \in f^{-1}(f(x))] \leq 1/n^q.$$

(The notation $\text{Prob}_{x \leftarrow U_n, \mathsf{Inverter}}$ means that the probability is over $U_n \times$ randomness of $\mathsf{Inverter}$.)

Assertion (2): The hypothesis "almost optimal compression is hard on average" : There exists a P-samplable distribution D and a constant c with the following property: For every probabilistic polynomial-time algorithm $\mathsf{Compress}$, at almost every length n,

$$\text{Prob}_{x \leftarrow D_n, \mathsf{Compress}}[\mathsf{Compress}\,(x) \text{ outputs a program of } x \text{ of length } \geq K(x) + c\log^2 n] > 1/100.$$

(The event in the probability expresses the failure of almost optimal compression, and thus assertion (2) states that for any efficient algorithm this failure happens with significant probability.)

Remark. The "infinitely often" version of Theorem 1 is also true, with essentially the same proof. More precisely, if we modify Assertions 1 and 2 by replacing "almost every length n" with "infinitely many lengths n," the modified assertions are also equivalent.

Also, the version in which the additive overhead $c\log^2 n$ is replaced by n^γ (for every $\gamma \in (0,1)$) is true with essentially the same proof.

Results from the literature that we use.

Theorem 2 ([IL89, IL90]; this variant is stated and proved in [IRS21])**.** *Assume the hypothesis "∃ OWF" is not true. Let $D = (D_n)_{n \in \mathbb{N}}$ be a P-samplable ensemble of distributions, and $q \in \mathbb{N}$. There exists a probabilistic polynomial-time algorithm A and a constant $c > 1$ such that for infinitely many n,*

$$\text{Prob}_{x \leftarrow D_n, A}\ [D_n(x)/c \le A(x) \le D_n(x)] \ge 1 - \frac{1}{n^q}.$$

In other words: If there are no one-way functions, then P-samplable distributions can be approximated efficiently in the average sense.

Theorem 3 ([BZ23])**.** *There exists a probabilistic polynomial-time algorithm* Compress *that for every input triple ($x \in \{0,1\}^*, m \in \mathbb{N}$, rational $\epsilon > 0$) outputs with probability 1 a string z of length $m + O(\log m \cdot \log |x|/\epsilon)$ and if $m \ge K(x)$ then*

$$\text{Prob}_{\textsf{Compress}}[z \textit{ is a program for } x] \ge 1 - \epsilon.$$

In other words: Given a good approximation of the Kolmogorov complexity of a string x, one can efficiently compress x almost optimally (where efficiently means in probabilistic polynomial time).

3 Proof of Theorem 1

Proof of assertion (2) → assertion (1).

We actually prove the contrapositive: ∄ OWF ⇒ ¬ assertion (2) (i.e, almost optimal compression is easy on average).

Let $D = (D_n)_{n \in \mathbb{N}}$ be a P-samplable ensemble. By Lemma 2 and Lemma 1, for some constant c, for every n

$$\text{Prob}_{x \leftarrow D_n}[\log \frac{1}{D_n(x)} - c \log n \le K(x) \le \log \frac{1}{D_n(x)} + c \log n] \ge 1 - 1/n.$$

Under our assumption "∄ OWF ," Theorem 2 states that there exists an algorithm that approximates $D_n(x)$ with high probability, and therefore it also approximates $K(x)$ with high probability. More precisely, by rescaling, we get a probabilistic polynomial-time algorithm A such that, for every n,

$$\text{Prob}_{x \leftarrow D_n, A}[K(x) \le A(x) \le K(x) + c \log n] \ge 1 - 1/n.$$

Then, the algorithm Compress from Theorem 3 with $m = A(x)$ and $\epsilon = 1/200$ shows the invalidity of assertion (2).

Proof of assertion (1) → assertion (2).

(∃ OWF ⇒ almost optimal compression is hard on average.)

The idea is that an efficient good compressor would break the security of any candidate pseudorandom generator (p.r.g.), because the output of the generator can be compressed to a much shorter string, whereas a genuinely random string cannot. Therefore, pseudorandom generators would not exist and hence there would be no OWF, contradicting assertion (1). Now, the details.

Suppose " ∃ OWF " is true. Then, by [HILL99] combined with the methods to obtain ensembles of p.r.g.'s with every possible output length [Gol01, Section 3.3.3], there exists an ensemble of p.r.g.'s $G = (G_n)_{n \in \mathbb{N}}$, computable in polynomial time (uniformly), with $G_n : \{0,1\}^{s(n)} \to \{0,1\}^n$, where the seed length $s(n)$ is bounded by $n^{1/3}$, that satisfies the following security guarantee: For every probabilistic polynomial-time algorithm T (the hypothetical distinguisher) and every $q \in \mathbb{N}$, for almost every $n \in \mathbb{N}$, the probabilities that (a) T accepts $G_n(U_{s(n)})$ and (b) T accepts U_n, differ by at most $1/n^q$.

Consider the following P-samplable distribution D_n:

with probability 1/2, output $G(U_{s(n)})$ and with probability 1/2, output U_n.

Clearly, if assertion (2) is false, then there exists a constant c, a probabilistic polynomial-time algorithm A (derived from Compress in the straightforward way), and an infinite set B, so that at every length $n \in B$, with $D_n \times$ (randomness of A)-probability $\geq 1 - 1/100$ approximates $K(x)$ with slack at most $c \log^2 n$. Let

$$\mathrm{BAD} = \{x \in \{0,1\}^n \mid \mathrm{Prob}_A[\,|A(x) - K(x)| \geq c \log^2 n\,] \geq 5/100\}.$$

(In other words: BAD is the event which says that A fails to approximate K with significant probability over the randomness of A.)

Then, for every $n \in B$, by Markov's inequality,

$$\mathrm{Prob}_{x \leftarrow D_n}[\mathrm{BAD}] \leq 1/5,$$

By inspecting the sampling procedure, we see that each element x in BAD has D_n-probability mass at least $(1/2) \cdot 2^{-n}$ and thus $1/5 \geq D_n(\mathrm{BAD}) \geq (\#BAD) \cdot (1/2 \cdot 2^{-n}) = 1/2 \cdot \mathrm{Prob}_{U_n}[BAD]$, and so

$$\mathrm{Prob}_{U_n}[BAD] \leq 2/5.$$

Also, each element in $\mathrm{BAD} \cap \mathrm{Im}(G(U_{s(n)})$ has D_n-probability mass at least $(1/2) \cdot 2^{-s(n)}$, which, similarly to the above, implies that

$$\mathrm{Prob}_{U_{s(n)}}[BAD \cap \mathrm{Im}(G(U_{s(n)}))] \leq 2/5.$$

We now define the probabilistic polynomial-time distinguisher T: T on input z of length n, executes A on input z, and *accepts*, if $A(z) \leq n^{1/2}$, and *rejects* otherwise. Note that $G(U_{s(n)})$ with probability 1 has prefix-free complexity at most $s(n) + 2 \log s(n) + O(1) \leq n^{1/3} + O(\log n)$. Therefore, if $G(U_{s(n)}) \notin$ BAD and if A uses randomness that yields good approximation, then T accepts $G(U_{s(n)})$. Also, U_n, with probability at least 1-1/n, has complexity at least $n - \log n$. If this is the case, and $U_n \notin$ BAD, and A uses correct randomness, then T rejects U_n.

Therefore, for every $n \in B$,

$$\mathrm{Prob}_{U_{s(n)},T}[T \text{ accepts } G(U_{s(n)})] \geq (1 - 2/5) \cdot (1 - 5/100) = 57/100$$

(the probability that $G(U_{s(n)})$ is not in BAD is at least $1 - 2/5$ and for strings not in BAD the probability that A uses correct randomness is at least $1 - 5/100$).

Also, for every $n \in B$,

$$\mathrm{Prob}_{U_n,T}[T \text{ accepts } U_n] \leq 1/n + 2/5 + 5/100 = 45/100 + 1/n.$$

($1/n$ is the probability that U_n has complexity less than $n - \log n$, $2/5$ is the probability that U_n is in BAD, and $5/100$ is the probability conditioned on $U_n \notin$ BAD that A is using wrong randomness).

We are done, because the two inequalities contradict the security of G.

References

[BZ23] Bruno Bauwens and Marius Zimand. Universal almost optimal compression and Slepian-Wolf coding in probabilistic polynomial time. *J. ACM*, 70(2):1–33, 2023. (arxiv version posted in 2019).

[Gol01] O. Goldreich. *Foundations of Cryptography*. Cambridge University Press, 2001.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999. A preliminary version appeared in 21rst STOC, 1989.

[HLO24] Shuichi Hirahara, Zhenjian Lu, and Igor C Oliveira. One-way functions and pkt complexity. *Cryptology ePrint Archive*, 2024.

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *30th Annual Symposium on Foundations of Computer Science, Research Triangle Park, North Carolina, USA, 30 October - 1 November 1989*, pages 230–235. IEEE Computer Society, 1989.

[IL90] Russell Impagliazzo and Leonid A. Levin. No better ways to generate hard NP instances than picking uniformly at random. In *31st Annual Symposium on Foundations of Computer Science, St. Louis, Missouri, USA, October 22-24, 1990, Volume II*, pages 812–821. IEEE Computer Society, 1990.

[IRS21] Rahul Ilango, Hanlin Ren, and Rahul Santhanam. Hardness on any samplable distribution suffices: New characterizations of one-way functions by meta-complexity. *Electron. Colloquium Comput. Complex.*, TR21-082, 2021.

[IRS22] Rahul Ilango, Hanlin Ren, and Rahul Santhanam. Robustness of average-case meta-complexity via pseudorandomness. In Stefano Leonardi and Anupam Gupta, editors, *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 1575–1583. ACM, 2022.

[LP20] Yanyi Liu and Rafael Pass. On one-way functions and kolmogorov complexity. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1243–1254. IEEE, 2020.

[LP21] Yanyi Liu and Rafael Pass. On the possibility of basing cryptography on exp $\neq$ bpp exp$\neq$ bpp. In *Advances in Cryptology–CRYPTO 2021: 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16–20, 2021, Proceedings, Part I 41*, pages 11–40. Springer, 2021.

[LP23a] Yanyi Liu and Rafael Pass. On one-way functions and the worst-case hardness of time-bounded kolmogorov complexity. *Cryptology ePrint Archive*, 2023.

[LP23b] Yanyi Liu and Rafael Pass. One-way functions and the hardness of (probabilistic) time-bounded kolmogorov complexity wrt samplable distributions. In *Annual International Cryptology Conference*, pages 645–673. Springer, 2023.

[LS24] Zhenjian Lu and Rahul Santhanam. Impagliazzo's worlds through the lens of conditional kolmogorov complexity. In *51st International Colloquium on Automata, Languages, and Programming (ICALP 2024)*, pages 110–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024.

[RS21] Hanlin Ren and Rahul Santhanam. Hardness of kt characterizes parallel cryptography. *Cryptology ePrint Archive*, 2021.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il