Optimal Coding for Randomized Kolmogorov Complexity and Its Applications

Shuichi Hirahara[*]   Zhenjian Lu[†]   Mikito Nanashima[‡]

Abstract

The coding theorem for Kolmogorov complexity states that any string sampled from a computable distribution has a description length close to its information content. A coding theorem for resource-bounded Kolmogorov complexity is the key to obtaining fundamental results in average-case complexity, yet whether any samplable distribution admits a coding theorem for randomized time-bounded Kolmogorov complexity ($\mathsf{rK}^{\mathsf{poly}}$) is open and a common bottleneck in the recent literature of meta-complexity. Previous works bypassed this issue by considering probabilistic Kolmogorov complexity ($\mathsf{pK}^{\mathsf{poly}}$), in which public random bits are assumed to be available.

In this paper, we present an efficient coding theorem for randomized Kolmogorov complexity under the non-existence of one-way functions, thereby removing the common bottleneck. This enables us to prove $\mathsf{rK}^{\mathsf{poly}}$ counterparts of virtually all the average-case results that were proved only for $\mathsf{pK}^{\mathsf{poly}}$, and enables the resolution of the following concrete open problems.

1. The existence of a one-way function is characterized by the failure of average-case symmetry of information for randomized time-bounded Kolmogorov complexity, as well as a conditional coding theorem for randomized time-bounded Kolmogorov complexity. This resolves the open problem of Hirahara, Ilango, Lu, Nanashima, and Oliveira (STOC'23).

2. Hirahara, Kabanets, Lu, and Oliveira (CCC'24) showed that randomized time-bounded Kolmogorov complexity admits search-to-decision reductions in the errorless average-case setting over any samplable distribution, and left open whether a similar result holds in the error-prone setting. We resolve this question affirmatively, and as a consequence, characterize the existence of a one-way function by the average-case hardness of computing $\mathsf{rK}^{\mathsf{poly}}$ with respect to an arbitrary samplable distribution, which is an $\mathsf{rK}^{\mathsf{poly}}$ analogue of the $\mathsf{pK}^{\mathsf{poly}}$ characterization of Liu and Pass (CRYPTO'23).

The key technical lemma is that any distribution whose next bits are efficiently predictable admits an efficient encoding and decoding scheme, which could be of independent interest to data compression.

---

[*]National Institute of Informatics, Japan. E-mail: `s_hirahara@nii.ac.jp`

[†]University of Warwick, UK. E-mail: `zhenjian.lu@warwick.ac.uk`

[‡]Tokyo Institute of Technology, Japan. E-mail: `nanashima@c.titech.ac.jp`

Contents

1 Introduction

Shannon's source coding theorem is a centerpiece of information theory. It shows that if m independent samples are drawn from a distribution $\mathcal{D}$, then the m samples can be encoded into a string of expected length $m \cdot (\mathrm{H}(\mathcal{D}) + o(1))$, where $\mathrm{H}(\mathcal{D})$ denotes the Shannon entropy of $\mathcal{D}$. A computationally efficient variant of Shannon's source coding theorem was given by Impagliazzo and Zuckerman [IZ89], who showed that m independent samples drawn from any polynomial-time samplable distribution $\mathcal{D}$ can be (inefficiently) compressed into a string of expected length $m \cdot (\mathrm{H}(\mathcal{D}) + o(1))$ that can be decoded in polynomial time. Thus, the amortized encoding length of one string in the many samples from $\mathcal{D}$ approaches to $\mathrm{H}(\mathcal{D})$, which is information-theoretically optimal.

Less understood is a "one-shot" setting, in which *one* string x is drawn from a distribution $\mathcal{D}$, and the question is whether x has a short description. Information theoretically, for any distribution $\mathcal{D}$ over $\{0,1\}^*$, there exists an encoding scheme that compresses any string x in the support of a distribution $\mathcal{D}$ into a string of length $\log \frac{1}{\mathcal{D}(x)} + O(1)$, where $\mathcal{D}(x)$ denotes the probability that x is sampled from $\mathcal{D}$. In terms of Kolmogorov complexity, this result is often referred to as the *coding theorem for Kolmogorov complexity* (coined in [LO21]; it is also called a source compression theorem [Lee06]). It states that any string x in the support of a computable distribution $\mathcal{D}$ satisfies that

$$\mathsf{K}(x) \leq \log \frac{1}{\mathcal{D}(x)} + O(1),$$

where $\mathsf{K}(x)$ denotes the Kolmogorov complexity of x, i.e., the length of a shortest program that prints x, and the constant $O(1)$ depends only on the distribution $\mathcal{D}$. Note that Kolmogorov complexity does not impose any time bound on the time it takes to print x. This limits the applicability of the coding theorem in the literature of computational complexity theory. More relevant to complexity theory is a coding theorem for resource-bounded Kolmogorov complexity measures, such as $\mathsf{K}^t(x)$, i.e., the length of a shortest program that prints x in time t.

The coding theorem is one of the most fundamental properties of Kolmogorov complexity,[1] and is the key to establishing fundamental theorems of average-case complexity theory. For example, Levin [Lev86] initiated the theory of average-case NP-completeness by presenting a natural distributional problem which is complete for NP with respect to the class PComp of polynomial-time computable distributions. Here, a distribution is said to be *polynomial-time computable* if the cumulative distribution function is computable in polynomial time. Levin showed this completeness result by showing that PComp admits an *efficient coding theorem* for K^t, that is, that any string in the support of a polynomial-time computable distribution $\mathcal{D}$ can be compressed into a polynomial-time program of length $\log \frac{1}{\mathcal{D}(x)} + O(1)$ in polynomial time. We refer the reader to the survey of Bogdanov and Trevisan [BT06] for the background on average-case complexity theory.

Can we obtain a coding theorem for resource-bounded Kolmogorov complexity with respect to a wider class of distributions? The most standard class of distributions considered in the literature of average-case complexity theory is the class PSamp of (polynomial-time) samplable distributions. A distribution $\mathcal{D} = \{\mathcal{D}_n\}_{n \in \mathbb{N}}$ is said to be (*polynomial-time*) *samplable* if there exists a randomized polynomial-time algorithm that, on input 1^n, outputs a string that is distributed according to $\mathcal{D}_n$. Under a plausible derandomization assumption, Antunes and Fortnow [AF09] proved a coding theorem for $\mathsf{K}^{\mathsf{poly}}$ with respect to any samplable distribution. The assumption can be removed if we consider a randomized variant of $\mathsf{K}^{\mathsf{poly}}$ in which public random bits are given to short programs. The *t-time-bounded probabilistic Kolmogorov complexity* of a string x, denoted by pK^t [GKLO22], is

---

[1]According to Lee [Lee06], a coding theorem is one of the four "pillars" of Kolmogorov complexity.

defined to be the minimum integer k such that the probability that $\mathsf{K}^t(x \mid r) \leq k$ over a uniformly random $r \in \{0,1\}^t$ is at least $\frac{2}{3}$, where $\mathsf{K}^t(x \mid r)$ denotes the conditional Kolmogorov complexity of x given r, i.e., the length of a shortest program that prints x given r as input in time t. Lu, Oliveira, and Zimand [LOZ22] showed that any samplable distribution admits a coding theorem for probabilistic Kolmogorov complexity. Note that the notion of pK^t deviates from the standard notion of Kolmogorov complexity in that depending on the public random bits r, the shortest program that prints x on input r may be different. In fact, $\mathsf{pK}^{\mathsf{poly}}$ is essentially equivalent to (the logarithm of the reciprocal of) the time-bounded universal probability [HN23], which is technically useful but somewhat artificial from the perspective of data compression.[2]

A more natural randomized variant of time-bounded Kolmogorov complexity is *randomized Kolmogorov complexity*. The t-time-bounded randomized Kolmogorov complexity of a string x, denoted by $\mathsf{rK}^t(x)$, is defined to be the length of a shortest randomized program that prints x in time t with probability $\frac{2}{3}$ over the internal randomness of the randomized program. This is arguably more natural than $\mathsf{pK}^{\mathsf{poly}}$ in that the program is fixed irrespective of the random bits used by the program. It is evident that $\mathsf{pK}^t(x) \leq \mathsf{rK}^t(x) \leq \mathsf{K}^t(x)$, and thus the compression power of rK^t is in between pK^t and K^t. Partial progress towards obtaining coding theorems for randomized Kolmogorov complexity was made by Lu and Oliveira [LO21] and Lu, Oliveira, and Zimand [LOZ22], who proved a (information-theoretically sub-optimal) coding theorem for an exponential-time variant of $\mathsf{rK}^{\mathsf{poly}}$ (the randomized variant of Levin's Kt-complexity [Oli19]). No optimal coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ is known for any class of distributions larger than PComp. This leads us to the following question: *For which distributions (and when) does a coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ holds?*

Answering this question is indispensable for a closely related area of research — *data compression*. The main question investigated in the literature of data compression [GS91; TVZ05; Wee04; BSW03; HLR07; HMS23] is which class of distributions admits *efficient* coding theorems rather than (existential) coding theorems. The difference between the two types of the coding theorems is that in the latter, we do not care about the efficiency of an encoding algorithm. In an *efficient* coding theorem for a distribution $\mathcal{D}$, we require that there exists a *polynomial-time* algorithm that takes a string x drawn the distribution $\mathcal{D}$ and outputs a compressed string of length close to its information content $\log \frac{1}{\mathcal{D}(x)}$. Goldberg and Sipser [GS91] and Trevisan, Vadhan, and Zuckerman [TVZ05] identified several classes of distributions that admit efficient coding theorems, such as distributions samplable with logspace machines [TVZ05], high entropy sources [GS91; TVZ05], and samplable witness sets for NP [TVZ05]. However, no efficient coding theorem for any class of distributions that strictly contains PComp is known, just because even existential coding theorems for $\mathsf{rK}^{\mathsf{poly}}$ are unknown.

1.1 Interplay between One-Way Functions and Kolmogorov Complexity

Faced with the lack of a coding theorem for $\mathsf{rK}^{\mathsf{poly}}$, previous works in the recent literature of meta-complexity bypassed this issue by considering probabilistic Kolmogorov complexity $\mathsf{pK}^{\mathsf{poly}}$ or resource-unbounded Kolmogorov complexity K. There has been a flurry of new characterizations of the existence of one-way functions based on Kolmogorov complexity [LP20; RS21; LP21; IRS22; ACMTV21; LP22; LP23a; LP23b; HILNO23; Hir23; IL90; HN23], starting from the influential work of Liu and Pass [LP20]. A one-way function is one of the most fundamental cryptographic primitives because its existence is equivalent to the existence of a variety of cryptographic primitives, such as

---

[2]In terms of data compression, the difference between $\mathsf{rK}^{\mathsf{poly}}$ and $\mathsf{pK}^{\mathsf{poly}}$ can be explained as follows. In $\mathsf{pK}^{\mathsf{poly}}$, we assume that an inefficient encoding algorithm and an efficient decoding algorithm share random bits, which may not be the case in practice. In $\mathsf{rK}^{\mathsf{poly}}$, an efficient decoding algorithm is allowed to be randomized, but the random bits are private and not shared with an encoding algorithm.

a private-key encryption scheme [GM84], a pseudorandom generator [HILL99], a digital signature [Rom90], and a commitment scheme [Nao91]. The new "meta-computational" characterizations of one-way functions provide us with the hope that the improved understanding of one-way functions might lead us to the resolution of long-standing open problems, such as the elimination of Pessiland [Imp95] (i.e., does the average-case hardness of NP imply the existence of a one-way function?). Among the characterizations, we highlight the characterizations that are based on an arbitrary samplable distribution.

1. Hirahara, Ilango, Lu, Nanashima, and Oliveira [HILNO23] showed that average-case asymmetry of information for probabilistic Kolmogorov complexity $\mathsf{pK}^{\mathsf{poly}}$ characterizes the existence of a one-way function. That is, a one-way function can be constructed if and only if for *some* samplable distribution $\mathcal{D}$, the symmetry of information for $\mathsf{pK}^{\mathsf{poly}}$, i.e.,

$$\mathsf{pK}^{\mathsf{poly}}(x \mid y) + \mathsf{pK}^{\mathsf{poly}}(y) \approx \mathsf{pK}^{\mathsf{poly}}(x, y) \approx \mathsf{pK}^{\mathsf{poly}}(y \mid x) + \mathsf{pK}^{\mathsf{poly}}(x)$$

   does not hold with a non-negligible probability over (x, y) drawn from $\mathcal{D}$.

2. Impagliazzo and Levin [IL90] and Hirahara and Nanashima [HN23] showed that a one-way function exists if and only if approximating time-bounded universal probability is hard with respect to some samplable distribution.

3. Ilango, Ren, and Santhanam [IRS22] characterized the existence of a one-way function by the average-case hardness of Kolmogorov complexity K with respect to an arbitrary samplable distribution.

4. Liu and Pass [LP23b] characterized the existence of a one-way function by the average-case hardness of computing probabilistic Kolmogorov complexity $\mathsf{pK}^{\mathsf{poly}}$ with respect to an arbitrary samplable distribution.

These results provide fascinating approaches to construct one-way functions in that it suffices to construct *some* samplable distribution that witnesses asymmetry of information or the computational intractability of computing Kolmogorov complexity measures, which appears to be intuitively easier. However, the results do not extend to randomized Kolmogorov complexity $\mathsf{rK}^{\mathsf{poly}}$, precisely because of the lack of a coding theorem for $\mathsf{rK}^{\mathsf{poly}}$. Indeed, all the proofs of the results above rely on a coding theorem for corresponding Kolmogorov complexity measures; for example, the result of [IRS22] relies on the coding theorem for resource-unbounded Kolmogorov complexity K.

1.2 Our Results

In this paper, we identify a class of distributions that contains PComp and admits an efficient coding theorem for $\mathsf{rK}^{\mathsf{poly}}$. Roughly speaking, we present an efficient and information-theoretically optimal coding theorem for a distribution $\mathcal{D}$ if there exists a "next-bits predictor" for $\mathcal{D}$ in the sense that any next bit of a given arbitrary prefix is predictable with high accuracy in randomized polynomial time. This enables us to show that $\mathsf{rK}^{\mathsf{poly}}$, $\mathsf{pK}^{\mathsf{poly}}$ and K are all approximately equal to each other on average if one-way functions do not exist. In particular, we demonstrate that virtually all the average-case results that were previously shown to hold only for $\mathsf{pK}^{\mathsf{poly}}$ can be translated into $\mathsf{rK}^{\mathsf{poly}}$ counterparts. This enables us to resolve the main open problems left in previous works [HILNO23; HMS23; HKLO24]. We describe details below.

3

1.2.1 Symmetry of Information for rK$^{\mathsf{poly}}$ versus One-Way Functions

Symmetry of information for Kolmogorov complexity [ZL70] is one of the most fundamental properties of Kolmogorov complexity and is yet another one of the four "pillars" of Kolmogorov complexity [Lee06]. It states that for all strings x and y of length n,

$$\mathsf{K}(x \mid y) + \mathsf{K}(y) \approx \mathsf{K}(x, y) \approx \mathsf{K}(y \mid x) + \mathsf{K}(x),$$

where the approximate equality holds up to an additive $O(\log n)$ term. The original proof of symmetry of information due to Kolmogorov and Levin [ZL70] relies on an exhaustive search, and thus does not extend to the case of resource-bounded Kolmogorov complexity. As early as the 1960s, Kolmogorov suggested that it is an interesting avenue of research to investigate symmetry of information for time-bounded Kolmogorov complexity [LR05]. After a long line of research [ZL70; LM93; LW95; LR05; Hir21; Hir22; GK22; GKLO22], Hirahara, Ilango, Lu, Nanashima, and Oliveira [HILNO23] presented two characterizations of an average-case variant of symmetry of information:

1. The average-case asymmetry of information for pK$^{\mathsf{poly}}$ characterizes the existence of a one-way function.

2. The average-case asymmetry of information for rK$^{\mathsf{quasipoly}}$ characterizes the existence of a one-way function secure against *quasi-polynomial-time* algorithms.

It was left as a main open problem (highlighted by Osamu Watanabe in [HILNO23]) whether the failure of the average-case symmetry of information for rK$^{\mathsf{poly}}$ characterizes the existence of a *standard* one-way function (i.e., secure against polynomial-time algorithms).

We resolve this open problem affirmatively and obtain the following new characterization of the non-existence of one-way functions through the validity of symmetry of information for rK$^{\mathsf{poly}}$, as well as an average-case conditional coding theorem.

Theorem 1.1. *The following are equivalent.*

1. *One-way functions do not exist.*

2. **(Infinitely-Often Average-Case Symmetry of Information for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$, the following holds for all $t \geq p(n)$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^t(x \mid y) \leq \mathsf{rK}^t(x, y) - \mathsf{rK}^t(y) + \log t\right] \geq 1 - \frac{1}{q(n)}.$$

3. **(Infinitely-Often Average-Case Conditional Coding for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(n)}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n)\right] \geq 1 - \frac{1}{q(n)},$$

where $\mathcal{D}_n(x \mid y)$ denotes the probability that (x, y) is sampled from $\mathcal{D}_n$ conditioned that the second item being sampled is y.

4. (**Infinitely-Often Average-Case Efficient Conditional Coding for** rK^t) *For every polynomial-time samplable distribution family* $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ *supported over* $\{0,1\}^n \times \{0,1\}^n$, *and every polynomial* q, *there exists a polynomial* p *such that for infinitely many* $n \in \mathbb{N}$,

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(n)}(x \mid y) \leq \log\frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n)\right] \geq 1 - \frac{1}{q(n)}.$$

Moreover, it admits an efficient encoder in the following sense: there exists an efficient algorithm Enc *that outputs, for given* $(x,y) \sim \mathcal{D}_n$, *a description of a* $p(n)$-*time program* Π *of length at most* $-\log\mathcal{D}_n(x \mid y) + \log p(n)$ *with probability at least* $1 - 1/q(n)$ *over the choice of* $(x,y) \sim \mathcal{D}_n$ *and randomness for* Enc, *such that* Π *outputs* x *for given* y *and randomness* $r \sim \{0,1\}^{p(n)}$ *with probability at least* $2/3$ *over the choice of* r.

Our proof is fundamentally different from the previous proof of [HILNO23] for $\mathsf{rK}^{\mathsf{quasipoly}}$. The proof of [HILNO23] relies on the reconstructive extractors of Trevisan [Tre01] and Raz, Reingold, and Vadhan [RRV02], whose advice complexity of the reconstruction procedure is information-theoretically sub-optimal by an additive $O(\log^3 n)$ term, and this term is what forced [HILNO23] to consider quasi-polynomial-time one-way functions. Roughly speaking, the additive error term corresponds to the seed length of an extractor. Even without the reconstructive property, the state-of-the-art extractor construction due to Guruswami, Umans, and Vadhan [GUV09] has seed length $O(\log^2 n)$. Thus, in order to obtain Theorem 1.1 using the approach of [HILNO23], we would need to improve the seed length of the state-of-the-art extractor construction to $O(\log n)$. We sidestep this issue by taking a new approach based on the proof techniques of Goldberg and Sipser [GS91] and Trevisan, Vadhan, and Zuckerman [TVZ05].

Haitner, Mazor, and Silbak [HMS23] obtained an efficient coding theorem for $\mathsf{pK}^{\mathsf{poly}}$ with respect to any samplable distribution under the non-existence of a one-way function; that is, any samplable distribution $\mathcal{D}$ admits a polynomial-time encoding and decoding scheme of expected length $H(\mathcal{D}) + O(1)$ *when shared random bits are available*. Item 4 of Theorem 1.1 provides the same conclusion (up to an additive logarithmic term) *without any shared random bit*, which resolves the natural open problem left in [HMS23]. We also make progress towards the uniform version of the main result of [HMS23] by showing that any distribution incompressible to k bits without shared random bits has $(1-\epsilon)k - O(\log n)$ bits of uniform-next-bit pseudoentropy for any constant $\epsilon > 0$; we defer the details to Section 4.2.3.

Theorem 1.1 elucidates that the non-existence of a one-way function is both necessary and sufficient for the conditional version of an average-case coding theorem for $\mathsf{rK}^{\mathsf{poly}}$. Moreover, both efficient and existential coding theorems for $\mathsf{rK}^{\mathsf{poly}}$ are equivalent to each other (Items 3 and 4). We also mention that $\mathsf{rK}^{\mathsf{poly}}$, $\mathsf{pK}^{\mathsf{poly}}$, and K are all approximately equal to each other on average under the non-existence of a one-way function, which enables us to translate any average-case result about $\mathsf{pK}^{\mathsf{poly}}$ into an $\mathsf{rK}^{\mathsf{poly}}$ counterpart (unless a one-way function exists); see Lemma 5.2.

We also have an analogous result for *infinitely-often* one-way functions. In this case, using the notion of computational depth [AFMV06], we obtain a characterization for a *worst-case* variant of symmetry of information for $\mathsf{rK}^{\mathsf{poly}}$, which comes tantalizingly closer to the worst-case symmetry of information for $\mathsf{K}^{\mathsf{poly}}$ investigated by Longpré and Mocas [LM93] and Longpré and Watanabe [LW95]. A t-time-bounded computational depth $\mathsf{cd}^t(x)$ is defined as $\mathsf{pK}^t(x) - \mathsf{K}(x)$.

Theorem 1.2. *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*

2. **(Almost-Everywhere Average-Case Conditional Coding for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, there exists a polynomial p such that for all $n, k \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(n,k)}(x \mid y) \le \log \frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n,k)\right] \ge 1 - \frac{1}{k}.$$

3. **(Almost-Everywhere Worst-Case Conditional Coding for rK^t with Computational Depth)** *There exists a constant $c > 0$ such that the following holds. For every computable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, all $n, t \in \mathbb{N}$ such that $t \ge n$ and all $(x, y) \in \mathsf{Support}(\mathcal{D}_n)$,*

$$\mathsf{rK}^{(2^\alpha \cdot t)^c}(x \mid y) \le \log \frac{1}{\mathcal{D}_n(x \mid y)} + c \cdot (\log t + \alpha),$$

where $\alpha := \mathsf{cd}^t(x, y)$.

4. **(Almost-Everywhere Average-Case Symmetry of Information for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, there exists a polynomial p such that for all $n, k \in \mathbb{N}$ and $t \ge p(n,k)$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^t(x \mid y) \le \mathsf{rK}^t(x, y) - \mathsf{rK}^t(y) + \log t\right] \ge 1 - \frac{1}{k}.$$

5. **(Almost-Everywhere Worst-Case Symmetry of Information for rK^t with Computational Depth)** *There exists a constant $c > 0$ such that the following holds. For all $n, t \in \mathbb{N}$ such that $t \ge n$ and all $x, y \in \{0,1\}^n$,*

$$\mathsf{rK}^{(2^\alpha \cdot t)^c}(x \mid y) \le \mathsf{rK}^t(x, y) - \mathsf{rK}^t(y) + c \cdot (\log t + \alpha),$$

where $\alpha := \mathsf{cd}^t(x, y)$.

1.2.2 Error-Prone Average-Case Search-to-Decision Reductions for $\mathsf{rK}^{\mathsf{poly}}$

Next, we investigate the open question left by Hirahara, Kabanets, Lu, and Oliveira [HKLO24] and an $\mathsf{rK}^{\mathsf{poly}}$ counterpart of the $\mathsf{pK}^{\mathsf{poly}}$ characterization of Liu and Pass [LP23b]. Whether a search-to-decision reduction exists for the problem of computing time-bounded Kolmogorov complexity is a long-standing open problem that dates back to as early as the 1960s [Tra84], and recently there has been progress on this question [CIKK16; Hir18; Ila20; LP20; MP24; HKLO24]. Hirahara, Kabanets, Lu, and Oliveira [HKLO24] presented a search-to-decision reduction for computing $\mathsf{rK}^{\mathsf{poly}}$ in the errorless average-case setting: if there exists an efficient errorless average-case algorithm for computing $\mathsf{rK}^{\mathsf{poly}}$ on average, then there exists an efficient errorless average-case algorithm that finds a shortest randomized program of length $\mathsf{rK}^{\mathsf{poly}}(x)$ on a random input x drawn from an arbitrary samplable distribution. Designing such a reduction is well motivated by the fact that such a reduction is *necessary* for excluding (an errorless variant of) Pessiland from Impagliazzo's five worlds [Imp95]; see [HKLO24] for more details on the background.

The proof of [HKLO24] is based on a highly non-trivial combination of a reconstructive disperser of [Hir20] and a non-reconstructive disperser of [TUZ07], and does not extend to the *error-prone* average-case setting. Designing a similar reduction in the error-prone average-case setting was left as one of the main open questions in [HKLO24]. The difference between error-prone and errorless

average-case complexities [HS22; HN22] is that in the latter, an average-case algorithm is not allowed to make any error and instead allowed to indicate its failure of an algorithm, which is equivalent to the notion of average-polynomial-time [Imp95; BT06].

Using our new coding theorem, we present a search-to-decision reduction in the error-prone average-case setting, thereby answering the open problem of [HKLO24]. To state the result formally, we need a couple of definitions. For $\lambda \in [0, 1)$, let λ-MINrKT be the following promise problem (YES, NO):

$$\mathsf{YES} := \left\{ (x, 1^s, 1^t, 1^\ell) \mid \mathsf{rK}_\lambda^t(x) \leq s \right\},$$
$$\mathsf{NO} := \left\{ (x, 1^s, 1^t, 1^\ell) \mid \mathsf{rK}_{\lambda-1/\ell}^t(x) > s \right\}.$$

We say that an algorithm A *decides* (YES, NO) on input x if $x \in \mathsf{YES}$ implies $A(x) = 1$ and $x \in \mathsf{NO}$ implies $A(x) = 0$. The promise problem has a natural search version. For $x \in \{0, 1\}^n$, $t \in \mathbb{N}$ and $0 < \varepsilon, \lambda < 1$, we say that a randomized program M is an ε-rK_λ^t-*witness of* x if

- $|M| \leq \mathsf{rK}_\lambda^t(x)$, and

- M outputs x within t steps with probability at least $\lambda - \varepsilon$ over the internal randomness of M.

For $\lambda \in [0, 1)$, let λ-Search-MINrKT be the following search problem: Given $(x, 1^t, 1^\ell)$, where $x \in \{0, 1\}^*$, $t, \ell \in \mathbb{N}$, find an $(1/\ell)$-rK_λ^t-witness of x.

Theorem 1.3. *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*

2. **(Search-MINrKT is easy on average over polynomial-time samplable distributions)** *For every $\lambda \in [0, 1]$, every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n \in \mathbb{N}}$, where each $\mathcal{D}_n$ is over $\{0, 1\}^n$, there exist a polynomial ρ and a probabilistic polynomial-time algorithm A such that for all $n, s, \ell, k \in \mathbb{N}$, and all $t \geq \rho(n)$,*

$$\Pr_{x \sim \mathcal{D}_n, A}\left[A(x, 1^t, 1^\ell, 1^k) \text{ outputs an } (1/\ell)\text{-}\mathsf{rK}_\lambda^t\text{-witness of } x\right] \geq 1 - \frac{1}{k}.$$

3. **(MINrKT is easy on average over polynomial-time samplable distributions)** *For every $\lambda \in [0, 1)$, every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n \in \mathbb{N}}$, where each $\mathcal{D}_n$ is over $\{0, 1\}^n$, there exist a polynomial ρ and a probabilistic polynomial-time algorithm A such that for all $n, s, \ell, k \in \mathbb{N}$, and all $t \geq \rho(n)$,*

$$\Pr_{x \sim \mathcal{D}_n, A}\left[A(-, 1^k) \text{ decides } \lambda\text{-MINrKT } on \text{ input } (x, 1^s, 1^t, 1^\ell)\right] \geq 1 - \frac{1}{k}.$$

4. **(MINrKT is easy on average over the uniform distribution)** *There exist a polynomial ρ and a probabilistic polynomial-time algorithm A such that for all $n, s, \ell, k \in \mathbb{N}$,*

$$\Pr_{x \sim \{0,1\}^n, A}\left[A(-, 1^k) \text{ decides } (2/3)\text{-MINrKT } on \text{ input } (x, 1^s, 1^{\rho(n)}, 1^\ell)\right] \geq 1 - \frac{1}{k}.$$

This extends the $\mathsf{pK}^{\mathsf{poly}}$ characterization of one-way functions by Liu and Pass [LP23b] to the $\mathsf{rK}^{\mathsf{poly}}$ counterparts.

7

1.2.3 Optimal Coding Theorems for Next-Bits Predictable Distributions

The key lemma behind all the results above is an *unconditional* efficient coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ with respect to *next-bits predictable distributions*.

Definition 1.4 (See also Definition 4.1)**.** For a family $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ of distributions and a function $\varepsilon\colon \mathbb{N} \to (0,1)$, a *next-bits predictor for $\mathcal{D}$ with accuracy ε* is a randomized polynomial-time algorithm such that for every $n \in \mathbb{N}$, every $x \in \mathsf{Support}(\mathcal{D}_n)$, every $i \in \{1, \cdots, |x|\}$, and every $b \in \{0,1\}$,

$$\Pr_P\big[\mathcal{D}_n^*(b \mid x_{[i-1]}) - \varepsilon(n) \leq P(x_{[i-1]}, b, 1^n) \leq \mathcal{D}_n^*(b \mid x_{[i-1]}) + \varepsilon(n)\big] \geq 1 - \varepsilon(n),$$

where $\mathcal{D}_n^*(b \mid x_{[i-1]})$ denotes the probability, over $X \sim \mathcal{D}_n$, that the i-th bit of X is b conditioned that the first $(i-1)$-bits prefix of X is equal to that of x.[3] We say that $\mathcal{D}$ is *next-bits predictable* if for all polynomials q, there exists a next-bits predictor for $\mathcal{D}$ with accuracy $1/q$.

This definition should be compared with Yao's next-bit predictor [Yao82; Vad12]. There are three differences between Yao's *next-bit* predictor and our *next-bits* predictor.

1. For a distribution $\mathcal{D}$, Yao's next-bit predictor only predicts the i-th bit given the first $i-1$ bits of a random string x sampled from $\mathcal{D}$ for *some* index $i \in \{1, \cdots, |x|\}$. In contrast, the definition of a *next-bits* predictor requires that for *all* indices $i \in \{1, \cdots, |x|\}$, the i-th bit is predictable given the first $i-1$ bits of x .

2. We require that the accuracy of the prediction can be made arbitrarily small, whereas Yao's next-bit predictor is accurate with a non-negligible probability.

3. The output of Yao's next-bit predictor is considered to be correct if the bit produced by Yao's next-bit predictor is equal to the next bit, whereas a next-bits predictor aims to estimate the probability density function of the next bits. For example, the uniform distribution is next-bits predictable, but does not have Yao's next-bit predictor. In this sense, our notion of next-bits predictor is close in spirit to the notion of KL predictor of Vadhan and Zheng [VZ12].

For next-bits predictable distributions, we present an optimal and efficient coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ up to an additive $O(\log n)$ term. This extends the previous work of Levin [Lev86] because any polynomial-time computable distribution is next-bits predictable.

Theorem 1.5 (See also Theorem 4.13)**.** *For any next-bits predictable family $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ of distributions, and for every polynomial q, there exists an* efficient *encoding and decoding scheme whose expected encoding length is $\mathrm{H}(\mathcal{D}_n) + \log q(n)$; that is, there exists a pair $(\mathrm{Enc}, \mathrm{Dec})$ of randomized polynomial-time algorithms such that for every $n \in \mathbb{N}$,*

$$\mathop{\mathbf{E}}_{x\sim\mathcal{D}_n,\mathrm{Enc}}[|\mathrm{Enc}(1^n, x)|] \leq \mathrm{H}(\mathcal{D}_n) + \log p(n)$$

and for every $x \in \mathsf{Support}(\mathcal{D}_n)$,

$$\Pr_{\mathrm{Enc},\mathrm{Dec}}[\mathrm{Dec}(1^n, \mathrm{Enc}(1^n, x)) = x] \geq \frac{2}{3}.$$

---

[3]Throughout this paper, we only consider a family $\mathcal{D}$ of distributions such that every $x \in \mathsf{Support}(\mathcal{D}_n)$ has length at most $p(n)$ for some polynomial p.

Moreover, we obtain a *worst-case* coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ optimal up to a $(1 + \epsilon)$-factor for every constant $\epsilon > 0$, which is instrumental in Section 4.2.3.

Theorem 1.6 (See also Theorem 4.14)**.** *For any next-bits predictable family $\mathcal{D} = \{\mathcal{D}_n\}_{n \in \mathbb{N}}$ of distributions, for every $\epsilon > 0$, there exists a polynomial p such that for every $n \in \mathbb{N}$ and every $x \in \mathsf{Support}(\mathcal{D}_n)$,*

$$\mathsf{rK}^{p(n)}(x) \leq (1 + \epsilon) \log \frac{1}{\mathcal{D}_n(x)} + \log p(n).$$

These results could be interesting to practical data compression. Data compressors in practice (see [SM10]) work by predicting next symbols by a *deterministic* algorithm. Our results show that compression is possible even if a predictor is *randomized* and makes a small additive error.

Although it remains open whether an existential coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ with respect to PSAMP can be obtained unconditionally, we remark that the non-existence of a one-way function is necessary for an *efficient* coding theorem for $\mathsf{rK}^{\mathsf{poly}}$. Thus, Theorem 1.5 is unlikely to be extended to any samplable distribution.

Theorem 1.7. *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*

2. *For every polynomial-time samplable distribution $\mathcal{D} = \{\mathcal{D}_n\}_{n \in \mathbb{N}}$, there exists an efficient encoding and decoding scheme whose expected encoding length is $\mathrm{H}(\mathcal{D}_n) + O(\log n)$.*

3. *For any constant $\varepsilon > 0$, for every polynomial-time samplable distribution over $\{0, 1\}^n$ with entropy n^ε, there exists a polynomial-time encoding and decoding with expected length $n - 3$.*

2 Proof Overview

At a high level, our proof of symmetry of information for $\mathsf{rK}^{\mathsf{poly}}$ under the non-existence of one-way functions proceeds as follows.

1. To prove an efficient coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ with respect to next-bits predictable distributions (Theorem 1.5), we apply arithmetic encoding [cf. CT06, Sections 5.9 and 13.3] to a *randomized* next-bits predictor and then "pseudo-derandomize" the encoding by using the techniques of Goldberg and Sipser [GS91] and Trevisan, Vadhan, and Zuckerman [TVZ05].

2. Using the distributional inverter of Impagliazzo and Luby [IL89] (as in [IL90; HN23]), it can be shown that under the non-existence of one-way functions, any samplable distribution has a next-bits predictor (on average). This enables us to deduce the average-case efficient conditional coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ under the non-existence of one-way functions (Item 1 $\implies$ Item 4 in Theorem 1.1) from Theorem 1.5.

3. Average-case symmetry of information follows from the average-case conditional coding theorem, as in [HILNO23].

We emphasize the simplicity over the previous work [HILNO23], which we view as the strength of this work. Below, we explain the proof ideas of the coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ in Section 2.1 and then the proof of the search-to-decision reduction in Section 2.2.

2.1 Optimal Coding Theorems for rK^{poly}

The starting point of this work is the insightful work of Haitner, Mazor, and Silbak [HMS23], who showed that any distribution incompressible to k bits has $k - 2$ bits of next-bit pseudoentropy for non-uniform algorithms. Although we defer the definition of next-bit pseudoentropy to Section 4.2.3, their proof proceeds as follows.

1. If a distribution $\mathcal{D}$ does not have $k - 2$ bits of next-bit pseudoentropy, then by the work of Vadhan and Zheng [VZ12], there exists a "KL predictor" P that predicts any next bit with a reasonably high accuracy.

2. The predictor P induces a *polynomial-size-computable distribution* $\mathcal{D}^P$ (i.e., a non-uniform analogue of PCOMP) such that $\mathcal{D}^P$ and $\mathcal{D}$ are close in terms of KL divergence.

3. Applying the arithmetic encoding to $\mathcal{D}^P$ as in the work of Levin [Lev86], they obtain an efficient coding theorem for $\mathcal{D}^P$ and thus for $\mathcal{D}$; i.e., $\mathcal{D}$ admits an encoding and decoding scheme that can be computed by *polynomial-size circuits.*

The main idea of the proof of (Item 1 $\implies$ Item 4) in Theorem 1.1 is to replace the KL predictor in the proof of [HMS23] with the next-bits predictor that can be constructed from [IL89; IL90; HN23]. However, this poses a technical challenge: We consider a one-way function secure against *uniform algorithms*, in which case the next-bits predictor P is a randomized algorithm, and arithmetic encoding may not be applicable. This issue was also noted in [HMS23] and prevented Haitner, Mazor, and Silbak from obtaining the uniform version of their results when shared random bits are not available. To explain the issue briefly, let us explain how the arithmetic encoding works. Assuming that the cumulative distribution function $F_{\mathcal{D}}(x) := \sum_{y < x} \mathcal{D}(x)$ is efficiently computable by a deterministic algorithm, a string x can be encoded into the first $\lceil -\log \mathcal{D}(x) \rceil + 1$ bits of the value $F_{\mathcal{D}}(x) + \mathcal{D}(x)/2$. If $F_{\mathcal{D}}(x)$ is efficiently approximated by a *randomized* algorithm P, the arithmetic encoding of x may largely depend on the internal randomness of P. To address this issue, we would need a *pseudo-deterministic* algorithm that approximates $F_{\mathcal{D}}(x)$, i.e., an algorithm that produces a fixed approximate value for $F_{\mathcal{D}}(x)$ with high probability.

We pseudo-derandomize arithmetic encoding by using the techniques of "adding noise and rounding" developed by Goldberg and Sipser [GS91] and Trevisan, Vadhan, and Zuckerman [TVZ05], where they addressed similar issues to obtain randomized compression algorithms for flat sources over a language in P [GS91] and a witness set for NP [TVZ05]. The rough ideas are as follows. When we execute the next-bits predictor for approximating $F_{\mathcal{D}}(x)$ and $\mathcal{D}(x)$, we add a random noise and round the noised value to the nearest value in the integer multiples of a certain real value. As was shown in the previous works, the outcome of the next-bits predictor is fixed with high probability over the choice of the random noise, where the random noise is required to be shared between the encoder and decoder, but the description length of the random noise is logarithmically small. This idea enables us to make the next-bits predictor pseudodeterministic only with a logarithmic amount of shared randomness.

The techniques of "adding noise and rounding" can cause another issue with the accuracy of the next-bits predictor. More specifically, adding noise and rounding yield an additional accuracy error, and when the error is much larger than the next-bit probability, the accuracy of the approximation of the next-bit probability can become insufficient for decoding in arithmetic encoding. To address this issue, we avoid using the approximate value produced by the next-bits predictor when the next-bit probability is small, and in this case, we embed the next bit into the encoding with the position. We call a next bit with a small next-bit probability a *light next bit.* Namely, our encoding algorithm first determines whether the next bit is light by using the next-bits predictor and if

so, it embeds the next bit into the encoding; otherwise, it uses the next-bit prediction with the techniques of addition noise and rounding. Using these ideas, we can show that, for each string $x \in \{0,1\}^n$ in the support of the distribution $\mathcal{D}$, the length of the encoding is roughly at most $-\log \mathcal{D}(x) + O((\mathfrak{m}(x) + 1) \cdot \log n)$, where $\mathfrak{m}(x)$ is the number of the light next bits of x. We can easily observe that the number of the light next-bits of x is 0 with high probability over $x \sim \mathcal{D}$, which implies the optimal coding property for $\mathsf{rK^{poly}}$ with an efficient encoder. The same idea enables us to prove the *conditional* coding because the next-bits predictor can approximate the next-bit probability starting from any conditional string.

2.2 Error-Prone Average-Case Search-to-Decision Reductions for $\mathsf{rK^{poly}}$

We describe the proof ideas behind Theorem 1.3. First of all, it was implicitly shown in [LP20] that average-case tractability of (decisional) MINrKT over the uniform distribution implies the non-existence of one-way functions. Then, to show Theorem 1.3, it suffices to show that if one-way functions do not exist, then $\mathsf{Search\text{-}MINrKT}$ can be solved on average over polynomial-time samplable distributions.

At a high level, our proof follows the approach in [LP20], which shows that the non-existence of one-way functions implies the average-case tractability of $\mathsf{Search\text{-}MINKT}$ over the *uniform* distribution. Here $\mathsf{Search\text{-}MINKT}$ is the problem of finding a minimum t-time program (or, a K^t-witness) of a given string. In fact, their result can be generalized to any *polynomial-time computable* distribution. Next, we describe this approach in more detail.

Roughly put, the approach consists of the following steps:

1. Construct a function f such that if f can be inverted on average over uniformly random inputs of f, then one can obtain an average-case algorithm for finding K^t-witnesses, over the distribution where each x has probability mass $2^{-\mathsf{K}^t(x)}$.

2. Show that such an average-case algorithm also works for any fixed-polynomial-time computable distribution.

The authors of [LP20] construct a function f as follows: f takes an integer $i \in [n + O(1)]$, representing the length of a program, and a program $\Pi \in \{0,1\}^i$. It then obtains the output string x of Π after running for t steps. Finally, it outputs (i, x).

Let us first suppose that we can invert f in the worst case. Then, given x, one can find the smallest i^* such that (i^*, x) is inverted successfully, in which case we obtain a program of length i^* that outputs x within t steps. Such a program will be a K^t-witness of x.

In the case that we can invert f on average over uniformly random inputs, such a search algorithm will succeed on average over x sampled according to $\mathcal{D}_f$, which is the distribution induced by f (over uniformly random inputs). It is easy to observe that for each x, $\mathcal{D}_f$ will output x with probability at least about $2^{-\mathsf{K}^t(x)}$. In this case, we have that $\mathcal{D}_f$ *dominates*[4] the distribution Q^t, which is defined as the (semi-)distribution that assigns each x with probability mass $2^{-\mathsf{K}^t(x)}$. As a result, the average-case search algorithm that works for $\mathcal{D}_f$ also works for Q^t. This completes the description of the first step.

For the second step, we want to show that the same average-case search algorithm also works for any fixed-polynomial-time computable distribution $\mathcal{D}$. Again, it suffices to show that for a sufficiently large polynomial t, Q^t dominates $\mathcal{D}$. In other words, for every x, $2^{-\mathsf{K}^t(x)} \gtrsim \mathcal{D}(x)$. Note that this essentially follows from the known coding theorem for polynomial-time computable distributions with respect to the measure $\mathsf{K^{poly}}$ [Lev86].

---

[4]Recall that a distribution $\mathcal{D}$ dominates another distribution $\mathcal{D}'$ if $\mathcal{D}(x) \geq \mathcal{D}'(x)/\mathsf{poly}(n)$ for every x.

Next, we describe how to apply the above approach to obtain an average-case algorithm for finding an $(1/\ell)$-rK^t-witness of x while x is sampled over a *polynomial-time samplable* distribution. More specifically, we will do the following.

1. Construct a function f such that if f can be inverted on average over uniformly random inputs, then one can obtain an average-case algorithm for finding an $(1/\ell)$-rK^t-witnesses, over the distribution where each x has probability mass $2^{-rK^t(x)}$.

2. Show that such an average-case algorithm also works for any fixed-polynomial-time samplable distribution.

We will need new ideas in both steps described above.

Our construction of the function f is as follows: f takes an integer $i \in [n + O(1)]$, a *randomized program* $\Pi \in \{0, 1\}^i$, as well as a string $r \in \{0, 1\}^t$, which will be used as the internal randomness for running Π. We then obtain x, which is the output of Π running with randomness r after t steps. Now the key step here is that we will ensure that Π is a random program that outputs x with probability at least $2/3 - 1/\ell$. This is done using a randomized polynomial algorithm V with the following property: With high probability, for every (Π, x),

- if within t steps, Π outputs x with probability at least $2/3$, then V accepts, and

- if within t steps, Π outputs x with probability less than $2/3 - 1/\ell$, then V rejects.

(For the sake of simplicity in this proof overview, think of V as a deterministic algorithm.) Finally, if (Π, x) passes the test of V, we output (i, x). Otherwise, we output $\perp$.

The idea is that for such a function f, if we invert an image (i, x) successfully, then we will obtain a randomized program Π along with some randomness r such that Π, running on r for t steps, outputs x. Moreover, it holds that (Π, x) passes the test of V, which means Π outputs x with probability at least $2/3 - 1/\ell$. Therefore, by finding the smallest i^* such that (i^*, x) is inverted successfully, we can obtain an $(1/\ell)$-rK^t-witness of x.

Also, when considering the distribution $\mathcal{D}_f$ induced by f (over uniformly random inputs), note that for every x, if a rK^t-witness of x is picked, which happens with probability at least $\frac{1}{O(n)} \cdot 2^{-rK^t(x)}$, then after running Π for t steps, we will obtain x with probability at least $2/3$. Moreover, (Π, x) will pass the test of V. Therefore, for every x, $\mathcal{D}_f$ will output x with probability at least about $2^{-rK^t(x)}$.

As discussed previously, this allows us to obtain an average-case search algorithm for Search-MINrKT over the distribution Q^t, which assigns each x with probability mass $2^{-rK^t(x)}$. Now, to show that the same (average-case) search algorithm also works for any fixed-polynomial-time samplable distribution $\mathcal{D}$, it suffices to show that Q^t dominates $\mathcal{D}$. Another key observation here is that to show the former, we do not necessarily need Q^t to dominate $\mathcal{D}$ *in the worst case*; it suffices for Q^t to dominate $\mathcal{D}$ *on average*. In other words, for almost all $x \sim \mathcal{D}$, $2^{-rK^t(x)} \gtrsim \mathcal{D}(x)$. Then this follows from our average-case coding theorem for polynomial-time samplable distributions with respect to rK^{poly} under the non-existence of one-way functions (Item 1 $\implies$ Item 2 in Theorem 1.2).

3 Preliminaries

3.1 Notation

We use the notation ε to represent an empty string.

For a distribution $\mathcal{D}$ supported over $\{0,1\}^n \times \{0,1\}^n$ and $y \in \{0,1\}^n$, we let $\mathcal{D}(\cdot \mid y)$ denote the conditional distribution of $\mathcal{D}$ on the first half given that the second half is y.

For $n, n' \in \mathbb{N}$ with $n \leq n'$, let $[n : n'] = \{n, n+1, \ldots, n'\}$. Let $[n] := [1 : n] = \{1, \ldots, n\}$ for each $n \in \mathbb{N}$.

For a string $x \in \{0,1\}^*$ and $S \subseteq [|x|]$, let x_S denote a substring of x indicated by S, i.e., $x_S = x_{i_1} \circ \cdots \circ x_{i_k}$ for $S = \{i_1, \ldots, i_k\}$, where $i_1 < \cdots < i_k$. Particularly, $x_{[i]} = x_1 \circ \cdots \circ x_i$ and $x_{[i:j]} = x_i \circ x_{i+1} \circ \cdots \circ x_j$. For simplicity, let $x_{[0]} = \varepsilon$ for every $x \in \{0,1\}^*$.

For a distribution $\mathcal{D}$ over $\{0,1\}^*$ and strings $x, y \in \{0,1\}^*$, we define $\mathcal{D}^*(x \mid y) \in [0,1]$ as

$$\mathcal{D}^*(x \mid y) := \Pr_{z \sim \mathcal{D}}\left[z_{[|y|+1:|y|+|x|]} = x \big| z_{[|y|]} = y\right].$$

When $y = \varepsilon$, we drop "$|y$" from the notation above, i.e.,

$$\mathcal{D}^*(x) := \Pr_{z \sim \mathcal{D}}\left[z_{[|x|]} = x\right].$$

For every distribution $\mathcal{D}$ over $\{0,1\}^*$, every $x \in \{0,1\}^*$, and $k \in \mathbb{N}$, we use the notation $\mathsf{Next}(\mathcal{D}; x)$ to refer to the conditional distribution of the next bit of a subsequent string of x selected according to $\mathcal{D}$. Namely, for each $b \in \{0,1\}$,

$$\Pr_{b' \sim \mathsf{Next}(\mathcal{D};x)}[b' = b] = \mathcal{D}^*(b \mid x).$$

When we consider a *polynomial-time* decoding algorithm, the time bound is regarded as a function in the length of the original string before being encoded rather than the given encoding or input.

3.2 Useful Tools

Theorem 3.1 (Coding Theorem [Lev74])**.** *Let $\mathcal{E}$ be a distribution whose cumulative distribution function can be computed by some program p. Then for every $x \in \mathsf{Support}(\mathcal{E})$,*

$$\mathsf{K}(x \mid p) \leq \log \frac{1}{\mathcal{E}(x)} + O(1).$$

Lemma 3.2 (See, e.g., [HILNO23, Lemma 9])**.** *There exists a universal constant $b > 0$ such that for every distribution family $\{\mathcal{D}_n\}_{n \in \mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, every $n \in \mathbb{N}$, and every $y \in \{0,1\}^n$,*

$$\Pr_{x \sim \mathcal{D}_n(\cdot|y)}\left[\mathsf{K}(x \mid y) < \log \frac{1}{\mathcal{D}_n(x \mid y)} - \alpha\right] < \frac{n^b}{2^\alpha}.$$

Fact 3.3. *For every $x \in \{0,1\}^*$ and $t \in \mathbb{N}$,*

$$\mathsf{K}(x) \leq \mathsf{rK}^t(x).$$

Lemma 3.4 (Success Amplification for rK^t)**.** *For any string $x \in \{0,1\}^*$, time bound $t \in \mathbb{N}$, and $q \in \mathbb{N}$, we have*

$$\mathsf{rK}^{t'}_{1-1/q}(x) \leq \mathsf{rK}^t(x) + O(\log \log q),$$

where $t' := t \cdot O(\log q)$.

Lemma 3.5 (See, e.g., [HN23, Lemma 6.14]). *For every polynomial-time samplable distribution family* $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, *there exists a polynomial* ρ *such that for every* $n \in \mathbb{N}$, *every* $t \geq \rho(n)$, *and every* $\alpha \in \mathbb{N}$,

$$\Pr_{x\sim\mathcal{D}_n}\left[\mathsf{cd}^t(x) > \alpha\right] \leq 2^{-\alpha+O(\log n)}.$$

Lemma 3.6 (Implicit in [IL90; IL89]; see also [HN23]). *If almost everywhere* (*resp. infinitely-often*) *secure one-way functions do not exist, then for every samplable distribution family* $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, *where each* $\mathcal{D}_n$ *is over* $\{0,1\}^{\ell(n)}$ *for an efficiently computable* $\ell(n) \leq \mathsf{poly}(n)$, *and for every polynomial* p, *there exists a polynomial-time randomized algorithm* Ext *such that for infinitely many* (*resp. for all*) $n \in \mathbb{N}$,

$$\Pr_{x\sim\mathcal{D}_n}\left[\forall i \in [n],\ \Delta_{\mathsf{TV}}\big(\mathsf{Ext}\big(x_{[i-1]}, 1^n\big), \mathsf{Next}(\mathcal{D}_n; x_{[i-1]})\big) \leq \frac{1}{p(n)}\right] \geq 1 - \frac{1}{p(n)},$$

where $\Delta_{\mathsf{TV}}(,)$ *represents the total variation distance between two distributions.*

Particularly, we obtain the following.

Theorem 3.7. *If almost everywhere* (*resp. infinitely-often*) *secure one-way functions do not exist, then for every samplable distribution family* $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, *where each* $\mathcal{D}_n$ *is over* $\{0,1\}^n \times \{0,1\}^n$, *there exists a probabilistic polynomial-time algorithm* Ext *such that for all* $\varepsilon^{-1}, \delta^{-1} \in \mathbb{N}$ *and for infinitely many* (*resp. for all*) $n \in \mathbb{N}$,

$$\Pr_{y\sim\mathcal{D}_n^{(2)}}\left[\Delta_{\mathsf{TV}}\Big(\mathsf{Ext}(y; 1^{\varepsilon^{-1}}, 1^{\delta^{-1}}), \mathcal{D}_n(\cdot \mid y)\Big) \leq \varepsilon\right] \geq 1 - \delta,$$

where $\mathcal{D}_n^{(2)}$ *denotes the marginal distribution of the second element of* $\mathcal{D}_n$.

4   Coding for $\mathsf{rK}^{\mathsf{poly}}$ Based on Next-bits Prediction

In this section, we first present a meta-theorem showing that the approximation of next-bit probability yields a coding theorem for $\mathsf{rK}^{\mathsf{poly}}$ with an efficient encoder, where the encoding length can be worse than optimal depending on the number of *light next-bits* defined below. Then, we prove Theorems 4.11 and 4.12 as corollaries.

We introduce some notions to state the meta-theorem. First, we present the definition of next-bits prediction on a *subset* of the support. For a distribution family $\mathcal{D} = \{\mathcal{D}_n\}_n$ and a subset $S \subseteq \mathsf{Support}(\mathcal{D})$, we use the notation S_n to represent $S \cap \mathsf{Support}(\mathcal{D}_n)$ for each $n \in \mathbb{N}$ throughout the section.

Definition 4.1. Let $\mathcal{D} = \{\mathcal{D}_n\}_n$ be a distribution family over $\{0,1\}^*$. For $S \subseteq \mathsf{Support}(\mathcal{D})$ and a polynomial q, the distribution $\mathcal{D}$ is said to be next-bits-predictable on S with error parameter q if there exists a randomized polynomial-time algorithm P such that for every $n \in \mathbb{N}$, every $x \in S_n$, every $i \in [|x|]$, and every $b \in \{0,1\}$,

$$\Pr_{P}\big[P(x_{[i-1]}, b, 1^n) \in [\mathcal{D}_n^*(b \mid x_{[i-1]}) - 1/q(n), \mathcal{D}_n^*(b \mid x_{[i-1]}) + 1/q(n)]\big] \geq 1 - 1/q(n).$$

Next, we introduce the key notion of light next-bits, which affects the bound on the length of encoding in the meta-theorem.

14

Definition 4.2 (Light next-bit)**.** For a distribution $\mathcal{D}$ over $\{0,1\}^*$, $\delta \in [0,1]$, $x \in \mathsf{Support}(\mathcal{D})$, and $i \in [|x|]$, we say that $b \in \{0,1\}$ is a δ-light next-bit of $x_{[i-1]}$ (with respect to $\mathcal{D}$) if $\mathcal{D}^*(b \mid x_{[i-1]}) \le \delta$. Moreover, we say that x has a δ-light next-bit if there exists $i \in [|x|]$ such that x_i is a δ-light next-bit of $x_{[i-1]}$.

Definition 4.3 ($\mathfrak{m}_{\mathcal{D},\delta}$)**.** For a distribution $\mathcal{D}$ over $\{0,1\}^*$, $\delta \in [0,1]$, and $x \in \mathsf{Support}(\mathcal{D})$, we define $\mathfrak{m}_{\mathcal{D},\delta}(x)$ as the number of δ-light next-bits in x, i.e.,

$$\mathfrak{m}_{\mathcal{D},\delta}(x) = |\{i : x_i \text{ is a } \delta\text{-light next-bit of } x_{[i-1]}\}|.$$

For $j, j' \in [|x|]$ with $j < j'$, we also define $\mathfrak{m}_{\mathcal{D},\delta}^{j,j'}(x)$ as

$$\mathfrak{m}_{\mathcal{D},\delta}^{j,j'}(x) = |\{i \in [j, j'] : x_i \text{ is a } \delta\text{-light next-bit of } x_{[i-1]}\}|.$$

We often omit the subscript "$\mathcal{D}$" from $\mathfrak{m}_{\mathcal{D},\delta}$ and $\mathfrak{m}_{\mathcal{D},\delta}^{j,j'}$ when $\mathcal{D}$ is trivially identified in context. The following property immediately follows from the definition.

Proposition 4.4. *Let $\mathcal{D}$ be a distribution over $\{0,1\}^*$. For any $x \in \mathsf{Support}(\mathcal{D})$, the following hold:*

- *For every $i, j \in [|x|]$ with $i < j$ and every $\delta, \delta' \in [0,1]$ with $\delta \le \delta'$, $\mathfrak{m}_\delta^{i,j}(x) \le \mathfrak{m}_{\delta'}^{i,j}(x)$.*

- *For every $\delta \in [0,1]$ and every $i, j, i'j' \in [|x|]$ such that $[i : j] \subseteq [i' : j']$, $\mathfrak{m}_\delta^{i,j}(x) \le \mathfrak{m}_\delta^{i',j'}(x)$.*

A sample x has no δ-light next-bit if and only if $\mathfrak{m}_\delta(x) = 0$. It is easily observed that x has no light next-bit with respect to a distribution $\mathcal{D}$ with high probability over the choice of $x \sim \mathcal{D}$.

Proposition 4.5. *For every $n \in \mathbb{N}$, $\delta \in [0,1]$ and every distribution $\mathcal{D}$ over $\{0,1\}^n$,*

$$\Pr_{x \sim \mathcal{D}}[x \text{ has a } \delta\text{-light next-bit with respect to } \mathcal{D}] \le n\delta.$$

Proof. Sampling according to $\mathcal{D}$ can be performed sequentially as $x_i \sim \mathsf{Next}(\mathcal{D}; x_{[i-1]})$ for $i = 1, \ldots, n$ (in this order). For each i, the probability that a δ-light next-bit is sampled is at most δ by the definition. Therefore, by the union bound, the probability that the sample has a δ-light next-bit is at most $n \cdot \delta$. $\qquad\square$

Now, we formally state the meta-theorem.

Theorem 4.6. *For any distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, where $\mathcal{D}_n$ is over $\{0,1\}^{\ell(n)}$ for an efficiently computable function $\ell(n) \le \mathsf{poly}(n)$, and any polynomial q, there exists a polynomial p such that if $\mathcal{D}$ is next-bits-predictable on $S \subseteq \mathsf{Support}(\mathcal{D})$ with error parameter p, then for every $n \in \mathbb{N}$, every $x \in S_n$, and every $i \in [|x|]$*

$$\mathsf{rK}^{p(n)}(x_{[i:\ell(n)]} \mid x_{[i-1]}) \le -\log \mathcal{D}_n^*(x_{[i:\ell(n)]} \mid x_{[i-1]}) + \mathfrak{m}_{1/q(n)}^{i,\ell(n)}(x) \cdot O(\log \ell(n)) + O(\log n\ell(n)q(n)),$$

where the hidden constants in $O(\cdot)$ are independent of ℓ and q. In particular,

$$\mathsf{rK}^{p(n)}(x) \le -\log \mathcal{D}_n(x) + \mathfrak{m}_{1/q(n)}(x) \cdot O(\log \ell(n)) + O(\log n\ell(n)q(n)).$$

Moreover, there exists a polynomial-time randomized algorithm Enc that outputs, for given input $(x, 1^n, i)$, a description of a polynomial-time (randomized) program that outputs $x_{[i:\ell(n)]}$ when $x_{[i-1]}$ is given, and the description length satisfies the upper bound on $\mathsf{rK}^{p(n)}(x_{[i:\ell(n)]} \mid x_{[i-1]})$, where the success probability of Enc is at least $1 - 1/q(n)$.

In Section 4.1, we present the proof of Theorem 4.6. In Section 4.2, we derive Theorems 4.11 and 4.12 and the *almost* optimal worst-case coding theorem and *optimal* (average-case) coding theorem for next-bits-predictable source from Theorem 4.6.

15

4.1 Proof of Theorem 4.6

First, we make a next-bits predictor pseudo-deterministic with the help of *short advice* that depends on the input. Note that the auxiliary advice can be selected with high probability from uniformly random sampling but is required to be selected for each encoded string.

Lemma 4.7. *For every distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, where $\mathcal{D}_n$ is over $\{0,1\}^{\ell(n)}$ for an efficiently computable function $\ell(n) \leq \mathsf{poly}(n)$, and for every polynomial q, if $\mathcal{D}$ is next-bits-predictable on $S \subseteq \mathsf{Support}(\mathcal{D})$ with error parameter $32\ell(n)q(n)^3$, then there exists a polynomial-time randomized algorithm $\tilde{P}$ such that for every $n \in \mathbb{N}$ and every $x \in S_n$, with probability at least $1 - 1/q(n)$ over the choices of advice $\alpha_x \sim \{0, 1, \dots, 2\ell(n)q(n) - 1\} \subseteq \mathbb{N}$ and $2\ell(n)$ independent random seeds $r_1, r_1', \dots, r_{\ell(n)}, r_{\ell(n)}'$ for $\tilde{P}$, the following properties hold for all $i \in [\ell(n)]$ and all $b \in \{0,1\}$,*

1. *($\tilde{P}$ is pseudo-deterministic.)*

$$\tilde{P}(x_{[i-1]}, b, \alpha_x, 1^n; r_i) = \tilde{P}(x_{[i-1]}, b, \alpha_x, 1^n; r_i'); \tag{1}$$

2. *($\tilde{P}$ determines a distribution.)*

$$\tilde{P}(x_{[i-1]}, 0, \alpha_x, 1^n; r_i) + \tilde{P}(x_{[i-1]}, 1, \alpha_x, 1^n; r_i') = 1; \tag{2}$$

3. *($\tilde{P}$ is accurate.)*

$$\mathcal{D}_n^*(b \mid x_{[i-1]}) - 1/q(n)^2 \leq \tilde{P}(x_{[i-1]}, b, \alpha_x, 1^n; r_i) \leq \mathcal{D}_n^*(b \mid x_{[i-1]}) + 1/q(n)^2. \tag{3}$$

Particularly, if b is not an $(1/q(n))$-light next-bit of $x_{[i-1]}$, then

$$(1 - 1/q(n))\mathcal{D}_n^*(b \mid x_{[i-1]}) \leq \tilde{P}(x_{[i-1]}, b, \alpha_x, 1^n; r_i) \leq (1 + 1/q(n))\mathcal{D}_n^*(b \mid x_{[i-1]}).$$

For convenience, we call the polynomial q above a *mordified error parameter*.

Proof. Since $\mathcal{D}$ is next-bits-predictable on S, there exists a polynomial-time next-bits predictor P satisfying the properties of Definition 4.1 with error parameter $q'(n) := 32\ell(n)q(n)^3$.

For given input $(x_{[i-1]}, b, \alpha_x, 1^n)$, where $\alpha_x \sim \{0, 1, \dots, 2\ell(n)q(n) - 1\} \subseteq \mathbb{N}$, the algorithm $\tilde{P}$ first executes $v_0 \leftarrow P(x_{[i-1]}, 0, 1^n)$. Then, $\tilde{P}$ applies the technique of adding noise and rounding to be pseudo-deterministic. Here, the amount of noise is $\alpha_{x,\delta} \cdot 1/(8\ell(n)q(n)^3)$. Let $\tilde{v}_0$ be the nearest value to $v_0 + \alpha_x \cdot 1/(8\ell(n)q(n)^3)$ in multiples of $1/(4q(n)^2)$ in $[0,1]$ (i.e., $\tilde{v}_0 = N \cdot 1/(4q(n)^2)$) for some $N \in \{0, 1, \dots, 4q(n)^2\}$), where ties are broken by choosing the smaller one.

If the input b is 0, the algorithm $\tilde{P}$ outputs $\tilde{v}_0$; otherwise (i.e., if $b = 1$), $\tilde{P}$ outputs $\tilde{v}_1 := 1 - \tilde{v}_0$. Note that $\tilde{P}$ uses its internal randomness only for executing P.

We show the three properties in the lemma. Recall that $\alpha_x \sim \{0, \dots, 2\ell(n)q(n) - 1\}$. For each $i \in [\ell(n)]$, we consider the execution of $\tilde{P}(x_{[i-1]}, b, \alpha_x, 1^n)$ with the global advice α_x. Let v_0 be the value produced by $P(x_{[i-1]}, 0, 1^n)$ during the execution.

Suppose that P does not fail in the sense that

$$\left| v_0 - \mathcal{D}_n^*(0 \mid x_{[i-1]}) \right| \leq 1/q'(n) = 1/(32\ell(n)q(n)^3).$$

Recall that the property of P shows that the event above occurs with probability at least $1 - 1/q'(n)$ as long as $x \in S_n$.

16

We define $I_{x,i} \subseteq [0,1]$ as

$$I_{x,i} := [\mathcal{D}_n^*(0 \mid x_{[i-1]}) - 1/(32\ell(n)q(n)^3), \mathcal{D}_n^*(0 \mid x_{[i-1]}) + 1/(32\ell(n)q(n)^3)] \cap [0,1].$$

Then, $v_0 \in I_{x,i}$ as long as P is performed successfully. Note that $|I_{x,i}| \le 1/(16\ell(n)q(n)^3)$, and $I_{x,i}$ is independent of α_x.

Notice that (i) the noise $\alpha_x \cdot 1/(8\ell(n)q(n)^3)$ varies in $1/(8\ell(n)q(n)^3)$ ($> |I_{x,i}|$) increments, and (ii) the rounded value $\tilde{v}_0$ varies in $1/(4q(n)^2)$ ($> (2\ell(n)q(n) - 1) \cdot 1/(8\ell(n)q(n)^3)$) increments. Thus, the number of α_x for which there exist two values $v_0, v_0' \in I_{x,i}$ that are rounded to two distinct values is at most 1.

Since the same argument holds for all $i \in [\ell(n)]$, we have that with probability at least $1 - \ell(n)/(2\ell(n)q(n)) = 1 - 1/(2q(n))$ over the choice of α_x, for all $i \in [\ell(n)]$, the rounded value $\tilde{v}_0$ for given $x_{[i-1]}$ always takes the same value as long as P is successfully executed. Below, we observe the three properties in the lemma under the events that (i) such a good value of α_x is selected, and (ii) all of the $2\ell(n)$ executions of P are successfully performed. This completes the proof of the lemma because, by the union bound, the two events occur simultaneously with probability at least $1 - 1/(2q(n)) - 2\ell(n)/q'(n) \ge 1 - 1/q(n)$.

The first property (Equation (1)) has already verified because the rounded value $\tilde{v}_0$ always takes the same value as long as P is successfully executed, and $\tilde{P}$ outputs either of $\tilde{v}_0$ and $\tilde{v}_1 = 1 - \tilde{v}_0$ depending on b.

Next, we observe the second property. Let $\tilde{v}_0$ and $\tilde{v}_1$ be the values produced by $\tilde{P}$ given $b = 0$ and $b = 1$, respectively, for fixed randomness r. By the construction of $\tilde{P}$, they always satisfy $\tilde{v}_0 + \tilde{v}_1 = 1$. Let $\tilde{v}_0'$ and $\tilde{v}_1'$ be the values produced by $\tilde{P}$ given $b = 0$ and $b = 1$, respectively, for fixed randomness r' different from r. The first property implies that $\tilde{v}_0 = \tilde{v}_0'$, and it further implies that $\tilde{v}_0' + \tilde{v}_1 = \tilde{v}_0 + \tilde{v}_1 = 1$. Therefore, Equation (2) holds under the same events.

Finally, we observe the third property and complete the proof. Recall that $\left|v_0 - \mathcal{D}_n^*(0 \mid x_{[i-1]})\right| \le 1/(32\ell(n)q(n)^3)$ under the condition. Adding the noise ($\alpha_{x,\delta} \cdot 1/(8\ell(n)q(n)^3)$) and rounding to the multiples of $1/(4q(n)^2)$ only changes the value at most

$$2\ell(n)q(n) \cdot \frac{1}{8\ell(n)q(n)^3} + \frac{1}{4q(n)^2} = \frac{1}{2q(n)^2}.$$

Thus, we have that $\left|\tilde{v}_0 - \mathcal{D}_n^*(0 \mid x_{[i-1]})\right| \le 1/(32\ell(n)q(n)^3) + 1/(2q(n)^2) \le 1/q(n)^2$. Notice that

$$\left|\tilde{v}_1 - \mathcal{D}_n^*(1 \mid x_{[i-1]})\right| = \left|\tilde{v}_0 - \mathcal{D}_n^*(0 \mid x_{[i-1]})\right| \le 1/q(n)^2.$$

Therefore, in any case,

$$\left|\tilde{v}_b - \mathcal{D}_n^*(b \mid x_{[i-1]})\right| \le \frac{1}{q(n)^2},$$

and Equation (3) holds because P outputs $\tilde{v}_b$ for the given b. In addition, if b is not an $(1/q(n))$-light next-bit of $x_{[i-1]}$, then

$$\left|\tilde{v}_b - \mathcal{D}_n^*(b \mid x_{[i-1]})\right| \le \frac{1}{q(n)^2} \le \frac{1}{q(n)}\mathcal{D}_n^*(b \mid x_{[i-1]}).$$

By rearranging the above,

$$(1 - 1/q(n))\mathcal{D}_n^*(b \mid x_{[i-1]}) \le \tilde{v}_b \le (1 + 1/q(n))\mathcal{D}_n^*(b \mid x_{[i-1]}),$$

as desired. $\qquad\square$

Next, we use the *modified* next-bits predictor $\tilde{P}$ for the arithmetic encoding (i.e., Shannon–Fano–Elias coding) to obtain the coding theorem for $\mathsf{rK}^{\mathsf{poly}}$.

Let $\mathcal{D} = \{\mathcal{D}_n\}$ be a distribution family that is next-bits-predictable on $S \subseteq \mathsf{Support}(\mathcal{D})$, where $\mathcal{D}_n$ is over $\{0,1\}^{\ell(n)}$. The encoding and decoding algorithms are the following, where $q(n)$ represents an arbitrary polynomial and $\tilde{P}$ represents the algorithm in Lemma 4.7 with *modified* error parameter $q'(n) := \ell(n)q(n)+1$. We only consider the encoding of $x_{[k:\ell(n)]}$ given $x_{[k-1]}$ for $x \in S_n$ and $k \in [\ell(n)]$.

Note that, at the end of each round i, the values of the variables $p_<$ and $p_=$ in the encoding and decoding algorithms are excepted to be the approximations of $\sum_{y<x_{[k:i]}} \mathcal{D}^*(y \mid x_{[k-1]})$ and $\mathcal{D}^*(x_{[k:i]} \mid x_{[k-1]})$, respectively. However, the algorithms ignore the round i when x_i is a light next-bit of x_{i-1} (i.e., the next-bit probability of x_i is regarded to be 1) and embed x_i to the encoding with the position i. The way of update is based on the following expressions:

$$\sum_{y<x_{[k:i]}} \mathcal{D}^*(y \mid x_{[k-1]}) = \begin{cases} \sum_{y<x_{[k:i-1]}} \mathcal{D}^*(y \mid x_{[k-1]}) & \text{if } x_i = 0 \\ \sum_{y<x_{[k:i-1]}} \mathcal{D}^*(y \mid x_{[k-1]}) + \mathcal{D}^*(x_{[k:i-1]} \mid x_{[k-1]}) \cdot \mathcal{D}^*(0 \mid x_{[i-1]}) & \text{if } x_i = 1, \end{cases}$$

and

$$\mathcal{D}^*(x_{[k:i]} \mid x_{[k-1]}) = \mathcal{D}^*(x_{[k:i-1]} \mid x_{[k-1]}) \cdot \mathcal{D}^*(x_i \mid x_{[i-1]}).$$

Algorithm 1: $\mathrm{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]})$

Input: $x_{[k:\ell(n)]} \in \{0,1\}^{\ell(n)-k}$, $n \in \mathbb{N}$, and a conditional string $x_{[k-1]} \in \{0,1\}^{k-1}$, where $x \in S_n$.

1 Let $p_< := 0$ and $p_= := 1$;
2 Select $\alpha \sim \{0, \ldots, 2\ell(n)q'(n) - 1\}$ uniformly at random;
3 Let L be an empty list (where the element is expected to be in $[\ell(n)] \times \{0,1\}$);
4 **for** $i := k$ **to** $\ell(n)$ **do**
5   Execute $q_i \leftarrow \tilde{P}(x_{[i-1]}, x_i, \alpha, 1^n)$;
6   **if** $q_i \leq 2/q'(n)$ **then**
7   Add $(i, x_i) \in [\ell(n)] \times \{0,1\}$ to L and go to the next loop;
8   **end**
9   **if** $x_i = 1$ **then** $p_< := p_< + p_= \cdot \tilde{P}(x_{[i-1]}, 0, \alpha, 1^n)$;
10     $p_= := p_= \cdot q_i$;
11 **end**
12 Let v be the first $\lceil -\log p_= \rceil + 1$ bits of $p_< + (p_=/2)$;
13 **return** (v, L, α, n, k);

It is easily observed that Enc_q and Dec_q halt in polynomial time in n since $\tilde{P}$ is polynomial time. We show that the encoding and decoding algorithms above work with high probability over the choice of *independent* random seeds and estimate the length of the encoding.

Lemma 4.8. *For every distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, where $\mathcal{D}_n$ is over $\{0,1\}^{\ell(n)}$ for an efficiently computable function $\ell(n) \leq \mathsf{poly}(n)$, and every polynomial q, if $\mathcal{D}$ is next-bits-predictable on $S \subseteq \mathsf{Support}(\mathcal{D})$ with error parameter $32\ell(n)q'(n)^3 (= 32\ell(n)^4q(n)^3)$, then for every $n \in \mathbb{N}$, every $x \in S_n$, and every $k \in [|x|]$, it holds that*

$$\Pr_{\mathrm{Enc}_q, \mathrm{Dec}_q}\left[\mathrm{Dec}_q(\mathrm{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]}); x_{[k-1]}) = x_{[k:\ell(n)]}\right] \geq 1 - \frac{1}{q(n)\ell(n)};$$

Algorithm 2: $\mathrm{Dec}_q(v, L, \alpha, n; x_{[k-1]})$

Input: an encoding $(v, \alpha_{x,\delta}, n, \delta)$ and a conditional string $x_{[k-1]} \in \{0,1\}^{k-1}$.

1 Let $p_< := 0$ and $p_= := 1$;

2 Let $\tilde{x} := x_{[k-1]}$;

3 for $i := k$ *to* $\ell(n)$ **do**

4   **if** $(i, b) \in L$ *for some* $b \in \{0,1\}$ **then**

5   Update $\tilde{x} := \tilde{x} \circ b$ and go to the next loop;

6   **end**

7   Compute $q_0 = \tilde{P}(\tilde{x}_{[i-1]}, 0, \alpha, 1^n)$ and $q_1 = 1 - q_0$;

8   **if** $v \geq p_< + p_= \cdot q_0$ **then** let $\tilde{x} := \tilde{x} \circ 1$, $p_< := p_< + p_= \cdot q_0$, and $p_= = p_= \cdot q_1$;

9   **else** $\tilde{x} := \tilde{x} \circ 0$ and $p_= := p_= \cdot q_0$;

10 end

11 return $\tilde{x}_{[k:\ell(n)]}$;

---

and

$$\Pr_{\mathrm{Enc}_q} \left[|\mathrm{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]})| \leq -\log \mathcal{D}_n(x) + \mathfrak{m}_{1/q(n)}^{k,\ell(n)}(x) \cdot O(\log \ell(n)) + O(\log n\ell(n)q(n)) \right]$$

$$\geq 1 - \frac{1}{q(n)\ell(n)},$$

where the hidden constants in $O(\cdot)$ are universal independent of ℓ and q.

In particular, for a sufficiently large polynomial p,

$$\mathsf{rK}^{p(n)}(x_{[k:\ell(n)]} \mid x_{[k-1]}) \leq -\log \mathcal{D}_n(x) + \mathfrak{m}_{1/q(n)}^{k,\ell(n)}(x) \cdot O(\log \ell(n)) + O(\log n\ell(n)q(n)),$$

and there exists a polynomial-time randomized algorithm Enc *that outputs, for given input $(x, 1^n, k)$, a description of a polynomial-time (randomized) program that outputs $x_{[k:\ell(n)]}$ when $x_{[k-1]}$ is given, and the description length satisfies the upper bound on $\mathsf{rK}^{p(n)}(x_{[k:\ell(n)]} \mid x_{[k-1]})$ above, where the success probability of* Enc *is at least $1 - 6/(q(n)\ell(n))$.*

Note that Theorem 4.6 immediately follows from Lemma 4.8.

Proof. Lemma 4.8 follows from the correctness of the arithmetic encoding and Lemma 4.7.

First, we observe that, in the execution of $\mathrm{Dec}_q(\mathrm{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]}); x_{[k-1]})$, the next-bits predictor $\tilde{P}$ is executed only on $(x_{[i-1]}, b, \alpha, 1^n)$ for $i \in [\ell(n)]$ and $b \in \{0,1\}$ as long as the sequential decoding is performed along x (i.e., $\tilde{x} = x_{[i]}$ for each stage i). Note that Enc_q and Dec_q do not share the randomness for executing $\tilde{P}$; thus, they may execute $\tilde{P}$ on the same input but using independent randomness. However, Lemma 4.7 shows that, with probability at least $1 - 1/(q(n)\ell(n))$ over the choice of α and randomness for $\tilde{P}$ (i.e., over the randomness for Enc_q and Dec_q), all the executions of $\tilde{P}$ yield consistent values and determine the conditional distribution of each next bit. In this case, $\mathrm{Dec}_q(\mathrm{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]}); x_{[k-1]})$ performs the arithmetic encoding [cf. CT06, Sections 5.9 and 13.3] for the distribution induced by executing $\tilde{P}$ on each prefix of x except for the positions i on which the value of q_i is less than $2/q'(n)$. Namely, under the good choices of α and randomness as indicated in Lemma 4.7, the value v produced by Enc_q satisfies that

$$p_<^{\mathrm{Enc}} \leq p_<^{\mathrm{Enc}} + p_=^{\mathrm{Enc}}/2 - 2^{-\lceil -\log p_=^{\mathrm{Enc}} \rceil - 1} < v < p_<^{\mathrm{Enc}} + p_=^{\mathrm{Enc}},$$

19

where $p_<^{\mathrm{Enc}}$ and $p_=^{\mathrm{Enc}}$ represent the values of variables $p_<$ and $p_=$ at the end of the execution of Enc_q, respectively. In addition, for every round i in the execution of Dec_q,

$$\begin{cases} v < p_<^{\mathrm{Enc}} + p_=^{\mathrm{Enc}} \leq p_< + p_= \cdot q_0 & \text{if } x_i = 0 \\ v \geq p_<^{\mathrm{Enc}} \geq p_< + p_= \cdot q_0 & \text{if } x_i = 1, \end{cases}$$

where $p_<, p_=$, and q_0 represent the variables computed in Dec_q. Thus, whenever $q_i > 2/q'(n)$, Dec_q successfully decodes the i-th next bit. In the other case where $q_i \leq 2/q'(n)$, the pair (i, x_i) is contained in the list L, and Dec_q also successfully decodes the next bit. Therefore, we obtain that

$$\Pr_{\mathrm{Enc}_q, \mathrm{Dec}_q} \left[\mathrm{Dec}_q(\mathrm{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]}); x_{[k-1]}) = x_{[k:\ell(n)]} \right] \geq 1 - \frac{1}{q(n)\ell(n)}.$$

Next, we evaluate the length of the encoding. Below we let $p_=$ represent the value of the variable $p_=$ at the end of the execution of Enc_q. We also assume that the randomness for Enc_q (i.e., α and randomness for executing $\tilde{P}$) satisfies the condition of Lemma 4.7. This event occurs with probability at least $1 - 1/(\ell(n)q(n))$.

Let L be the set of indices i such that $q_i \leq 2/q'(n)$ in the execution of Enc_q, and let $H = [\ell(n)] \setminus L$. Recall that the variable $p_=$ is updated only when $i \in H$. We observe that for every $i \in L$, the i-th bit x_i is a $3/q'(n)$-light next bit of $x_{[i-1]}$ because

$$\mathcal{D}^*(x_i \mid x_{[i-1]}) \leq q_i + \frac{1}{q'(n)^2} \leq \frac{2}{q'(n)} + \frac{1}{q'(n)^2} \leq \frac{3}{q'(n)}.$$

In addition, for every $i \in H$, the i-th bit x_i is not an $(1/q'(n))$-light next bit of $x_{[i-1]}$ because

$$\mathcal{D}^*(x_i \mid x_{[i-1]}) \geq q_i - \frac{1}{q'(n)^2} > \frac{2}{q'(n)} - \frac{1}{q'(n)^2} \geq \frac{1}{q'(n)}.$$

Without loss of generality, we assume that $(\lceil -\log p_= \rceil + 1) = O(\ell(n))$; otherwise, we can replace the encoding for x with the canonical encoding of length $\ell(n) + O(1)$ by embedding x. In addition, we assume that $\ell(n) \geq 3$. Then, by the standard prefix-free encoding, the output (v, L, α, n, k) of Enc_q is represented in

$$\lceil -\log p_= \rceil + 1 + |H| \cdot O(\log \ell(n)) + O(\log(\lceil -\log p_= \rceil + 1)) + O(\log nq(n)\ell(n))$$
$$= -\log p_= + \mathfrak{m}_{3/q'(n)}^{k,\ell(n)}(x) \cdot O(\log \ell(n)) + O(\log nq(n)\ell(n))$$
$$\leq -\log p_= + \mathfrak{m}_{1/q(n)}^{k,\ell(n)}(x) \cdot O(\log \ell(n)) + O(\log n\ell(n)q(n)) \qquad \text{bits,}$$

where all constants in $O(\cdot)$ notations are independent of q and ℓ, and we used

$$3/q'(n) \leq 3/(\ell(n)q(n)) \leq 1/q(n) \quad \text{and} \quad \mathfrak{m}_{3/q'(n)}^{k,\ell(n)}(x) \leq \mathfrak{m}_{1/q(n)}^{k,\ell(n)}(x).$$

Lemma 4.7 further shows that for every $i \in H$,

$$(1 - 1/q'(n))\mathcal{D}_n^*(x_i \mid x_{[i-1]}) \leq \tilde{P}(x_{[i-1]}, b, \alpha, 1^n, 1^{\delta^{-1}}) \leq (1 + 1/q'(n))\mathcal{D}_n^*(x_i \mid x_{[i-1]})$$

since x_i is not an $(1/q'(n))$-light next bit of $x_{[i-1]}$ whenever $i \in H$.

Therefore, at the end of the execution of Enc_q, the variable $p_=$ takes the value

$$
\begin{aligned}
p_= &= \prod_{i \in H} \tilde{P}(x_{[i-1]}, x_i, \alpha, 1^n) \\
&\geq (1 - 1/(\ell(n)q(n) + 1))^{|H|} \cdot \prod_{i \in H} \mathcal{D}_n^*(x_i \mid x_{[i-1]}) \\
&\geq (1 - 1/(\ell(n)q(n) + 1))^{\ell(n)} \cdot \prod_{i=1}^{\ell(n)} \mathcal{D}_n^*(x_i \mid x_{[i-1]}) \\
&= (1 - 1/(\ell(n)q(n) + 1))^{\ell(n)} \cdot \mathcal{D}(x) \\
&\geq e^{-1/q(n)} \cdot \mathcal{D}(x).
\end{aligned}
$$

Thus, we have

$$
-\log p_= \leq -\log \mathcal{D}(x) + 1/q(n) + \log e \leq -\log \mathcal{D}(x) + 3.
$$

Therefore, the length of the encoding is at most

$$
-\log p_= + \mathfrak{m}_{1/q(n)}^{k,\ell(n)}(x) \cdot C \log \ell(n) + C \cdot \log n\ell(n)q(n)
$$
$$
\leq -\log \mathcal{D}(x) + \mathfrak{m}_{1/q(n)}^{k,\ell(n)}(x) \cdot C \log \ell(n) + C \log n\ell(n)q(n) + 3,
$$

where C is a universal constant independent of ℓ and q.

The final statement on rK is based on the following probabilistic argument: By the union bound, with probability at least $1 - 2/(\ell(n)q(n))$ over the choice of randomness for Enc_q and Dec_q, it holds that (i) the length of the output of Enc_q is at most $-\log \mathcal{D}(x) + \mathfrak{m}_{1/q(n)}^{k,\ell(n)}(x) \cdot O(\log \ell(n)) + O(\log n\ell(n)q(n))$, and (ii) $\mathsf{Dec}_q(\mathsf{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]}); x_{[k-1]}) = x_{[k:\ell(n)]}$. By Markov's inequality, with probability at least $1 - 6/(\ell(n)q(n))$ over the randomness for Enc_q, (i) the length of the encoding satisfies the same bound, and (ii) $\mathsf{Dec}_q(\mathsf{Enc}_q(x_{[k:\ell(n)]}, n; x_{[k-1]}); x_{[k-1]})$ produces $x_{[k:\ell(n)]}$ with probability at least $2/3$ over the randomness for Dec_q. Since Enc_q and Dec_q are polynomial-time algorithms and uniform, the statement on rK holds. $\qquad\square$

4.2 Corollaries

We derive Theorems 4.11 and 4.12 from Theorem 4.6 in Section 4.2.1 and show Theorem 1.7 in Section 4.2.1. In Section 4.2.2, we show the almost optimal *worst-case* coding theorem and optimal (average-case) coding theorem for next-bits-predictable distributions. In Section 4.2.3, we show the uniform variant of the result of [HMS23] (stated in the nonuniform model or the uniform model with shared randomness) at the expense of arbitrarily small multiplicative error.

4.2.1 Proofs of Conditional Coding Theorems

First, we observe that, under the non-existence of one-way functions, every samplable distribution becomes next-bits predictable on a subset of large weight.

Lemma 4.9. *If there is no almost everywhere one-way function, then for every samplable distribution $\mathcal{D} = \{\mathcal{D}_n\}$ and every polynomials $q(n)$ and $s(n)$, there exists a subset $S \subseteq \mathsf{Support}(\mathcal{D})$ such that (i) $\mathcal{D}$ is next-bits-predictable on S with error parameter q and (ii) for infinitely many $n \in \mathbb{N}$, $\mathbf{Pr}_{x \sim \mathcal{D}_n}[x \in S_n] \geq 1 - 1/s(n)$.*

Proof. The lemma follows from Lemma 3.6 for almost everywhere one-way functions and the standard empirical estimation of the probability that the algorithm Ext outputs each bit. $\qquad\square$

We also obtain the lemma for the infinitely-often security in the same way.

Lemma 4.10. *If there is no infinitely-often one-way function, then for every samplable distribution $\mathcal{D} = \{\mathcal{D}_n\}$ and every polynomials $q(n)$ and $s(n)$, there exists a subset $S \subseteq \mathsf{Support}(\mathcal{D})$ such that (i) $\mathcal{D}$ is next-bits-predictable on S with error parameter q and (ii) for all $n \in \mathbb{N}$, $\mathbf{Pr}_{x\sim\mathcal{D}_n}[x \in S_n] \geq 1 - 1/s(n)$.*

Proof. The proof is the same as Lemma 4.9 except we use Lemma 3.6 for infinitely-often one-way functions. $\qquad\square$

Now, we prove (Item 1 $\implies$ Item 4) in Theorem 1.1, which is restated as follows.

Theorem 4.11. *If there is no one-way function, then for every samplable distribution $\mathcal{D} = \{\mathcal{D}_n\}$ supported over $\{0,1\}^n \times \{0,1\}^n$ and every polynomial q, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(n)}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n)\right] \geq 1 - \frac{1}{q(n)}.$$

Moreover, there exists an efficient algorithm Enc that outputs, for given $(x,y) \sim \mathcal{D}_n$, a description of a $p(n)$-time program Π of length at most $-\log \mathcal{D}_n(x \mid y) + \log p(n)$ with probability at least $1 - 1/q(n)$ over the choice of $(x,y) \sim \mathcal{D}_n$ and randomness for Enc, such that Π outputs x for given y and randomness $r \sim \{0,1\}^{p(n)}$ with probability at least $2/3$ over the choice of r.

Proof. Let $\mathcal{D} = \{\mathcal{D}_n\}$ be an arbitrary samplable distribution, where each $\mathcal{D}_n$ is supported over $\{0,1\}^n \times \{0,1\}^n$. Let $\bar{\mathcal{D}} = \{\bar{\mathcal{D}}_n\}$ be another samplable distribution defined as $\bar{\mathcal{D}}_n \equiv \mathcal{D}_n^{(2)} \circ \mathcal{D}_n^{(1)}$. Namely, $\bar{\mathcal{D}}_n$ is distributed over $\{0,1\}^{2n}$.

Let q be an arbitrary polynomial. We apply Theorem 4.6 for $\bar{\mathcal{D}}$ and a polynomial $4q(n)n$. Then, there exists a polynomial p such that if $\bar{\mathcal{D}}$ is next-bits-predictable on $S \subseteq \mathsf{Support}(\bar{\mathcal{D}})$ with error parameter p, then for every $n \in \mathbb{N}$, every $y \circ x \in S_n$ with $|y| = |x| = n$,

$$\mathsf{rK}^{p(n)}(x \mid y) \leq -\log \bar{\mathcal{D}}_n^*(x \mid y) + (\mathfrak{m}_{\bar{\mathcal{D}}_n,1/4q(n)n}^{n,2n}(y \circ x) + 1) \cdot O(\log nq(n))$$

$$\leq -\log \mathcal{D}_n(x \mid y) + (\mathfrak{m}_{\bar{\mathcal{D}}_n,1/4q(n)n}(y \circ x) + 1) \cdot O(\log nq(n)).$$

Suppose that there is no almost everywhere one-way function. By Lemma 4.9, there exists a subset $S \in \mathsf{Support}(\bar{\mathcal{D}})$ such that (i) $\bar{\mathcal{D}}$ is next-bits-predictable on S with error parameter p and (ii) for infinitely many $n \in \mathbb{N}$, $\mathbf{Pr}_{y\circ x\sim\bar{\mathcal{D}}_n}[y \circ x \in S_n] \geq 1 - 1/(2q(n))$. Below, we fix such an n arbitrarily.

By Proposition 4.5, $\mathbf{Pr}_{y\circ x\sim\bar{\mathcal{D}}_n}[\mathfrak{m}_{1/4q(n)n}(y \circ x) = 0] \geq 1 - 1/(2q(n))$. Thus, by the union bound,

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathfrak{m}_{\bar{\mathcal{D}}_n,1/4q(n)n}(y \circ x) = 0 \text{ and } y \circ x \in S_n\right] \geq 1 - \frac{1}{q(n)}.$$

For any (x,y) satisfying the event above, we obtain that

$$\mathsf{rK}^{p(n)}(x \mid y) \leq -\log \mathcal{D}_n(x \mid y) + (\mathfrak{m}_{\bar{\mathcal{D}}_n,1/4q(n)n}(y \circ x) + 1) \cdot O(\log nq(n))$$

$$\leq -\log \mathcal{D}_n(x \mid y) + O(\log nq(n)).$$

By selecting a large enough polynomial p', we conclude that for infinitely many $n \in \mathbb{N}$,

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p'(n)}(x \mid y) \leq -\log\mathcal{D}_n(x \mid y) + \log p'(n)\right] \geq 1 - \frac{1}{q(n)},$$

where the existence of the efficient encoder follows from that of Theorem 4.6. $\square$

Next, we show (Item 1 $\implies$ Item 2) in Theorem 1.2 in almost the same way.

Theorem 4.12. *If there is no infinitely-often one-way function, then for every samplable distribution $\mathcal{D} = \{\mathcal{D}_n\}$ supported over $\{0,1\}^n \times \{0,1\}^n$, there exists a polynomial p such that for all $n, k \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(n,k)}(x \mid y) \leq \log\frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n,k)\right] \geq 1 - \frac{1}{k}.$$

Proof. Let $\mathcal{D} = \{\mathcal{D}_n\}$ be an arbitrary samplable distribution. We define another samplable distribution $\bar{\mathcal{D}} = \{\bar{\mathcal{D}}_{\langle n,k\rangle}\}_{n,k\in\mathbb{N}}$ as $\bar{\mathcal{D}}_{\langle n,k\rangle} \equiv \mathcal{D}_n^{(2)} \circ \mathcal{D}_n^{(1)}$ for each $n, k \in \mathbb{N}$, where $\langle,\rangle$ is the (standard) efficiently computable and efficiently invertible one-to-one pairing function satisfying that $\max\{n,k\} \leq \langle n,k\rangle \leq \mathsf{poly}(n,k)$.

We apply Theorem 4.6 for $\bar{\mathcal{D}}$ and a polynomial $4n \cdot n$. We use the same argument as the proof of Theorem 4.11 except we use Lemma 4.10 instead of Lemma 4.9. Then, we can show that there exists a polynomial p such that for *all* $n, k \in \mathbb{N}$ (because of Lemma 4.10),

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(\langle n,k\rangle)}(x \mid y) \leq -\log\mathcal{D}_n(x \mid y) + \log p(\langle n,k\rangle)\right] \geq 1 - \frac{1}{\langle n,k\rangle} \geq 1 - \frac{1}{k},$$

which implies Theorem 4.12 because $\langle n,k\rangle \leq \mathsf{poly}(n,k)$. Again, the existence of the efficient encoder follows from that of Theorem 4.6. $\square$

Proof of Theorem 1.7. (Item 2 $\implies$ Item 3) trivially holds. (Item 3 $\implies$ Item 1) has been proved in [TVZ05, Proposition 3.2] based on Levin's observation. Thus, it suffices to show (Item 1 $\implies$ Item 2).

Theorem 4.12 (in the formulation using (Enc, Dec) of Lemma 4.8) shows that, under the non-existence of infinitely-often one-way functions, for every samplable distribution $\mathcal{D} = \{\mathcal{D}_n\}$ (without loss of generality, we assume that each $\mathcal{D}_n$ is over $\{0,1\}^{\ell(n)}$ for an efficiently computable function $\ell(n) \leq \mathsf{poly}(n)$ by padding), there exists a pair of polynomial-time computable functions Enc and Dec such that for every $n \in \mathbb{N}$,

$$\Pr_{x\sim\mathcal{D}_n,\mathrm{Dec},\mathrm{Enc}}[\mathrm{Dec}(\mathrm{Enc}(x,1^n)) = x \text{ and } |\mathrm{Enc}(x,1^n)| \leq -\log\mathcal{D}(x) + O(\log n)] \geq 1 - \frac{1}{4\ell(n)^2}.$$

By Markov's inequality, we have

$$\Pr_{x}\left[\Pr_{\mathrm{Enc},\mathrm{Dec}}[\mathrm{Dec}(\mathrm{Enc}(x,1^n)) = x \text{ and } |\mathrm{Enc}(x,1^n)| \leq -\log\mathcal{D}(x) + O(\log n)] \geq 1 - \frac{1}{2\ell(n)}\right]$$

$$\geq 1 - \frac{1}{2\ell(n)}. \tag{4}$$

We consider a modified efficient encoder Enc' that, for a given $x \sim \mathcal{D}_n$ and 1^n, performs the empirical estimation of the probability that $\mathrm{Dec}(\mathrm{Enc}(x,1^n)) = x$ and $|\mathrm{Enc}(x,1^n)| \leq -\log\mathcal{D}(x) + O(\log n)$ hold with additive accuracy error $1/(8\ell(n))$ and with failure probability at most 2^{-n}. If the estimated probability $\tilde{p}$ is at least $1 - \frac{3}{4\ell(n)}$, the encoder Enc' sends $0\mathrm{Enc}(x,1^n)$ (executed with fresh random

seeds); otherwise, Enc' sends $1x$. We also define a modified efficient decoder Dec' that outputs for given encoding e', if e' takes the form of $0e$, it outputs $\text{Dec}(e)$; otherwise if e' takes the form of $1x$, it outputs x.

We verify the *worst-case* correctness of $(\text{Enc}', \text{Dec}')$. For every $n \in \mathbb{N}$ and every $x \in \mathsf{Support}(\mathcal{D}_n)$, if x passes the empirical test in Enc' under the condition that the empirical estimation is performed successfully, the probability that $\text{Dec}(\text{Enc}(x, 1^n)) = x$ holds is at least $1 - 3/(4\ell(n)) - 1/(8\ell(n)) = 1 - 7/(8\ell(n))$. Thus, the failure probability that $\text{Dec}'(\text{Enc}'(x, 1^m)) \neq x$ given this condition is at most $7/(8\ell(n))$. By contrast, if x does not pass the test, $\text{Dec}'(\text{Enc}'(x, 1^m)) = x$ holds with probability 1 given this event. Thus, for every $x \in \mathsf{Support}(\mathcal{D}_n)$, it holds that

$$\Pr_{\text{Enc}', \text{Dec}'}[\text{Dec}'(\text{Enc}'(x, 1^n)) = x] \geq 1 - \frac{7}{8\ell(n)} - 2^{-n} \geq \frac{2}{3}.$$

We also evaluate the expected length of the encoding. For every $x \in \mathsf{Support}(\mathcal{D}_n)$, if x satisfies the event in Equation (4), the probability that x passes the empirical test is at least $1 - 2^{-n}$ (in this case, Enc' outputs $\text{Enc}(x, 1^n)$). Thus, the expected length of the encoding under this condition that x satisfies the event in Equation (4) is at most

$$2^{-n} \cdot (\ell(n) + 1) + 1 \cdot |\text{Enc}(x, 1^n)| \leq -\log \mathcal{D}(x) + O(\log n).$$

By contrast, if x does not satisfy the event in Equation (4), the length of the outcome of $\text{Enc}'(x, 1^n)$ is always at most $\ell(n) + 1$. Thus, we conclude that

$$\mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[|\text{Enc}'(x, 1^n)|] \leq \mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[-\log \mathcal{D}_n(x)] + O(\log n) + \frac{\ell(n) + 1}{2\ell(n)} = \mathrm{H}(\mathcal{D}_n) + O(\log n).$$

This completes the proof of Theorem 1.7. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

4.2.2 Coding Theorems for Next-Bits-Predictable Source

We show that the next-bits prediction on the whole support yields the *optimal* (average-case) coding theorem.

Theorem 4.13. *If a distribution family* $\mathcal{D} = \{\mathcal{D}_n\}$*, where each* $\mathcal{D}_n$ *is over* $\{0, 1\}^n$*, is next-bits-predictable on* $\mathsf{Support}(\mathcal{D})$ *with arbitrary polynomial error parameter, then for every polynomial* q*, there exists a polynomial* p *such that for every* $n \in \mathbb{N}$*,*

$$\Pr_{x \sim \mathcal{D}_n}\left[\mathsf{rK}^{p(n)}(x) \leq -\log \mathcal{D}_n(x) + \log p(n)\right] \geq 1 - \frac{1}{q(n)}.$$

In particular, there exists a pair (Enc, Dec) *of randomized polynomial-time algorithms such that for every* $n \in \mathbb{N}$*,*

$$\mathop{\mathbf{E}}_{x \sim \mathcal{D}_n, \text{Enc}}[|\text{Enc}(x, 1^n)|] \leq \mathrm{H}(\mathcal{D}_n) + \log p(n)$$

and for every $x \in \mathsf{Support}(\mathcal{D}_n)$*,*

$$\Pr_{\text{Enc}, \text{Dec}}[\text{Dec}(\text{Enc}(x, 1^n)) = x] \geq \frac{2}{3}.$$

Proof. We apply Theorem 4.6 for $\mathcal{D}$ and a polynomial $nq(n)$. Then, there exists a polynomial p such that for every $n \in \mathbb{N}$ and every $x \in \mathsf{Support}(\mathcal{D}_n)$,

$$\mathsf{rK}^{p(n)}(x) \leq -\log \mathcal{D}_n(x) + \mathfrak{m}_{1/(nq(n))}(x) \cdot O(\log n) + \log p(n).$$

By Proposition 4.5, it holds that for every $n \in \mathbb{N}$,

$$\Pr_{x \sim \mathcal{D}_n}\left[\mathfrak{m}_{1/(nq(n))}(x) > 0\right] \leq \frac{n}{nq(n)} = \frac{1}{q(n)}.$$

From the two expressions above, we obtain

$$\Pr_{x \sim \mathcal{D}_n}\left[\mathsf{rK}^{p(n)}(x) \leq -\log \mathcal{D}_n(x) + \log p(n)\right] \geq 1 - \frac{1}{q(n)}.$$

In particular, Lemma 4.8 shows that there exists a pair $(\mathrm{Enc}, \mathrm{Dec})$ of randomized polynomial-time algorithms such that for every $n \in \mathbb{N}$,

$$\Pr_{x \sim \mathcal{D}_n, \mathrm{Enc}}[|\mathrm{Enc}(x, 1^n)| \leq -\log \mathcal{D}_n(x) + \log p(n)] \geq 1 - \frac{1}{2n}.$$

and for every $x \in \mathsf{Support}(\mathcal{D}_n)$,

$$\Pr_{\mathrm{Enc}, \mathrm{Dec}}[\mathrm{Dec}(\mathrm{Enc}(x, 1^n)) = x] \geq \frac{2}{3}.$$

Without loss of generality, we assume that Enc always outputs the encoding of length at most $2n$ for given $x \in \{0,1\}^n$ and 1^n; otherwise, we can replace the encoding to the canonical one into which embedded x. Thus, the expected length of the encoding is bounded as follows:

$$\mathop{\mathbf{E}}_{x \sim \mathcal{D}_n, \mathrm{Enc}}[|\mathrm{Enc}(x, 1^n)|] \leq \mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[-\log \mathcal{D}_n(x)] + O(\log n) + \frac{2n}{2n} = \mathrm{H}(\mathcal{D}_n) + O(\log n),$$

as desired. $\qquad\square$

Furthermore, the same class of next-bits-predictable distributions admits the almost optimal worst-case coding theorem in the following sense.

Theorem 4.14. *If a distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, where each $\mathcal{D}_n$ is over $\{0,1\}^n$, is next-bits-predictable on $\mathsf{Support}(\mathcal{D})$ with arbitrary polynomial error parameter, then for every $\epsilon > 0$, there exists a polynomial p such that for every $n \in \mathbb{N}$ and every $x \in \mathsf{Support}(\mathcal{D}_n)$,*

$$\mathsf{rK}^{p(n)}(x) \leq -(1 + \epsilon) \log \mathcal{D}_n(x) + \log p(n).$$

Note that the term "almost" optimal is due to the arbitrarily small constant $\epsilon > 0$ above.

Proof. Let $k \in \mathbb{N}$ be a sufficiently large constant (with respect to ϵ^{-1}) specified later. We apply Theorem 4.6 for $\mathcal{D}$ and a polynomial $q(n) = n^k$. Then, there exists a polynomial p such that for every $n \in \mathbb{N}$ and every $x \in \mathsf{Support}(\mathcal{D}_n)$,

$$\mathsf{rK}^{p(n)}(x) \leq -\log \mathcal{D}_n(x) + C \cdot \mathfrak{m}_{n^{-k}}(x) \log n + O(\log n),$$

where $C > 0$ is a universal constant independent of k.

For every $x \in \mathsf{Support}(\mathcal{D}_n)$, we can observe that

$$\mathcal{D}_n(x) = \prod_{i=1}^{n} \mathcal{D}^*(x_i \mid x_{[i-1]}) \leq (n^{-k})^{\mathfrak{m}_{n^{-k}}(x)} = n^{-k\mathfrak{m}_{n^{-k}}(x)}.$$

By rearranging the above,

$$\mathfrak{m}_{n^{-k}}(x) \log n \leq -\frac{1}{k} \log \mathcal{D}_n(x).$$

Therefore, by selecting k to be sufficiently large so that $C/k \leq \epsilon$, we have

$$\begin{aligned}
\mathsf{rK}^{p(n)}(x) &\leq -\log \mathcal{D}_n(x) + C \cdot \mathfrak{m}_{n^{-k}}(x) \log n + O(\log n) \\
&\leq -\left(1 + \frac{C}{k}\right) \log \mathcal{D}_n(x) + O(\log n) \\
&\leq -(1 + \epsilon) \log \mathcal{D}_n(x) + O(\log n),
\end{aligned}$$

as desired. □

4.2.3 Towards the Uniform Version of the Haitner–Mazor–Silbak Theorem

Recently, Haitner, Mazor, and Silbak [HMS23] presented the clear relationship between incompressibility and next-bit pseudoentropy in the *nonuniform* computational model. They further mentioned that the result holds in the uniform computational model when *shared randomness is available* between the encoder and decoder [see HMS23, Remark 6]. Note that the shared randomness is used for executing a distinguisher, and polynomially many shared random bits are required in general. We extend the relationship to the *uniform* computational model without the usage of the shared randomness only at the expense of *a small multiplicative loss*.

First, we review the main result of [HMS23]. For this, we recall the notions of incompressibility and next-bit pseudoentropy.

Definition 4.15 (Imcompressibility). For $k \colon \mathbb{N} \to \mathbb{N}$, a distribution family $\mathcal{D} = \{\mathcal{D}_n\}$ is said to be nonuniformly k-incompressible if for every nonuniform polynoimal-time algorithms Enc and Dec such that $\mathrm{Dec}(\mathrm{Enc}(x, 1^n)) = x$ for all $x \in \mathsf{Support}(\mathcal{D}_n)$, it holds that for every large enough $n \in \mathbb{N}$,

$$\mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[|\mathrm{Enc}(x, 1^n)|] \geq k(n).$$

Definition 4.16 (Pseudoentropy). Let $(X, B) = \{(X_n, B_n)\}_n$ be a family of joint distributions over strings. We say that B has nonuniform-conditional-pseudoentropy (resp. uniform-conditional-pseudoentropy) $k \colon \mathbb{N} \to \mathbb{N}$ given X if for every polynomial p, there exists a distribution family $C = \{C_n\}$ that jointly distributed with $\{X_n\}$ as satisfies the following:

- $\mathrm{H}(C_n \mid x_n) \geq k(n) - 1/p(n)$ for each $n \in \mathbb{N}$;

- (X, B) and (X, C) are computationally indistinguishable by nonuniform (resp. uniform) randomized polynomial-time algorithms.

Definition 4.17 (Next-bit pseudoentropy). A distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, where each $\mathcal{D}_n$ is over $\{0, 1\}^{\ell(n)}$, is said to have nonuniform-next-bit-pseudoentropy (resp. uniform-next-bit-pseudoentropy) $k \colon \mathbb{N} \to \mathbb{N}$ if a distribution family $\{(\mathcal{D}_n)_{I_n}\}_n$, where each I_n is the uniform distribution over $[\ell(n)]$ and $(\mathcal{D}_n)_{I_n}$ represents the I_n-th bit of $\mathcal{D}_n$, has nonuniform-conditional-pseudoentropy (resp. uniform-conditional-pseudoentropy) $k(n)/\ell(n)$ given $\{(\mathcal{D}_n)_{[I_n-1]}\}_n$.

One of the main theorems of [HMS23] is stated as follows:

Theorem 4.18 ([HMS23, Lemma 1]). *For a distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, if $\mathcal{D}$ is nonuniformly $k(n)$-incompressible, then $\mathcal{D}$ has nonuniform-next-bit-pseudoentropy $k(n) - 2$.*

In this section, we show the uniform variant with a multiplicative loss in the parameter $k(n)$. First, we present the notion of randomized compression in the uniform computational model, which was formally studied in [TVZ05].

Definition 4.19 (Randomized compression). We say that a pair $(\mathrm{Enc}, \mathrm{Dec})$ of randomized algorithms compresses a distribution family $\mathcal{D} = \{\mathcal{D}_n\}_{n \in \mathbb{N}}$ to length $k \colon \mathbb{N} \to \mathbb{N}$ with decoding error $\delta \colon \mathbb{N} \to [0, 1]$ if it satisfies that for infinitely many $n \in \mathbb{N}$,[5]

- for any $x \in \mathsf{Support}(\mathcal{D}_n)$, $\mathbf{Pr}_{\mathrm{Enc}, \mathrm{Dec}}[\mathrm{Dec}(\mathrm{Enc}(x, 1^n)) = x] \geq 1 - \delta(n)$;

- $\mathbf{E}_{x \sim \mathcal{D}_n, \mathrm{Enc}}[|\mathrm{Enc}(x)|] \leq k(n)$.

We say that $\mathcal{D}$ is randomly compressible to length m with decoding error δ if there exists a pair of randomized polynomial-time algorithms that compresses $\mathcal{D}$ to length m with decoding error δ. Moreover, we say that $\mathcal{D}$ is randomly incompressible to length m with decoding error δ if not randomly compressible to length m with decoding error δ.

It is known that the decoding error can be exponentially reduced.

Lemma 4.20 ([TVZ05, Lemma 2.11]). *For a distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, where each $\mathcal{D}_n$ is over $\{0, 1\}^{\ell(n)}$, if $\mathcal{D}$ is randomly compressible to length m with decoding error δ, then it is also randomly compressible to the length $m + 3\delta \cdot \ell(n) + 2$ with decoding error 2^{-n}.*

Now, we state the main result in this section.

Theorem 4.21. *For a distribution family $\mathcal{D} = \{\mathcal{D}_n\}$ and every constant $\epsilon > 0$, if $\mathcal{D}$ is randomly $k(n)$-incompressible with decoding error 2^{-n}, then $\mathcal{D}$ has uniform-next-bit-pseudoentropy $(1 - \epsilon)k(n) - O(\log n)$.*

A large part of the proof follows from that of [HMS23], so we strongly recommend the reader to refer to the prior work first. Below, we extract the relevant points to our work.

We introduce some notions following from [VZ12; HMS23]. For a function $p \colon \{0, 1\}^* \times \{0, 1\} \to \mathbb{R}_{>0}$, we define a conditional probability $C_p(\cdot | \cdot)$ as for each $b \in \{0, 1\}$ and $x \in \{0, 1\}^*$,

$$C_p(b \mid x) = \frac{p(x, b)}{p(x, 0) + p(x, 1)}.$$

For a randomized algorithm P that maps $(x, b) \in \{0, 1\}^* \times \{0, 1\}$ to a real positive value, we extend the notion above as

$$C_P(b \mid x) = \mathbf{E}_r \left[\frac{P(x, b; r)}{P(x, 0; r) + P(x, 1; r)} \right].$$

Notice that

$$C_P(0 \mid x) + C_P(1 \mid x) = \mathbf{E}_r \left[\frac{P(x, 0; r) + P(x, 1; r)}{P(x, 0; r) + P(x, 1; r)} \right] = 1.$$

Furthermore, for each $m \in \mathbb{N}$, we define a distribution $\mathcal{D}_m^P$ over $\{0, 1\}^m$ as for each $x \in \{0, 1\}^m$,

$$\mathcal{D}_m^P(x) = \prod_{i=1}^{m} C_P(x_i \mid x_{[i-1]}).$$

The prior work [HMS23] showed the following technical lemma.

---

[5]In this work, we consider a randomized compression only on infinitely many $n \in \mathbb{N}$ to discuss the uniform version of *almost everywhere* imcompressibility

Lemma 4.22 ([HMS23, Section 3.3] building upon [VZ12]). *If a distribution family $\mathcal{D} = \{\mathcal{D}_n\}$, where $\mathcal{D}_n$ is over $\{0,1\}^{\ell(n)}$, does not have uniform-next-bit pseudoentropy $k(n)$, then there exists a randomized polynomial-time algorithm P that maps $(x, b) \in \{0,1\}^* \times \{0,1\}$ to a real positive value so that for infinitely many $n \in \mathbb{N}$,*

$$\mathrm{KL}(\mathcal{D}_n \| \mathcal{D}_{\ell(n)}^P) \leq k(n) - \mathrm{H}(\mathcal{D}_n).$$

We also use the following well-known fact.

Lemma 4.23 ([cf. CT06, Theorem 5.4.3]). *For every distributions $\mathcal{D}$ and $\mathcal{E}$ with $\mathrm{KL}(\mathcal{D}\|\mathcal{E}) < \infty$,*

$$\mathop{\mathbf{E}}_{x \sim \mathcal{D}}[-\log \mathcal{E}(x)] = \mathrm{H}(\mathcal{D}) + \mathrm{KL}(\mathcal{D}\|\mathcal{E}).$$

Now, we prove Theorem 4.21 based on Theorem 4.14 and Lemmas 4.22 and 4.23.

Proof of Theorem 4.21. Let $k(n)$ be an arbitrary polynomial. Let $\mathcal{D} = \{\mathcal{D}_n\}$ be a distribution family, where each $\mathcal{D}_n$ is over $\{0,1\}^{\ell(n)}$. By Lemma 4.22, if $\mathcal{D}$ does not have uniform-next-bit pseudoentropy $k(n)$, then there exists a randomized polynomial-time algorithm P that maps $(x, b) \in \{0,1\}^* \times \{0,1\}$ to a real positive value so that for infinitely many $n \in \mathbb{N}$,

$$\mathrm{KL}(\mathcal{D}_n \| \mathcal{D}_{\ell(n)}^P) \leq k(n) - \mathrm{H}(\mathcal{D}_n). \tag{5}$$

We observe that for each $n \in \mathbb{N}$, $x \in \mathsf{Support}(\mathcal{D}_{\ell(n))}^P)$, $b \in \{0,1\}$, and $i \in [\ell(n)]$, the conditional probability $\mathcal{D}_{\ell(n))}^{P*}(b \mid x_{[i-1]}) = C_P(b \mid x_{[i-1]})$ is predictable with additive error $1/p(n)$, where p is an arbitrarily large polynomial, by the empirical estimation of the quantity

$$\mathop{\mathbf{E}}_{r}\left[\frac{P(x, b; r)}{P(x, 0; r) + P(x, 1; r)}\right].$$

Note that the approximation halts in polynomial time in n and $p(n)$.

Therefore, by Theorem 4.14 (in the form of Lemma 4.8), there exists a pair (Enc, Dec) of randomized polynomial-time algorithms such that for every $n \in \mathbb{N}$ and every $x \in \mathsf{Support}(\mathcal{D}_{\ell(n)}^P)$, it holds that

$$\mathop{\mathbf{Pr}}_{\mathrm{Enc,Dec}}[\mathrm{Dec}(\mathrm{Enc}(x, 1^n)) = x] \geq 1 - \frac{1}{3\ell(n)} \tag{6}$$

and

$$|\mathrm{Enc}(x, 1^n)| \leq -(1 + \epsilon) \log \mathcal{D}_{\ell(n)}^P(x) + O(\log n).$$

Thus, we have that for every n satisfying Equation (5),

$$\mathop{\mathbf{E}}_{\mathrm{Enc}, x \sim \mathcal{D}_n}[|\mathrm{Enc}(x, 1^n))|] \leq (1 + \epsilon) \mathop{\mathbf{E}}_{x \sim \mathcal{D}_n}[-\log \mathcal{D}_{\ell(n)}^P(x)] + O(\log n)$$

$$\leq (1 + \epsilon)\left(\mathrm{H}(\mathcal{D}_n) + \mathrm{KL}(\mathcal{D}_n \| \mathcal{D}_{\ell(n)}^P)\right) + O(\log n)$$

$$\leq (1 + \epsilon) \cdot k(n) + O(\log n), \tag{7}$$

where the second inequality follows from Lemma 4.23, and the last inequality follows from Equation (5).

Thus, from Equations (6) and (7), $\mathcal{D}$ is randomly compressible to length $(1+\epsilon) \cdot k(n) + O(\log n)$ with decoding error $1/(3\ell(n))$. By Lemma 4.20, $\mathcal{D}$ is also randomly compressible with decoding error 2^{-n} to the length

$$(1 + \epsilon) \cdot k(n) + O(\log n) + \frac{3\ell(n)}{3\ell(n)} + 2 = (1 + \epsilon) \cdot k(n) + O(\log n).$$

By retaking $k(n)$ to be $(1/1+\epsilon)(k(n) - O(\log n)) = (1 - \epsilon/(1+\epsilon))(k(n) - O(\log n))$, we obtain the theorem. □

28

5 One-Way Functions, Conditional Coding and Symmetry of Information for rK^poly

In this section, we prove Theorem 1.1 and Theorem 1.2. We first show Theorem 1.1, which is restated below.

Theorem 1.1. *The following are equivalent.*

1. *One-way functions do not exist.*

2. **(Infinitely-Often Average-Case Symmetry of Information for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$, the following holds for all $t \geq p(n)$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^t(x \mid y) \leq \mathsf{rK}^t(x,y) - \mathsf{rK}^t(y) + \log t\right] \geq 1 - \frac{1}{q(n)}.$$

3. **(Infinitely-Often Average-Case Conditional Coding for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(n)}(x \mid y) \leq \log\frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n)\right] \geq 1 - \frac{1}{q(n)},$$

where $\mathcal{D}_n(x \mid y)$ denotes the probability that (x,y) is sampled from $\mathcal{D}_n$ conditioned that the second item being sampled is y.

4. **(Infinitely-Often Average-Case Efficient Conditional Coding for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p(n)}(x \mid y) \leq \log\frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n)\right] \geq 1 - \frac{1}{q(n)}.$$

Moreover, it admits an efficient encoder in the following sense: there exists an efficient algorithm Enc that outputs, for given $(x,y) \sim \mathcal{D}_n$, a description of a $p(n)$-time program Π of length at most $-\log \mathcal{D}_n(x \mid y) + \log p(n)$ with probability at least $1 - 1/q(n)$ over the choice of $(x,y) \sim \mathcal{D}_n$ and randomness for Enc, such that Π outputs x for given y and randomness $r \sim \{0,1\}^{p(n)}$ with probability at least $2/3$ over the choice of r.

Proof. (Item 1 $\implies$ Item 3) follows directly from Theorem 4.11.

We then show the following implications in subsequent sections.

- Item 3 $\implies$ Item 2 (Lemma 5.1 in Section 5.1).

- Item 2 $\implies$ Item 1 (Lemma 5.4 in Section 5.2).

This will complete the proof of Theorem 1.1. $\qquad\square$

5.1 Average-Case Symmetry of Information from Conditional Coding

Lemma 5.1 (Item 3 $\implies$ Item 2 in Theorem 1.1). *If average-case conditional coding holds for* rK^t, *then average-case symmetry of information also holds.*

We first need the following lemma.

Lemma 5.2. *If one-way functions do not exist, then for every samplable distribution family* $\mathcal{D} = \{\mathcal{D}_n\}$ *supported over* $\{0,1\}^n \times \{0,1\}^n$ *and every polynomial* q, *there exists a polynomial* p *such that for infinitely many* $n \in \mathbb{N}$,

$$\Pr_{(x,y) \sim \mathcal{D}_n} \left[\mathsf{rK}^{p(n)}(x \mid y) \leq \mathsf{K}(x \mid y) + \log p(n) \right] \geq 1 - \frac{1}{q(n)}.$$

Proof. Let $\mathcal{D} = \{\mathcal{D}_n\}$ be a polynomial-time samplable distribution family and q be a polynomial.

Assuming one-way functions do not exist, by Theorem 4.11, we have that there exists some p' such that for infinitely many $n \in \mathbb{N}$,

$$\Pr_{(x,y) \sim \mathcal{D}_n} \left[\mathsf{rK}^{p'(n)}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + \log p'(n) \right] \geq 1 - \frac{1}{2q(n)}.$$

Also, by Lemma 3.2, we get that for every n, with probability at least $1 - (2q(n))$ over $(x,y) \sim \mathcal{D}_n$, we have

$$\mathsf{K}(x \mid y) \geq \log \frac{1}{\mathcal{D}_n(x \mid y)} - O(\log q(n)).$$

By a union bound, we get that for infinitely many n, with probability at least $1 - q(n)$ over $(x,y) \sim \mathcal{D}_n$,

$$\mathsf{rK}^{p'(n)}(x \mid y) \leq \mathsf{K}(x \mid y) + \log p'(n) + O(\log q(n)).$$

The lemma follows by letting p be a sufficient large polynomial. $\square$

We are now ready to show Lemma 5.1.

Proof of Lemma 5.1. Let $\mathcal{D} = \{\mathcal{D}_n\}$ be a polynomial-time samplable distribution family and q be a polynomial.

We show the following claim.

Claim 5.3. *There exists a polynomial* p' *such that for infinitely many* n, *both the following hold with probability at least* $1 - 1/q(n)$ *over* $(x,y) \sim \mathcal{D}_n$.

1. $\mathsf{rK}^{p'(n)}(x \mid y) \leq \log \mathsf{K}(x \mid y) + \log p'(n)$.

2. $\mathsf{rK}^{p'(n)}(y) \leq \mathsf{K}(y) + \log p'(n) + O(\log n)$.

Proof of Claim 5.3. Let $\mathcal{D}' := \{\mathcal{D}'_n\}$ be the polynomial-time samplable distribution family where each $\mathcal{D}'_n$ is sampled by first sampling $y \sim \mathcal{D}_n^{(2)}$ and outputs $(y, 0^n)$, where $\mathcal{D}_n^{(2)}$ is the marginal distribution of $\mathcal{D}_n$ on the second half.

Finally, let $\mathcal{E}$ be the uniform mixture of $\mathcal{D}$ and $\mathcal{D}'$.

Suppose one-way functions do not exist. Then by Lemma 5.2, there exists a polynomial p' such that for infinitely many $n \in \mathbb{N}$,

$$\Pr_{(x,y) \sim \mathcal{E}_n} \left[\mathsf{rK}^{p'(n)}(x \mid y) > \mathsf{K}(x \mid y) + \log p'(n) \right] \leq \frac{1}{4q(n)}.$$

Since $\mathcal{E}_n$ samples $\mathcal{D}_n$ with probability $1/2$,

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^{p'(n)}(x\mid y) > \mathsf{K}(x\mid y) + \log p'(n)\right] \leq \frac{1}{2q(n)}. \tag{8}$$

Similarly, we get

$$\Pr_{(a,b)\sim\mathcal{D}'_n}\left[\mathsf{rK}^{p'(n)}(a\mid b) > \mathsf{K}(a\mid b) + \log p'(n)\right] \leq \frac{1}{2q(n)}. \tag{9}$$

Note that the Equation (9) essentially means

$$\Pr_{y\sim\mathcal{D}_n^{(2)}}\left[\mathsf{rK}^{p'(n)}(y\mid 0^n) > \mathsf{K}(y\mid 0^n) + \log p'(n)\right] \leq \frac{1}{2q(n)}.$$

Finally, note that $\mathsf{rK}^{p'(n)}(y) \leq \mathsf{rK}^{p'(n)}(y\mid 0^n) + O(\log n)$ and $\mathsf{K}(y) \geq \mathsf{K}(y\mid 0^n)$. Therefore, the above implies

$$\Pr_{y\sim\mathcal{D}_n^{(2)}}\left[\mathsf{rK}^{p'(n)}(y) > \mathsf{K}(y) + \log p'(n) + O(\log n)\right] \leq \frac{1}{2q(n)}. \tag{10}$$

The claim follows by taking a union bound over Equations (8) and (10). ◇

Fix any n and (x,y) such that both the conditions stated in Claim 5.3 hold. Then we have

$$\begin{aligned}
\mathsf{rK}^{p'(n)}(x\mid y) &\leq \mathsf{K}(x\mid y) + \log p'(n,k) && \text{(by Item 1 of Claim 5.3)}\\
&\leq \mathsf{K}(x,y) - \mathsf{K}(y) + O(\log n) + \log p'(n) && \text{(by Symmetry of Information for } \mathsf{K}\text{)}\\
&\leq \mathsf{K}^t(x,y) - \mathsf{rK}^{p'(n)}(y) + O(\log n) + 2\log p'(n). && \text{(by Item 2 of Claim 5.3)}
\end{aligned}$$

By letting p be a sufficiently large polynomial, we get that for every $t \geq p(n)$,

$$\mathsf{rK}^t(x\mid y) \leq \mathsf{rK}^t(x,y) - \mathsf{rK}^t(y) + \log t,$$

as desired. □

5.2 Inverting One-Way Functions from Average-Case Symmetry of Information

Lemma 5.4 (Item 2 $\implies$ Item 1 in Theorem 1.1). *If average-case symmetry of information holds for rK^t, then one-way functions do not exist.*

Proof. The proof uses ideas from [LW95].

Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be any function that is supposed to be infinitely-often secure. Let q be any polynomial, we show that, for infinitely many n, we can invert f with high probability $1 - 1/q(n)$ over $x \sim \{0,1\}^n$ in time $\mathsf{poly}(n)$.

We first observe the following.

Claim 5.5 ([LW95, Lemma 3.5]). *For every n and $x \in \{0,1\}^n$, we have*

$$\mathsf{K}(f(x)) \geq \mathsf{K}(x) - \log|f^{-1}(f(x))| - O(\log n).$$

Proof of Claim 5.5. First of all, for every $x \in \{0,1\}^n$, we have

$$\mathsf{K}(x \mid f(x)) \leq \log |f^{-1}(f(x))| + O(\log n). \tag{11}$$

To see this, note that given $f(x)$, we can specify x using the index of x in the set $f^{-1}(f(x))$, which takes $\leq \log |f^{-1}(f(x))|$ bits. Then we have

$$\begin{aligned}
\mathsf{K}(x) &\leq \mathsf{K}(x \mid f(x)) + \mathsf{K}(f(x)) \\
&\leq \log |f^{-1}(f(x))| + \mathsf{K}(f(x)) + O(\log n), \qquad \text{(by Equation (11))}
\end{aligned}$$

as desired. ◇

Next, assuming that average-case symmetry of information hods for $\mathsf{rK}^{\mathsf{poly}}$ (Item 3 in Theorem 1.1), we show the following.

Claim 5.6. *There is a polynomial p such that for infinitely many n, with probability at least $1 - 1/q(n)^2$ over $x \sim \{0,1\}^n$, we have*

$$\mathsf{rK}^{p(n)}(x \mid f(x)) \leq \log |f^{-1}(f(x))| + \log p(n).$$

Proof of Claim 5.6. Consider the polynomial-time samplable distribution family $\{\mathcal{D}_n\}$ where each $\mathcal{D}_n$ samples $x \sim \{0,1\}^n$ and outputs $(x, f(x))$.

By the assumption that average-case symmetry of information hods for $\mathsf{rK}^{\mathsf{poly}}$ (Item 3 in Theorem 1.1), there exists a polynomial p' such that for infinitely many n, with probability at least $1 - 1/(2q(n)^2)$ over $x \sim \{0,1\}^n$, we have

$$\begin{aligned}
\mathsf{rK}^{p'(n)}(x \mid f(x)) &\leq \mathsf{rK}^{p'(n)}(x, f(x)) - \mathsf{rK}^{p'(n)}(f(x)) + \log p'(n) \\
&\leq \mathsf{rK}^{p'(n)/2}(x) - \mathsf{rK}^{p'(n)}(f(x)) + \log p'(n) + O(\log n) \\
&\leq \mathsf{rK}^{p'(n)/2}(x) - \mathsf{K}(f(x)) + \log p'(n) + O(\log n) \\
&\leq \mathsf{rK}^{p'(n)/2}(x) - \left(\mathsf{K}(x) - \log |f^{-1}(f(x))| - O(\log n)\right) + \log p'(n) + O(\log n) \\
&\leq \mathsf{rK}^{p'(n)/2}(x) - \mathsf{K}(x) + \log |f^{-1}(f(x))| + \log p'(n) + O(\log n), \tag{12}
\end{aligned}$$

where the second inequality uses the fact that given x we can compute $f(x)$ efficiently, and the second last inequality follows from Claim 5.5.

Note that by a counting argument, for every n, with probability at least $1 - 1/(2q(n)^2)$ over $x \sim \{0,1\}^n$, we have

$$\mathsf{K}(x) \geq n - O(\log q(n)),$$

which yields

$$\mathsf{rK}^{p'(n)/2}(x) - \mathsf{K}(x) \leq O(\log q(n)). \tag{13}$$

By Plugging Equation (13) into Equation (12) and by a union bound, we get that, for infinitely many n, with probability at least $1 - 1/q(n)^2$ over $x \sim \{0,1\}^n$

$$\mathsf{rK}^{p'(n)}(x \mid f(x)) \leq \log |f^{-1}(f(x))| + \log p'(n) + O(\log q(n)).$$

The claim follows by letting p be a sufficiently large polynomial. ◇

In what follows, we fix n so that the condition stated in Claim 5.6 holds.

Next, we observe the following equivalent way of sampling $(x, f(x))$ while x is uniformly at random: We first sample $y := f(z)$ for a uniformly random z and then sample $x \sim f^{-1}(y)$. By an averaging argument, Claim 5.6 yields that with probability at least $1 - 1/q(n)$ over y sampled this way, for at least $1 - 1/q(n)$ fraction of the $x \in f^{-1}(y)$, we have

$$\mathsf{rK}^{p(n)}(x \mid y) \le \log |f^{-1}(y)| + \log p(n). \tag{14}$$

Consider any *good* y such that Equation (14) holds for at least $1 - 1/q(n)$ fraction of the $x \in f^{-1}(y)$. Also, let S_y be the set of $x \in f^{-1}(y)$ such that Equation (14) holds. Note that

$$|S_y| \ge (1 - 1/q(n)) \cdot |f^{-1}(y)|.$$

Consider the following procedure A that takes n and y as input and does the following.

1. Pick $s \sim [2n]$,

2. Pick $\Pi \sim \{0,1\}^s$,

3. View Π as a *randomized* program, run $U(\Pi, y)$ for $p(n)$ steps, and return its output.

It is easy to see from the definition of rK^t that for *every* $x \in S_y$ (which satisfies Equation (14)), the above procedure A outputs x with probability at least

$$\frac{1}{O(n)} \cdot \frac{1}{2^{\log |f^{-1}(y)| + \log p(n)}} \cdot \frac{2}{3} \ge \frac{1}{|f^{-1}(y)|} \cdot \frac{1}{p(n)^2}.$$

Since the above holds for every $x \in S_y$, we get that the probability that $\mathsf{A}(1^n, y)$ outputs *some* $x \in S_y$ is at least

$$|S_y| \cdot \frac{1}{|f^{-1}(y)|} \cdot \frac{1}{p(n)^2} \ge \frac{1}{\mathsf{poly}(n)}.$$

In other words, with probability at least $1 - 1/k$ over $x \sim \{0,1\}^n$ (in which case $f(x)$ is good), $\mathsf{A}(1^n, f(x))$ outputs some pre-image of $f(x)$ with probability at least $1/\mathsf{poly}(n)$. This breaks the one-way-ness of f. $\qquad\square$

5.3 Characterizing Infinitely-Often One-Way Functions

In this subsection, we show Theorem 1.2, which is restated below.

Theorem 1.2. *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*

2. **(Almost-Everywhere Average-Case Conditional Coding for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n \in \mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, there exists a polynomial p such that for all $n, k \in \mathbb{N}$,*

$$\Pr_{(x,y) \sim \mathcal{D}_n}\left[\mathsf{rK}^{p(n,k)}(x \mid y) \le \log \frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n,k)\right] \ge 1 - \frac{1}{k}.$$

33

3. **(Almost-Everywhere Worst-Case Conditional Coding for rK^t with Computational Depth)** *There exists a constant $c > 0$ such that the following holds. For every computable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, all $n, t \in \mathbb{N}$ such that $t \geq n$ and all $(x, y) \in \mathsf{Support}(\mathcal{D}_n)$,*

$$\mathsf{rK}^{(2^\alpha \cdot t)^c}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + c \cdot (\log t + \alpha),$$

where $\alpha := \mathsf{cd}^t(x, y)$.

4. **(Almost-Everywhere Average-Case Symmetry of Information for rK^t)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$, there exists a polynomial p such that for all $n, k \in \mathbb{N}$ and $t \geq p(n, k)$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{rK}^t(x \mid y) \leq \mathsf{rK}^t(x, y) - \mathsf{rK}^t(y) + \log t\right] \geq 1 - \frac{1}{k}.$$

5. **(Almost-Everywhere Worst-Case Symmetry of Information for rK^t with Computational Depth)** *There exists a constant $c > 0$ such that the following holds. For all $n, t \in \mathbb{N}$ such that $t \geq n$ and all $x, y \in \{0,1\}^n$,*

$$\mathsf{rK}^{(2^\alpha \cdot t)^c}(x \mid y) \leq \mathsf{rK}^t(x, y) - \mathsf{rK}^t(y) + c \cdot (\log t + \alpha),$$

where $\alpha := \mathsf{cd}^t(x, y)$.

Proof. (Item 1 $\implies$ Item 2) follows directly from Theorem 4.12. The proof of (Item 2 $\implies$ Item 4) can be easily adapted from that of Lemma 5.1. Also, the proof of (Item 4 $\implies$ Item 1) can be easily adapted from that of Lemma 5.4. This shows the equivalence of Item 1, Item 2, and Item 4.

We then show the following implications in the rest of this subsection.

- Item 2 $\iff$ Item 3 (Lemma 5.7 and Lemma 5.8).

- Item 4 $\iff$ Item 5 (Lemma 5.9 and Lemma 5.10).

This will complete the proof of Theorem 1.2. □

Lemma 5.7. *We have (Item 2 $\implies$ Item 3) in Theorem 1.2.*

Proof. Fix $n, t \in \mathbb{N}$ such that $t \geq n$. Let $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ be any computable distribution family. Also, Let α be any integer and let $c > 0$ be a constant to be specified later. It suffices to show that for any $(x, y) \in \{0,1\}^n \times \{0,1\}^n$, if $\mathsf{cd}^t(x, y) \leq \alpha$, then

$$\mathsf{rK}^{(2^\alpha \cdot t)^c}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + c \cdot (\log t + \alpha). \tag{15}$$

We will show the contrapositive. That is, if Equation (15) is false, then $\mathsf{cd}^t(x, y) > \alpha$.

Let $d > 0$ be a sufficiently large constant, and let

$$k := 2^\alpha \cdot t^d.$$

We defined the following polynomial-time samplable distribution family $\{Q_{\langle n,t\rangle}\}_{n,t\in\mathbb{N}}$, where each $Q_{\langle n,t\rangle}$ does the following.

1. Pick $s \sim [2n]$.

2. Pick $r \sim \{0,1\}^t$.

3. Pick $\Pi \sim \{0,1\}^s$.

4. Run $U(\Pi, r)$ for t steps. If we obtain a pair $(x,y) \in \{0,1\}^n \times \{0,1\}^n$, output (x,y). Otherwise output $(0^n, 0^n)$[6].

It is easy to see from the definition of pK^t that for every $(x,y) \in \{0,1\}^n \times \{0,1\}^n$,

$$Q_{\langle n,t \rangle}(x,y) \geq \frac{1}{O(n)} \cdot \frac{2}{3} \cdot \frac{1}{2^{\mathsf{pK}^t(x,y)}}. \tag{16}$$

By applying Item 2 of Theorem 1.2 to $\{Q_{\langle n,t \rangle}\}$ and by letting d be sufficiently large, we have

$$\Pr_{(x,y) \sim Q_{\langle n,t \rangle}} \left[\mathsf{rK}^{(tk)^d}(x \mid y) \leq \log \frac{1}{Q_{\langle n,t \rangle}(x \mid y)} + d \cdot \log(tk) \right] \geq 1 - \frac{1}{2k}. \tag{17}$$

Also, by Lemma 3.2, we have

$$\Pr_{(x,y) \sim Q_{\langle n,t \rangle}} \left[\mathsf{K}(x \mid y) > \log \frac{1}{Q_{\langle n,t \rangle}(x \mid y)} - \log k - O(\log n) \right] \geq 1 - \frac{1}{2k}. \tag{18}$$

Moreover, by the coding theorem for (time-unbounded) Kolmogorov complexity (Theorem 3.1), we have that for every $(x,y) \in \mathsf{Support}(\mathcal{D}_n)$

$$\mathsf{K}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + O(\log n). \tag{19}$$

By combining Equations (17), (18) and (19), we get that

$$\Pr_{(x,y) \sim Q_{\langle n,t \rangle}} \left[\mathsf{rK}^{(tk)^d}(x \mid y) \leq \frac{1}{\mathcal{D}_n(x \mid y)} + 2d \cdot \log(tk) \right] \geq 1 - \frac{1}{k}.$$

Now, consider the set E of (x,y) such that

$$\mathsf{rK}^{(tk)^d}(x \mid y) \leq \frac{1}{\mathcal{D}_n(x \mid y)} + 2d \cdot \log(tk).$$

Note that by letting $c > d$ be a sufficiently large constant, for any (x,y) such that Equation (15) is false, we get that $(x,y) \in E$. Therefore, it suffices to show that for every $(x,y) \in E$, we have $\mathsf{cd}^t(x,y) > \alpha$.

First of all, we have

$$\sum_{(x,y) \in E} Q_{\langle n,t \rangle}(x,y) \leq \frac{1}{k},$$

which implies

$$\sum_{(x,y) \in E} Q_{\langle n,t \rangle}(x,y) \cdot k \leq 1.$$

---

[6]Here, we let $Q_{\langle n,t \rangle}$ output pairs of strings in $\{0,1\}^n \times \{0,1\}^n$. By using padding, we can also ensure that $Q_{\langle n,t \rangle}$ outputs pairs of strings in $\{0,1\}^{\langle n,t \rangle} \times \{0,1\}^{\langle n,t \rangle}$. This will not affect the correctness of the argument. We omit this technicality for simplicity of presentation.

We can then define a distribution $\mathcal{E}$ whose support is E and $\mathcal{E}(x,y) = Q_{\langle n,t \rangle}(x,y) \cdot k$. Note that $\mathcal{E}$ is computable since E is decidable.

Applying the coding theorem (Theorem 3.1) on $\mathcal{E}$, we get that for every $(x,y) \in E$,

$$\mathsf{K}(x,y) \leq \log \frac{1}{\mathcal{E}(x,y)} + O(\log t)$$

$$= \log \frac{1}{Q_{\langle n,t \rangle}(x,y) \cdot k} + O(\log t). \tag{20}$$

Finally, we get that for every $(x,y) \in E$,

$$\mathsf{pK}^t(x,y) - \mathsf{K}(x,y) \geq \log \frac{1}{Q_{\langle n,t \rangle}(x,y)} - \mathsf{K}(x,y) - O(\log n) \qquad \text{(by Equation (16))}$$

$$\geq \log k - O(\log t) \qquad \text{(by Equation (20))}$$

$$> \alpha,$$

as desired. $\qquad \square$

Lemma 5.8. *We have* (*Item 3 $\implies$ Item 2*) *in Theorem 1.2.*

Proof. Fix $n \in \mathbb{N}$ and any polynomial-time samplable distribution samplable distribution family $\{\mathcal{D}_n\}$.

By Lemma 3.5, there exists a polynomial ρ such that

$$\Pr_{(x,y) \sim \mathcal{D}_n} \left[\mathsf{cd}^{\rho(n)}(x,y) \leq O(\log nk) \right] \geq 1 - \frac{1}{k}.$$

Also, by the assumption that Item 3 in Theorem 1.2 is true, there exists a constant $c > 0$ such that for $t := \rho(n)$ and all $(x,y) \in \mathsf{Support}(\mathcal{D}_n)$

$$\mathsf{rK}^{(2^{\mathsf{cd}^t(x,y)} \cdot t)^c}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + c \cdot (\log t + \mathsf{cd}^t(x,y)).$$

It follows that with probability at least $1 - 1/k$ over $(x,y) \sim \mathcal{D}_n$,

$$\mathsf{rK}^{(nk\rho(n))^{O(c)}}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + c \cdot (\log \rho(n) + O(\log nk)),$$

which implies

$$\mathsf{rK}^{p(n,k)}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n,k),$$

where p is a polynomial. $\qquad \square$

Lemma 5.9. *We have* (*Item 4 $\implies$ Item 5*) *in Theorem 1.2.*

Proof Sketch. The proof can be easily adapted from that of Lemma 5.7. The idea is to apply average-case symmetry of information (Item 4 in in Theorem 1.2) to the distribution family $\{Q_{\langle n,t \rangle}\}_{n,t \in \mathbb{N}}$ as defined in the proof of Lemma 5.7. This allows us to say that the set of pairs of strings where symmetry of information fails must have large computational depth. We omit the details here. $\qquad \square$

Lemma 5.10. *We have* (*Item 5 $\implies$ Item 4*) *in Theorem 1.2.*

Proof Sketch. The proof is essentially the same as that of Lemma 5.8, using the fact that a string drawn from any polynomial-time samplable distribution has small computational depth (Lemma 3.5). We omit the details here. $\qquad \square$

6   One-Way Functions and Average-Case Hardness of rK$^{\mathsf{poly}}$

In this section, we prove Theorem 1.3, which is restated below.

Theorem 1.3. *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*

2. **(Search-MINrKT is easy on average over polynomial-time samplable distributions)**
   For every $\lambda \in [0,1]$, every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, where each $\mathcal{D}_n$ is over $\{0,1\}^n$, there exist a polynomial ρ and a probabilistic polynomial-time algorithm A such that for all $n,s,\ell,k \in \mathbb{N}$, and all $t \geq \rho(n)$,

$$\Pr_{x\sim\mathcal{D}_n,A}\Big[A(x,1^t,1^\ell,1^k) \text{ outputs an } (1/\ell)\text{-rK}^t_\lambda\text{-witness of } x\Big] \geq 1 - \frac{1}{k}.$$

3. **(MINrKT is easy on average over polynomial-time samplable distributions)** *For every $\lambda \in [0,1)$, every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, where each $\mathcal{D}_n$ is over $\{0,1\}^n$, there exist a polynomial ρ and a probabilistic polynomial-time algorithm A such that for all $n,s,\ell,k \in \mathbb{N}$, and all $t \geq \rho(n)$,*

$$\Pr_{x\sim\mathcal{D}_n,A}\Big[A(-,1^k) \text{ decides } \lambda\text{-MINrKT } on \ input \ (x,1^s,1^t,1^\ell)\Big] \geq 1 - \frac{1}{k}.$$

4. **(MINrKT is easy on average over the uniform distribution)** *There exist a polynomial ρ and a probabilistic polynomial-time algorithm A such that for all $n,s,\ell,k \in \mathbb{N}$,*

$$\Pr_{x\sim\{0,1\}^n,A}\Big[A(-,1^k) \text{ decides } (2/3)\text{-MINrKT } on \ input \ (x,1^s,1^{\rho(n)},1^\ell)\Big] \geq 1 - \frac{1}{k}.$$

6.1   Technical Lemmas

Lemma 6.1 ([LP20])**.** *If MINrKT is easy on average over the uniform distribution (i.e., Item 4 in Theorem 1.3 holds), then infinitely-often one-way functions do not exist.*

The following is a key lemma for proving Theorem 1.3.

Lemma 6.2. *If infinitely-often one-way functions do not exist, then for every $\lambda \in [0,1)$, there exists a probabilistic polynomial-time algorithm A such that for all $n,t,\ell,k \in \mathbb{N}$ with $t \geq n^{1.01}$, with probability at least $1 - 1/k$ over the internal randomness of A,*

$$\sum_{x\in\{0,1\}^n} 2^{-\mathsf{rK}^t_\lambda(x)} \cdot \mathbf{1}_{\big[A(x,1^t,1^\ell,1^k;r_A) \notin \lambda\text{-Search-MINrKT}(x,1^t,1^\ell)\big]} \leq \frac{\mathsf{poly}(n)}{k}. \tag{21}$$

To show Lemma 6.2, we will need the following simple proposition.

Proposition 6.3. *For every $\lambda \in [0,1)$, there is a probabilistic polynomial-time algorithm V such that for all $n,t,\ell,k \in \mathbb{N}$, the following holds with probability at least $1 - 2^k$ over the interval randomness of V. For every randomized program $\Pi \in \{0,1\}^{\leq 2n}$ and $x \in \{0,1\}^n$,*

1. *if within t steps, Π outputs x with probability at least λ, then $V(x,\Pi,1^t,1^\ell,1^k)$ accepts, and*

2. *if within t steps, Π outputs x with probability less than $\lambda - 1/\ell$, then $V(x,\Pi,1^t,1^\ell,1^k)$ rejects.*

Proof Sketch. Given a randomized program $\Pi \in \{0,1\}^{\leq 2n}$ and $x \in \{0,1\}^n$, we repeatedly runs the randomized program Π for t steps, for $\mathsf{poly}(n,k,\ell)$ times and counts the fraction μ of times that x is obtained. Using standard concentration bounds, it is easy to show that the following holds with probability at least $1 - 2^{-n^2} \cdot 2^{-k}$. If the condition stated in the first item of the proposition is true, then $\mu \geq \lambda - 1/(2\ell)$, and if the condition stated in the first item, then $\mu < \lambda - 1/(2\ell)$. Finally, we apply a union bound over all $\Pi \in \{0,1\}^{\leq 2n}$ and $x \in \{0,1\}^n$. $\qquad\square$

We now show Lemma 6.2.

Proof of Lemma 6.2. We will show the lemma for $\lambda := 2/3$. It is not hard to adapt the proof to any $\lambda \in [0,1)$.

Let $c > 0$ be a constant so that $\mathsf{rK}^t(x) \leq n + c$ for every $x \in \{0,1\}^n$ and $t \geq n^{1.01}$. Let V be the algorithm in Proposition 6.3 instantiated with λ.

Let f be a polynomial-time computable function defined as follows.

> On input $(r_V, i, \Pi, r_t, r_\ell, r_k)$, where $r_V \in \{0,1\}^{\mathsf{poly}(t,\ell,k)}$, $i \in [n+c]$, $\Pi \in \{0,1\}^{n+c}$, $r_t \in \{0,1\}^t$, $r_\ell \in \{0,1\}^\ell$ $r_k \in \{0,1\}^k$ and $r_2 \in \{0,1\}^k$. We run $U(\Pi_{[i]}, r_t)$ for t steps and obtain a string x. If $|x| = n$ and $V(x, \Pi_{[i]}, 1^t, 1^\ell, 1^k; r_V) = 1$, we output $(r_V, i, x, 1^t, 1^\ell, 1^k)$; otherwise we output $\perp$.

Since we assume infinitely-often one-way functions do not exist (which implies infinitely-often weak one-way functions do not exist), there is a polynomial-time algorithm Invert such that for all $n, t, \ell, k \in \mathbb{N}$, it holds that

$$\mathbf{Pr}\Big[\mathsf{Invert}(r_V, i, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}}) \text{ succeeds}\Big] \geq 1 - \frac{1}{2k^2},$$

where $(r_V, i, x, 1^t, 1^\ell, 1^k)$ is sampled according to f, $r_{\mathsf{Invert}} \in \{0,1\}^{\mathsf{poly}(t,\ell,k)}$ is the internal randomness of Invert, and "$\mathsf{Invert}(r_V, i, x, 1^t, 1^\ell, 1^k)$ succeeds" means $\mathsf{Invert}(i, x, 1^t, 1^\ell, 1^k)$ returns a pre-image of $(r_V; i, x, 1^t, 1^\ell, 1^k)$.

By an averaging argument, we get that with probability at least $1 - 1/(2k)$ over r_V (the internal randomness of V used in the definition of f) and r_{Invert} (the internal randomness of Invert), it holds that

$$\mathbf{Pr}\Big[\mathsf{Invert}(r_V, i, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}}) \text{ succeeds}\Big] \geq 1 - \frac{1}{k}, \tag{22}$$

where the above probability is only over i and x. Also, with probability at least $1 - 2^{-k}$ over a uniformly random r_V, the condition stated in Proposition 6.3 will hold. By a union bound, with probability at least $1 - 1/k$, both Equation (24) and the condition stated in Proposition 6.3 hold for a uniform random $(r_V, r_{\mathsf{Invert}})$. Let us consider any such *good* pair $(r_V, r_{\mathsf{Invert}})$.

By a union bound, Equation (22) yields that for all $i \in [n+c]$,

$$\mathbf{Pr}\Big[\mathsf{Invert}(r_V, i, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}}) \text{ succeeds}\Big] \geq 1 - \frac{n+c}{k}, \tag{23}$$

where now the probability is only over x.

Next, for every $i \in [n+c]$, let $\mathcal{D}_i$ be the following distribution (note that we have also fixed a good r_V):

1. Pick $\Pi \sim \{0,1\}^{n+c}$.

2. Pick $r_t \sim \{0,1\}^t$.

3. Run $U(\Pi_{[i]}, r_t)$ for t steps and obtain a string x. If $|x| = n$ and $V(x, \Pi_{[i]}, 1^t, 1^\ell, 1^k; r_V) = 1$, we output x; otherwise output $\perp$.

Then Equation (23) implies that for all $i \in [n + c]$,

$$\Pr_{x \sim \mathcal{D}_i}\left[\mathsf{Invert}(r_V, i, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}}) \text{ fails}\right] \leq \frac{n + c}{k}. \tag{24}$$

Now consider the following algorithm A for solving Search-MINrKT:

On input $(x, 1^t, 1^\ell, 1^k)$, we pick $r_V \sim \{0, 1\}^{\mathsf{poly}(t, \ell, k)}$ and $r_{\mathsf{Invert}} \sim \{0, 1\}^{\mathsf{poly}(t, \ell, k)}$. We then try to find the smallest $i \in [n + c]$ such that the following holds: $\mathsf{Invert}(r_V, i, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}})$ returns some $(r_V, i, \Pi, r_t, r_\ell, r_k)$ such that after running $U(\Pi_{[i]}, r_t)$ for t steps, x is obtained, and that $V(x, \Pi_{[i]}, 1^t, 1^\ell, 1^k; r_V) = 1$. In this case, we output $\Pi_{[i]}$.

We claim that the algorithm A satisfies the condition stated in the lemma.

First note that in the description of the above algorithm, with probability at least $1 - 1/k$, the internal randomness of A, $r_A := (r_V, r_{\mathsf{Invert}})$, will be good, and hence satisfy Equation (24) and the condition stated in Proposition 6.3. In what follows, we fix such a good r_A.

For the sake of contradiction, suppose

$$\sum_{x \in \{0,1\}^n} 2^{-\mathsf{rK}^t(x)} \cdot 1_{\left[A(x, 1^t, 1^\ell, 1^k; r_A) \notin \text{Search-MINrKT}(x, 1^t, 1^\ell)\right]} \leq \frac{n^b}{k}, \tag{25}$$

where $b > 0$ is a constant to be specified later.

Note that for every $i \in [n + c]$ and every $x \in \{0, 1\}^n$ such that $\mathsf{rK}^t(x) = i$, we have

$$\mathcal{D}_i(x) \geq 2^{-\mathsf{rK}^t(x)} \cdot \frac{2}{3}. \tag{26}$$

This is because in the sampling procedure of $\mathcal{D}_i$, with probability $2^{-\mathsf{rK}^t(x)}$, we will pick a Π whose i-bit prefix corresponds to a rK^t-witness of x. In this case, with probability at least $2/3$ over r_t, we obtain x. Finally, note that by the property of V and the fact that r_V is good, in this case we have $V(x, \Pi_{[i]}, 1^t, 1^\ell, 1^k; r_V) = 1$ and hence x will be output.

Also, for every $i \in [n + c]$ and every $x \in \{0, 1\}^n$ such that $\mathsf{rK}^t(x) = i$, if there exists an $i^* \leq i$ such that $\mathsf{Invert}(r_V, i^*, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}})$ succeeds, then it means we obtain some $(r_V, i^*, \Pi, r_t, r_\ell, r_k)$ such that after running $U(\Pi_{[i^*]}, r_t)$ for t steps, x is obtained and that $V(x, \Pi_{[i^*]}, 1^t, 1^\ell, 1^k; r_V) = 1$. Again, by the property of V and the fact that r_V is good, $\Pi_{[i^*]}$ is an $(1/\ell)$-rK^t-witness of x, which means $A(x, 1^t, 1^\ell, 1^k; r_V) \in \text{Search-MINrKT}(x, 1^t, 1^\ell)$.

In other words, for every $i \in [n+c]$ and every $x \in \{0, 1\}^n$ such that $\mathsf{rK}^t(x) = i$, if $A(x, 1^t, 1^\ell, 1^k; r_A) \notin$ Search-MINrKT$(x, 1^t, 1^\ell)$, then for all $i^* \leq i$, $\mathsf{Invert}(r_V, i^*, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}})$ fails. In particular, the latter holds for $i^* = i$.

Then we have

$$\frac{n^b}{k} \leq \sum_{i \in [n+c]} \sum_{\substack{x \in \{0,1\}^n: \\ \mathsf{rK}^t(x) = i}} 2^{-\mathsf{rK}^t(x)} \cdot 1_{\left[A(x, 1^t, 1^\ell, 1^k; r_A) \notin \text{Search-MINrKT}(x, 1^t, 1^\ell)\right]} \qquad \text{(by Equation (25))}$$

$$\leq \sum_i \sum_{\substack{x \in \{0,1\}^n: \\ \mathsf{rK}^t(x) = i}} \frac{3}{2} \cdot \mathcal{D}_i(x) \cdot 1_{\left[A(x, 1^t, 1^\ell, 1^k; r_A) \notin \text{Search-MINrKT}(x, 1^t, 1^\ell)\right]} \qquad \text{(by Equation (26))}$$

$$\leq \frac{3}{2} \cdot \sum_i \sum_{\substack{x \in \{0,1\}^n: \\ \mathsf{rK}^t(x) = i}} \mathcal{D}_i(x) \cdot 1_{\left[\mathsf{Invert}(r_V, i^*, x, 1^t, 1^\ell, 1^k; r_{\mathsf{Invert}}) \text{ fails}\right]}.$$

By averaging, the above implies that there exists some i such that

$$\sum_{\substack{x \in \{0,1\}^n: \\ \mathsf{rK}^t(x)=i}} \mathcal{D}_i(x) \cdot 1_{\left[\mathsf{Invert}(r_V, i, x, y, 1^t, 1^k; r_{\mathsf{Invert}}) \text{ fails}\right]} \geq \frac{2 \cdot n^b}{3 \cdot (n+c) \cdot k},$$

which contradicts Equation (24) by letting b be a sufficiently large constant. □

6.2 Proof of Theorem 1.3

We now show Theorem 1.3.

Proof of Theorem 1.3. (Item 2 $\implies$ Item 3) and (Item 3 $\implies$ Item 4) are trivial. (Item 4 $\implies$ Item 1) follows from Lemma 6.1. Below, we show (Item 1 $\implies$ Item 2).

Let $\lambda \in [0, 1)$, and let $\{\mathcal{D}_n\}_n$ be a polynomial-time samplable distribution family. Also let ρ to be a polynomial to be specified later.

Fix $n, \ell, k \in \mathbb{N}$ and $t \geq \rho(n, k)$.

Let A be the polynomial-time algorithm in Lemma 6.2, and let $d > 0$ be some constant to be specified later. We have that with probability at least $1 - 1/(2k)$ over the internal randomness r_A of A,

$$\sum_{x \in \{0,1\}^n} 2^{-\mathsf{rK}_\lambda^t(x)} \cdot 1_{\left[A(x, 1^t, 1^\ell, 1^{(nk)^d}; r_A) \notin \mathsf{Search\text{-}MINrKT}(x, 1^t, 1^\ell)\right]} \leq \frac{1}{(nk)^d}. \tag{27}$$

Also, by Theorem 4.12, there exists a polynomial p such that,

$$\Pr_{x \sim \mathcal{D}_n}\left[\mathsf{rK}^{p(n,k)}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x)} + \log p(n, k)\right] \geq 1 - \frac{1}{4k}. \tag{28}$$

Fix any *good* r_A such that Equation (27) holds. We claim that

$$\Pr_{x \sim \mathcal{D}_n}\left[A(x, 1^t, 1^\ell, 1^{(nk)^d}; r_A) \in \mathsf{Search\text{-}MINrKT}(x, 1^t, 1^\ell)\right] \geq 1 - \frac{1}{2k}. \tag{29}$$

Note that this suffices to show the theorem, since r_A is good with probability at least $1 - 1/(2k)$.

Suppose, for the sake of contradiction, Equation (29) is not true. Then

$$\Pr_{x \sim \mathcal{D}_n}\left[A(x, 1^t, 1^\ell, 1^{(nk)^d}; r_A) \notin \mathsf{Search\text{-}MINrKT}(x, 1^t, 1^\ell)\right] > \frac{1}{2k}. \tag{30}$$

Let $\mathcal{E}(x)$ be the event that both the following hold.

- $A(x, 1^t, 1^\ell, 1^{(nk)^d}; r_A) \notin \mathsf{Search\text{-}MINrKT}(x, 1^t, 1^\ell)$

- $\mathsf{rK}^{p(n,k)}(x) \leq \log \frac{1}{\mathcal{D}_n(x)} + \log p(n, k)$.

By Equation (28) and Equation (30), we get that

$$\sum_{x \in \{0,1\}^n} \mathcal{D}_n(x) \cdot 1_{\mathcal{E}(x)} \geq \frac{1}{4k}. \tag{31}$$

Note that whenever $\mathcal{E}(x)$ holds, we have

$$\mathcal{D}_n(x) \leq \frac{p(n, k)}{2^{\mathsf{rK}^{p(n,k)}(x)}}. \tag{32}$$

40

Now we have

$$\frac{1}{4k} \leq \sum_{x \in \{0,1\}^n} \mathcal{D}_n(x) \cdot 1_{\mathcal{E}(x)} \qquad \text{(by Equation (31))}$$

$$\leq \sum_{x \in \{0,1\}^n} \frac{p(n,k)}{2^{\mathsf{rK}^{p(n,k)}(x)}} \cdot 1_{\mathcal{E}(x)} \qquad \text{(by Equation (32))}$$

$$\leq p(n,k) \cdot \sum_{x \in \{0,1\}^n} 2^{-\mathsf{K}^{p(n,k)}(x)} \cdot 1_{\mathcal{E}(x)}$$

$$\leq p(n,k) \cdot \sum_{x \in \{0,1\}^n} 2^{-\mathsf{rK}^{p(n,k)}(x)} \cdot 1_{\left[A(x,1^t,1^\ell,1^{(nk)^d};r_A) \notin \mathsf{Search\text{-}MINrKT}(x,1^t,1^\ell)\right]} \qquad (33)$$

Note that if $\lambda \leq 2/3$, we have $2^{-\mathsf{rK}^{p(n,k)}(x)} \leq 2^{-\mathsf{rK}_\lambda^t(x)}$ for all $t \geq p(n,k)$. On the other hand, if $\lambda > 2/3$, then by Lemma 3.4, we have $2^{-\mathsf{rK}^{p(n,k)}(x)} \leq 2^{-\mathsf{rK}_\lambda^t(x)+O(\log(1/(1-\lambda)))}$ for $t \geq p(n,k) \cdot O(\log(1/(1-\lambda)))$. In both cases, if $t \geq \rho(n,k)$ for some sufficiently large polynomial ρ, we have

$$\text{Equation (33)} \leq \left(\frac{1}{1-\lambda}\right)^{O(1)} \cdot p(n,k) \cdot \sum_{x \in \{0,1\}^n} 2^{-\mathsf{rK}_\lambda^t(x)} \cdot 1_{\left[A(x,1^t,1^\ell,1^{(nk)^d};r_A) \notin \mathsf{Search\text{-}MINrKT}(x,1^t,1^\ell)\right]}.$$

By rearranging, we get

$$\sum_{x \in \{0,1\}^n} 2^{-\mathsf{rK}_\lambda^t(x)} \cdot 1_{\left[A(x,1^t,1^\ell,1^{(nk)^d};r_A) \notin \mathsf{Search\text{-}MINrKT}(x,1^t,1^\ell)\right]} \geq \frac{(1-\lambda)^{O(1)}}{4k \cdot p(n,k)}.$$

However, this contradicts Equation (27) by letting d be a sufficiently large constant. $\square$

References

[ACMTV21] Eric Allender, Mahdi Cheraghchi, Dimitrios Myrisiotis, Harsha Tirumala, and Ilya Volkovich. "One-Way Functions and a Conditional Variant of MKTP". In: *Proceedings of the Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*. 2021, 7:1–7:19. DOI: `10.4230/LIPIcs.FSTTCS.2021.7`.

[AF09] Luis Filipe Coelho Antunes and Lance Fortnow. "Worst-Case Running Times for Average-Case Algorithms". In: *Proceedings of the Conference on Computational Complexity (CCC)*. 2009, pp. 298–303. DOI: `10.1109/CCC.2009.12`.

[AFMV06] Luis Antunes, Lance Fortnow, Dieter van Melkebeek, and N. V. Vinodchandran. "Computational depth: Concept and applications". In: *Theor. Comput. Sci.* 354.3 (2006), pp. 391–404. DOI: `10.1016/j.tcs.2005.11.033`.

[BSW03] Boaz Barak, Ronen Shaltiel, and Avi Wigderson. "Computational Analogues of Entropy". In: *Proceedings of the Randomization and Approximation Techniques in Computer Science (RANDOM/APPROX)*. 2003, pp. 200–215. DOI: `10.1007/978-3-540-45198-3_18`.

[BT06] Andrej Bogdanov and Luca Trevisan. "Average-Case Complexity". In: *Foundations and Trends in Theoretical Computer Science* 2.1 (2006). DOI: `10.1561/0400000004`.

[CIKK16]   Marco L. Carmosino, Russell Impagliazzo, Valentine Kabanets, and Antonina Kolokolova. "Learning Algorithms from Natural Proofs". In: *Proceedings of the Conference on Computational Complexity (CCC)*. 2016, 10:1–10:24. DOI: 10.4230/LIPIcs.CCC.2016.10.

[CT06]   Thomas M. Cover and Joy A. Thomas. *Elements of information theory (2. ed.)* Wiley, 2006. ISBN: 978-0-471-24195-9.

[GK22]   Halley Goldberg and Valentine Kabanets. "A Simpler Proof of the Worst-Case to Average-Case Reduction for Polynomial Hierarchy via Symmetry of Information". In: *Electronic Colloquium on Computational Complexity (ECCC)* 007 (2022).

[GKLO22]   Halley Goldberg, Valentine Kabanets, Zhenjian Lu, and Igor Carboni Oliveira. "Probabilistic Kolmogorov Complexity with Applications to Average-Case Complexity". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2022, 16:1–16:60. DOI: 10.4230/LIPIcs.CCC.2022.16.

[GM84]   Shafi Goldwasser and Silvio Micali. "Probabilistic Encryption". In: *J. Comput. Syst. Sci.* 28.2 (1984), pp. 270–299. DOI: 10.1016/0022-0000(84)90070-9.

[GS91]   Andrew V. Goldberg and Michael Sipser. "Compression and Ranking". In: *SIAM J. Comput.* 20.3 (1991), pp. 524–536. DOI: 10.1137/0220034.

[GUV09]   Venkatesan Guruswami, Christopher Umans, and Salil P. Vadhan. "Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes". In: *J. ACM* 56.4 (2009), 20:1–20:34. DOI: 10.1145/1538902.1538904.

[HILL99]   Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. "A Pseudorandom Generator from any One-way Function". In: *SIAM J. Comput.* 28.4 (1999), pp. 1364–1396. DOI: 10.1137/S0097539793244708.

[HILNO23]   Shuichi Hirahara, Rahul Ilango, Zhenjian Lu, Mikito Nanashima, and Igor C. Oliveira. "A Duality between One-Way Functions and Average-Case Symmetry of Information". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2023, pp. 1039–1050. DOI: 10.1145/3564246.3585138.

[Hir18]   Shuichi Hirahara. "Non-Black-Box Worst-Case to Average-Case Reductions within NP". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2018, pp. 247–258. DOI: 10.1109/FOCS.2018.00032.

[Hir20]   Shuichi Hirahara. "Characterizing Average-Case Complexity of PH by Worst-Case Meta-Complexity". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2020, pp. 50–60. DOI: 10.1109/FOCS46700.2020.00014.

[Hir21]   Shuichi Hirahara. "Average-case hardness of NP from exponential worst-case hardness assumptions". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2021, pp. 292–302. DOI: 10.1145/3406325.3451065.

[Hir22]   Shuichi Hirahara. "Symmetry of Information from Meta-Complexity". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2022, 26:1–26:41. DOI: 10.4230/LIPIcs.CCC.2022.26.

[Hir23]   Shuichi Hirahara. "Capturing One-Way Functions via NP-Hardness of Meta-Complexity". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2023, pp. 1027–1038. DOI: 10.1145/3564246.3585130.

[HKLO24]   Shuichi Hirahara, Valentine Kabanets, Zhenjian Lu, and Igor C. Oliveira. "Exact Search-To-Decision Reductions for Time-Bounded Kolmogorov Complexity". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2024, 29:1–29:56. DOI: `10.4230/LIPICS.CCC.2024.29`.

[HLR07]   Chun-Yuan Hsiao, Chi-Jen Lu, and Leonid Reyzin. "Conditional Computational Entropy, or Toward Separating Pseudoentropy from Compressibility". In: *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*. 2007, pp. 169–186. DOI: `10.1007/978-3-540-72540-4_10`.

[HMS23]   Iftach Haitner, Noam Mazor, and Jad Silbak. "Incompressiblity and Next-Block Pseudoentropy". In: *Proceedings of the Innovations in Theoretical Computer Science Conference (ITCS)*. 2023, 66:1–66:18. DOI: `10.4230/LIPIcs.ITCS.2023.66`.

[HN22]   Shuichi Hirahara and Mikito Nanashima. "Finding Errorless Pessiland in Error-Prone Heuristica". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2022, 25:1–25:28. DOI: `10.4230/LIPIcs.CCC.2022.25`.

[HN23]   Shuichi Hirahara and Mikito Nanashima. "Learning in Pessiland via Inductive Inference". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2023, pp. 447–457. DOI: `10.1109/FOCS57990.2023.00033`.

[HS22]   Shuichi Hirahara and Rahul Santhanam. "Errorless Versus Error-Prone Average-Case Complexity". In: *Proceedings of the Innovations in Theoretical Computer Science Conference (ITCS)*. 2022, 84:1–84:23. DOI: `10.4230/LIPIcs.ITCS.2022.84`.

[IL89]   Russell Impagliazzo and Michael Luby. "One-way Functions are Essential for Complexity Based Cryptography (Extended Abstract)". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1989, pp. 230–235. DOI: `10.1109/SFCS.1989.63483`.

[IL90]   Russell Impagliazzo and Leonid A. Levin. "No Better Ways to Generate Hard NP Instances than Picking Uniformly at Random". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1990, pp. 812–821. DOI: `10.1109/FSCS.1990.89604`.

[Ila20]   Rahul Ilango. "Connecting Perebor Conjectures: Towards a Search to Decision Reduction for Minimizing Formulas". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2020, 31:1–31:35. DOI: `10.4230/LIPIcs.CCC.2020.31`.

[Imp95]   Russell Impagliazzo. "A Personal View of Average-Case Complexity". In: *Proceedings of the Structure in Complexity Theory Conference*. 1995, pp. 134–147. DOI: `10.1109/SCT.1995.514853`.

[IRS22]   Rahul Ilango, Hanlin Ren, and Rahul Santhanam. "Robustness of average-case meta-complexity via pseudorandomness". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2022, pp. 1575–1583. DOI: `10.1145/3519935.3520051`.

[IZ89]   Russell Impagliazzo and David Zuckerman. "How to Recycle Random Bits". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1989, pp. 248–253. DOI: `10.1109/SFCS.1989.63486`.

[Lee06]   Troy Lee. "Kolmogorov Complexity and Formula Size Lower Bounds". PhD thesis. University of Amsterdam, 2006.

[Lev74] Leonid A. Levin. "Laws of information conservation (nongrowth) and aspects of the foundation of probability theory". In: *Problemy Peredachi Informatsii* 10.3 (1974), pp. 30–35.

[Lev86] Leonid A. Levin. "Average Case Complete Problems". In: *SIAM J. Comput.* 15.1 (1986), pp. 285–286. DOI: 10.1137/0215020.

[LM93]   Luc Longpré and Sarah Mocas. "Symmetry of Information and One-Way Functions". In: *Inf. Process. Lett.* 46.2 (1993), pp. 95–100. DOI: 10.1016/0020-0190(93)90204-M.

[LO21]   Zhenjian Lu and Igor Carboni Oliveira. "An Efficient Coding Theorem via Probabilistic Representations and Its Applications". In: *Proceedings of the International Colloquium on Automata, Languages, and Programming (ICALP)*. 2021, 94:1–94:20. DOI: 10.4230/LIPIcs.ICALP.2021.94.

[LOZ22] Zhenjian Lu, Igor Carboni Oliveira, and Marius Zimand. "Optimal Coding Theorems in Time-Bounded Kolmogorov Complexity". In: *Proceedings of the International Colloquium on Automata, Languages, and Programming (ICALP)*. 2022, 92:1–92:14. DOI: 10.4230/LIPIcs.ICALP.2022.92.

[LP20]   Yanyi Liu and Rafael Pass. "On One-way Functions and Kolmogorov Complexity". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2020, pp. 1243–1254. DOI: 10.1109/FOCS46700.2020.00118.

[LP21]   Yanyi Liu and Rafael Pass. "On the Possibility of Basing Cryptography on EXP≠BPP". In: *Proceedings of the International Cryptology Conference (CRYPTO)*. 2021, pp. 11–40. DOI: 10.1007/978-3-030-84242-0_2.

[LP22]   Yanyi Liu and Rafael Pass. "On One-Way Functions from NP-Complete Problems". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2022, 36:1–36:24. DOI: 10.4230/LIPIcs.CCC.2022.36.

[LP23a] Yanyi Liu and Rafael Pass. "On One-Way Functions and Sparse Languages". In: *Proceedings of the Theory of Cryptography Conference (TCC)*. 2023, pp. 219–237. DOI: 10.1007/978-3-031-48615-9_8.

[LP23b] Yanyi Liu and Rafael Pass. "One-Way Functions and the Hardness of (Probabilistic) Time-Bounded Kolmogorov Complexity w.r.t. Samplable Distributions". In: *Proceedings of the International Cryptology Conference (CRYPTO)*. 2023, pp. 645–673. DOI: 10.1007/978-3-031-38545-2_21.

[LR05]   Troy Lee and Andrei E. Romashchenko. "Resource bounded symmetry of information revisited". In: *Theor. Comput. Sci.* 345.2-3 (2005), pp. 386–405. DOI: 10.1016/j.tcs.2005.07.017.

[LW95]   Luc Longpré and Osamu Watanabe. "On Symmetry of Information and Polynomial Time Invertibility". In: *Inf. Comput.* 121.1 (1995), pp. 14–22. DOI: 10.1006/inco.1995.1120.

[MP24]   Noam Mazor and Rafael Pass. "Search-To-Decision Reductions for Kolmogorov Complexity". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2024, 34:1–34:20. DOI: 10.4230/LIPICS.CCC.2024.34.

[Nao91] Moni Naor. "Bit Commitment Using Pseudorandomness". In: *J. Cryptol.* 4.2 (1991), pp. 151–158. DOI: 10.1007/BF00196774.

[Oli19]   Igor Carboni Oliveira. "Randomness and Intractability in Kolmogorov Complexity". In: *Proceedings of the International Colloquium on Automata, Languages, and Programming (ICALP)*. 2019, 32:1–32:14. DOI: 10.4230/LIPIcs.ICALP.2019.32.

[Rom90]   John Rompel. "One-Way Functions are Necessary and Sufficient for Secure Signatures". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 1990, pp. 387–394. DOI: 10.1145/100216.100269.

[RRV02]   Ran Raz, Omer Reingold, and Salil P. Vadhan. "Extracting all the Randomness and Reducing the Error in Trevisan's Extractors". In: *J. Comput. Syst. Sci.* 65.1 (2002), pp. 97–128. DOI: 10.1006/jcss.2002.1824.

[RS21]   Hanlin Ren and Rahul Santhanam. "Hardness of KT Characterizes Parallel Cryptography". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2021, 35:1–35:58. DOI: 10.4230/LIPIcs.CCC.2021.35.

[SM10]   David Salomon and Giovanni Motta. *Handbook of Data Compression (5. ed.)* Springer, 2010. ISBN: 978-1-84882-902-2. DOI: 10.1007/978-1-84882-903-9.

[Tra84]   Boris A. Trakhtenbrot. "A Survey of Russian Approaches to Perebor (Brute-Force Searches) Algorithms". In: *IEEE Annals of the History of Computing* 6.4 (1984), pp. 384–400. DOI: 10.1109/MAHC.1984.10036.

[Tre01]   Luca Trevisan. "Extractors and pseudorandom generators". In: *J. ACM* 48.4 (2001), pp. 860–879. DOI: 10.1145/502090.502099.

[TUZ07]   Amnon Ta-Shma, Christopher Umans, and David Zuckerman. "Lossless Condensers, Unbalanced Expanders, And Extractors". In: *Combinatorica* 27.2 (2007), pp. 213–240. DOI: 10.1007/s00493-007-0053-2.

[TVZ05]   Luca Trevisan, Salil P. Vadhan, and David Zuckerman. "Compression of Samplable Sources". In: *Computational Complexity* 14.3 (2005), pp. 186–227. DOI: 10.1007/s00037-005-0198-6.

[Vad12]   Salil P. Vadhan. "Pseudorandomness". In: *Foundations and Trends in Theoretical Computer Science* 7.1-3 (2012), pp. 1–336. DOI: 10.1561/0400000010.

[VZ12]   Salil P. Vadhan and Colin Jia Zheng. "Characterizing pseudoentropy and simplifying pseudorandom generator constructions". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2012, pp. 817–836. DOI: 10.1145/2213977.2214051.

[Wee04]   Hoeteck Wee. "On Pseudoentropy versus Compressibility". In: *Proceedings of the Conference on Computational Complexity (CCC)*. 2004, pp. 29–41. DOI: 10.1109/CCC.2004.1313782.

[Yao82]   Andrew Chi-Chih Yao. "Theory and Applications of Trapdoor Functions (Extended Abstract)". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1982, pp. 80–91. DOI: 10.1109/SFCS.1982.45.

[ZL70]   Alexander K Zvonkin and Leonid A Levin. "The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms". In: *Russian Mathematical Surveys* 25.6 (1970), pp. 83–124.