Electronic Colloquium on Computational Complexity, Report No. 85 (2024)

Impagliazzo's Worlds Through the Lens of Conditional Kolmogorov Complexity

Zhenjian Lu* Rahul Santhanam†

April 25, 2024

Abstract

We develop new characterizations of Impagliazzo's worlds Algorithmica, Heuristica and Pessiland by the intractability of conditional Kolmogorov complexity K and conditional probabilistic time-bounded Kolmogorov complexity pK^t.

In our first set of results, we show that $\mathsf{NP} \subseteq \mathsf{BPP}$ iff $\mathsf{pK}^t(x \mid y)$ can be computed efficiently in the worst case when t is sublinear in $|x|+|y|$; $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ iff $\mathsf{pK}^t(x \mid y)$ can be computed efficiently over all polynomial-time samplable distributions when t is sublinear in $|x| + |y|$; and infinitely-often one-way functions fail to exist iff $\mathsf{pK}^t(x \mid y)$ can be computed efficiently over all polynomial-time samplable distributions for t a sufficiently large polynomial in $|x| + |y|$. These results characterize Impagliazzo's worlds Algorithmica, Heuristica and Pessiland purely in terms of the tractability of conditional pK^t. Notably, the results imply that Pessiland fails to exist iff the average-case intractability of conditional pK^t is insensitive to the difference between sub-linear and polynomially bounded t. As a corollary, while we prove conditional pK^t to be NP-hard for sublinear t, showing NP-hardness for large enough polynomially bounded t would eliminate Pessiland as a possible world of average-case complexity.

In our second set of results, we characterize Impagliazzo's worlds Algorithmica, Heuristica and Pessiland by the distributional tractability of a natural problem, i.e., approximating the conditional Kolmogorov complexity, that is provably intractable in the worst case. We show that $\mathsf{NP} \subseteq \mathsf{BPP}$ iff conditional Kolmogorov complexity can be approximated in the *semi-worst case*; and $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ iff conditional Kolmogorov complexity can be approximated on average over all *independent polynomial-time samplable distributions*. It follows from a result by Ilango, Ren, and Santhanam (STOC 2022) that infinitely-often one-way functions fail to exist iff conditional Kolmogorov complexity can be approximated on average over all *polynomial-time samplable distributions*. Together, these results yield the claimed characterizations. Our techniques, combined with previous work, also yield a characterization of auxiliary-input one-way functions and equivalences between different average-case tractability assumptions for conditional Kolmogorov complexity and its variants. Our results suggest that novel average-case tractability assumptions such as tractability in the semi-worst case and over independent polynomial-time samplable distributions might be worthy of further study.

*University of Warwick, UK. E-mail: zhenjian.lu@warwick.ac.uk

†University of Oxford, UK. E-mail: rahul.santhanam@cs.ox.ac.uk

ISSN 1433-8092

Contents

1 Introduction

In his influential survey on average-case complexity [Imp95], Impagliazzo described five possible computational worlds: Algorithmica, Heuristica, Pessiland, Minicrypt and Cryptomania. Algorithmica is a world where NP is easy in the worst case; Heuristica a world where NP is hard in the worst case but easy on average; Pessiland a world where NP is hard on average but one-way functions do not exist; Minicrypt a world where one-way functions exist but public-key cryptography does not; and Cryptomania a world where public-key cryptography exists. The general belief among complexity theorists and cryptographers is that we live in Cryptomania, but we are very far from a proof, as even ruling out Algorithmica would involve showing $\mathsf{NP} \neq \mathsf{P}$.

There is the possibility, however, that we might be able to unconditionally rule out some of the intermediate worlds, such as Heuristica, Pessiland and Minicrypt. Until recently, there was little progress on ruling out these intermediate worlds. All that was known was that there are various black-box and relativization barriers to ruling out these worlds.

The study of *meta-complexity*, i.e., the complexity of computational problems that are themselves about complexity, has enabled new attacks on these questions. Examples of meta-complexity problems are the Minimum Circuit Size Problem (MCSP), which asks whether a Boolean function represented by its truth table has circuits of a given size, and the problem of computing Kolmogorov complexity and its resource-bounded variants such as Levin's time-bounded Kolmogorov complexity. The average-case complexity of meta-complexity problems is of particular interest [HS17]. Hirahara [Hir20] gave an approach via meta-complexity to ruling out the analogue of Heuristica for the Polynomial Hierarchy. More recently, the Polynomial Hierarchy analogue of Pessiland has been ruled out [HS22], again using meta-complexity techniques.

There have been several successful efforts to characterize the existence of one-way functions via meta-complexity. In [San20], a conditional characterization was given, based on a believable but seemingly hard-to-establish conjecture. Liu and Pass [LP20] unconditionally characterized one-way functions by the average-case hardness of polynomial-time-bounded Kolmogorov complexity over the uniform distribution. This characterization was extended to other meta-complexity problems and notions of one-way function in [LP21, RS21, ACM+21]. A different characterization of one-way functions via the hardness of approximating Kolmogorov complexity over samplable distributions was given in [IRS22]. More recently, Hirahara [Hir23] introduced a meta-complexity problem whose NP-hardness and the worst-case hardness of NP characterize the existence of one-way functions.

These connections between meta-complexity, average-case complexity and one-way functions raise the following question: Can we characterize Impagliazzo's worlds Algorithmica, Heuristica and Pessiland by different notions of hardness for a single computational problem? A positive answer to this question is implicit in [LP22], who study the problem of conditional polynomial-time-bounded Kolmogorov complexity. They show that the worst-case hardness of conditional polynomial-time-bounded Kolmogorov complexity captures worst-case hardness of NP, and the average-case hardness of conditional polynomial-time-bounded Kolmogorov complexity over the uniform distribution captures the existence of one-way functions. Their result on worst-case hardness immediately implies that the average-case hardness of NP is equivalent to the hardness of conditional polynomial-time-bounded Kolmogorov complexity over some samplable distribution.

In this work, we give two new characterizations of Impagliazzo's worlds by different notions of hardness for a single problem - first for conditional probabilistic time-bounded Kolmogorov complexity pK^t [GKLO22], and second for the standard notion of conditional Kolmogorov complexity. These new characterizations have some interesting features. The first characterization implies that ruling out Pessiland corresponds to *robustness* of the average-case tractability of conditional pK^t over time regimes t that vary from sublinear to polynomial. As a consequence, while we are able

to prove (by building on [Hir22]) that pK^t is NP-hard to compute exactly when t is sublinear, Pessiland would fail to exist if pK^t were NP-hard to compute for *arbitrary* polynomial t. This could be a promising route to ruling out Pessiland, since pK^t is a fairly powerful complexity measure with nice properties such as the coding theorem which could potentially be exploited when showing hardness, and the computational version is in (promise) AM but is not known to be in NP.

The second characterization is for a fundamental problem that is *provably intractable in the worst case*, i.e., the problem of approximating conditional Kolmogorov complexity. A somewhat surprising aspect of our results (which is also present in the main result of [IRS22] on which we build) is that conditional Kolmogorov complexity is uncomputable, yet natural average-case hardness assumptions on conditional Kolmogorov complexity capture complexity worlds related to average-case hardness of NP. What this indicates is that the distinctions between Impagliazzo's worlds can be encoded in a natural way into the *distributional assumptions* that are made, while considering a single well-understood problem.

As a corollary of our second set of results together with those in [LP22], we get new equivalences between hardness assumptions for conditional Kolmogorov complexity and hardness assumptions for conditional time-bounded Kolmogorov complexity. The proofs of these equivalences crucially use the various characterizations of Impagliazzo's worlds, and it seems tricky to show such equivalences directly.

1.1 Results

We state our results formally in this subsection.

1.1.1 Characterizing Both $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ and Non-Existence of One-Way Functions by Average-Case Easiness of Conditional pK^t

We present a meta-complexity problem whose average-case tractability over polynomial-time samplable distributions can be used to characterize both the non-existence of one-way functions and $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$, while considering different time regimes in the measure of time-bounded Kolmogorov complexity. Specifically, we consider the problem of computing conditional probabilistic t-time-bounded Kolmogorov complexity.

As defined in [GKLO22], we let $\mathsf{pK}^t_\lambda(x \mid y)$ be the smallest integer k such that, with probability at least λ over the choice of a random string $w \sim \{0,1\}^t$, there is a (deterministic) program of size k that, when running on w and given oracle access to y, prints x within t steps (see Definition 16 for the formal definition).

For $\tau\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, let $\mathsf{Cond\text{-}pK}[\tau]$ be the following promise problem $(\mathsf{YES}, \mathsf{NO})$:

$$\mathsf{YES} := \left\{ (x, y, 1^s) \mid \mathsf{pK}^{\tau(|x|,|y|)}_{2/3}(x \mid y) \leq s \right\},$$
$$\mathsf{NO} := \left\{ (x, y, 1^s) \mid \mathsf{pK}^{\tau(|x|,|y|)}_{1/3}(x \mid y) > s \right\}.$$

We will refer to this problem as "computing conditional pK^t".

We will consider two specific settings for the time bound function τ. For the purpose of illustration, let us consider the following simplified problem. For $\tau\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, we are given x, y and s, and the task is to decide whether $\mathsf{K}^{\tau(|x|,|y|)}(x \mid y) \leq s$, i.e., whether there is a program of size at most s such that given *oracle access* to y, the program outputs x within time $\tau(|x|, |y|)$.

A typical setting of τ is $\tau(n, m) := n^c \cdot m^c$, where $c > 1$ is some constant. For this τ, we want to decide if there is a program of size at most s that, given *oracle access* to y, outputs x within time $\tau(|x|, |y|)$, and such a program has enough time to read the entire string y.

Now consider another setting of τ where $\tau(n,m) := n^c \cdot m^{1-1/c}$ for a constant $c > 1$. In this case, for a string $y \in \{0,1\}^m$, where $m := n^{2c^2}$, we have

$$\tau(n,m) = n^c \cdot m^{1-1/c} = n^{2c^2-c} \ll m.$$

Again, we want to decide if there is a program of size at most s that, given *oracle access* to y, outputs x within time $\tau(|x|,|y|)$. However, in this case any such program does not have time to read the entire string y

We will show that the non-existence of one-way functions corresponds to the average-case tractability of $\mathsf{Cond\text{-}pK}[\tau]$ over polynomial-time samplable distributions for the "polynomial-time regime" of τ, and that $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ corresponds to that of the "sublinear-time regime".[1] We state our results formally next.

For an algorithm A, $x, y \in \{0,1\}^*$, and $s \in \mathbb{N}$, we say that A *decides* $\mathsf{Cond\text{-}pK}[\tau]$ *on* $(x, y, 1^s)$ if the following holds:

$$A(x,y,1^s) = \begin{cases} 1 & \text{if } \mathsf{pK}^{\tau(|x|,|y|)}_{2/3}(x \mid y) \le s, \\ 0 & \text{if } \mathsf{pK}^{\tau(|x|,|y|)}_{1/3}(x \mid y) > s, \\ \textbf{either } 0 \textbf{ or } 1 & \text{otherwise.} \end{cases}$$

Theorem 1. *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*
2. **(Computing conditional pK^t in the polynomial-time regime is easy-on-average over samplable distributions.)**

   For every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$, every polynomial q, and for all large enough constant c, there exists a probabilistic polynomial-time algorithm A such that for all $n, m, s \in \mathbb{N}$,

   $$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}}[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x,y,1^s)] \ge 1 - \frac{1}{q(n,m)},$$

   where $\tau(n,m) := n^c \cdot m^c$.

Theorem 2. *The following are equivalent.*

1. $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$.
2. **(Computing conditional pK^t in the sublinear-time regime is easy-on-average over samplable distributions.)**

   For every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$, every polynomial q, and for all large enough constant c, there exists a probabilistic polynomial-time algorithm A such that for all $n, m, s \in \mathbb{N}$,

   $$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}}[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x,y,1^s)] \ge 1 - \frac{1}{q(n,m)},$$

   where $\tau(n,m) := n^c \cdot m^{1-1/c}$.

[1] Note that even in the "sublinear-time regime" of τ, the program can still run in polynomial time with respect to the the length of x; the word "sublinear-time" refers to the fact that the program runs in sub-linear time with respect to the length of y.

In proving Theorem 2, we also show that it is NP-hard to compute conditional pK^t in the sublinear-time regime in the worst case.

Theorem 3 (Informal)**.** *For any constant $c > 1$, $\mathsf{Cond\text{-}pK}[\tau]$ is NP-hard under randomized polynomial-time reductions, where $\tau(n, m) := n^c \cdot m^{1-1/c}$.*

In fact, Theorem 3 holds even if we consider the problem of approximating $\mathsf{pK}^t(x \mid y)$ in the sublinear-time regime within a multiplicative factor of $|x|^{1/\log\log|x|^{O(1)}}$. This also extends a result by Liu and Pass [LP22] and Hirahara [Hir22], which showed that the problem of computing/approximating conditional K^t in the sublinear-time regime is NP-hard.

Theorem 3, Theorem 1 and Theorem 2 together give characterizations of Impagliazzo's worlds Algorithmica, Heuristica and Pessiland based on different hardness assumptions for the computation of conditional pK^t.

In particular, Theorem 1 and Theorem 2 imply that the task of ruling out Pessiland[2] is equivalent to showing that the problem of computing conditional pK^t on average over polynomial-time samplable distributions is robust with respect to the two different time regimes.

Also, we get that to rule out Pessiland, it suffices to show that it is NP-hard to compute conditional pK^t in the *polynomial-time regime* in the worst case.

Corollary 4 (Informal. See Corollary 55 for the formal version)**.** *If computing conditional pK^t in the polynomial-time regime is NP-hard, then Pessiland does not exist.*

A proof sketch of Corollary 4 can be found in Section 4.4.

For comparison, it was observed in [Hir23] that if one can show the NP-hardness of *approximating* a certain variant of time-bounded Kolmogorov complexity called q^t, then Pessiland does not exist. It is known that $\mathsf{q}^{\mathsf{poly}}$ and $\mathsf{pK}^{\mathsf{poly}}$ are equivalent to each other up to an additive logarithmic factor. This implies that showing the NP-hardness of *approximating* pK^t will allow us to rule out Pessiland.[3] It can also be shown that the problem of *approximating* pK^t is reducible to that of computing conditional pK^t.[4] On the other hand, Corollary 4 only requires showing the NP-hardness of computing conditional pK^t, which might be easier. Moreover, we note that the barrier of [SS22] to showing NP-hardness of approximating Kolmogorov complexity and its variants does not seem to apply directly to exact computation.

Equivalences between Average-Case Easiness of Approximating and Computing (Conditional) pK^t. By combining Theorem 1 with existing characterizations of one-way functions, we get that the average-case easiness of approximating and computing different variants of probabilistic (conditional) time-bounded Kolmogorov complexity are in fact equivalent. We state this result more formally below.

We say that "approximating pK^t is easy-on-average over samplable distributions" if the following holds.

> For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, every polynomial q, and for all large enough polynomial τ, there is a probabilistic polynomial-time algorithm A that can decide, given as input $(x, 1^s, 1^t)$, whether $\mathsf{pK}^t(x) \leq$

[2] In this case, we mean basing infinitely-often one-way functions on $\mathsf{DistNP} \not\subseteq \mathsf{HeurBPP}$.

[3] Here, we refer to the problem called Gap-MINpKT. For a polynomial τ, $\mathsf{Gap\text{-}MINpKT}[\tau]$ is the (promise) problem of deciding, given as input $(x, 1^s, 1^t)$, whether $\mathsf{pK}^t(x) \leq s$ or $\mathsf{pK}^{\tau(|x|,t)}(x) > s + \log\tau(|x|, t)$.

[4] More precisely, if we can solve $\mathsf{Cond\text{-}pK}[\tau]$ for some polynomial τ, then we can also solve $\mathsf{Gap\text{-}MINpKT}[\tau']$ for some polynomial τ'.

s or $\mathsf{pK}^{\tau(|x|,t)}(x) > s + \log \tau(|x|,t)$,[5] with probability at least $1 - 1/q(n)$ over $x \sim \mathcal{D}_n$ and the internal randomness of A.

The above can be naturally generalized to the conditional setting, where we consider any samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$, and for all large enough polynomial τ, we can decide whether $\mathsf{pK}^t(x \mid y) \leq s$ or $\mathsf{pK}^{\tau(|x|,|y|,t)}(x \mid y) > s + \log \tau(|x|,|y|,t)$ with high probability over (x,y) sampled from $\mathcal{D}_{\langle n,m\rangle}$. In this case, we say that "approximating conditional pK^t is easy-on-average over samplable distributions"

Also, we say that "computing pK^t is easy-on-average over samplable distributions" if the following holds.

> For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, every polynomial q, and for all large enough polynomial τ, there is a probabilistic polynomial-time algorithm A that can decide, given as input $(x, 1^s)$, whether $\mathsf{pK}^{\tau(|x|)}_{2/3}(x) \leq s$ or $\mathsf{pK}^{\tau(|x|)}_{1/3}(x) > s$,[6] with probability at least $1 - 1/q(n)$ over $x \sim \mathcal{D}_n$ and the internal randomness of A.

Theorem 5 (Informal)**.** *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*
2. *Approximating pK^t is easy-on-average over samplable distributions.*
3. *Approximating conditional pK^t is easy-on-average over samplable distributions.*
4. *Computing pK^t is easy-on-average over samplable distributions.*
5. *Computing conditional pK^t is easy-on-average over samplable distributions.*

A sketch of the proof of Theorem 5 can be found in Section 3.3.

1.1.2 Characterizing Impagliazzo's Worlds by Tractability of Conditional Time-Unbounded Kolmogorov Complexity

We present a meta-complexity problem, namely approximating conditional Kolmogorov complexity up to an $O(\log n)$ additive term, that is unconditionally hard (even uncomputable) in the worst case, but such that its average-case intractability for different classes of distributions characterize Algorithmica, Heuristica and Pessiland.

Characterizing $\mathsf{DistNP} \subseteq \mathsf{BPP}$ and $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ by Tractability of Time-Unbounded Kolmogorov Complexity. To begin, we recall a recent result by Ilango, Ren, and Santhanam [IRS22] characterizing the non-existence of one-way functions by the tractability of approximating Kolmogorov complexity over polynomial-time samplable distributions. We consider the following conditional variant from [HIL+23].

Theorem 6 ([HIL+23, Lemma 27], cf. [IRS22])**.** *The following are equivalent.*

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

[5] Note that this is the problem $\mathsf{Gap\text{-}MINpKT}$ mentioned in Footnote 3.
[6] This problem is referred to as $\mathsf{MpK}^\tau\mathsf{P}$ in [LP23].

2. **(Approximating conditional Kolmogorov complexity is easy-on-average over polynomial-time samplable distributions.)**

   For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$, where each $\mathcal{D}_n$ is over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,

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

Note that a one-way function is a function that is efficiently computable but hard to invert on average; thus, this notion is based on *average-case* hardness. Theorem 6 characterizes the existence of one-way functions by the average-case hardness of approximating (conditional) Kolmogorov complexity. Then, for $\mathsf{NP} \not\subseteq \mathsf{BPP}$, which is a worst-case hardness notion, one might think that it can be characterized by the worst-case hardness of approximating (conditional) Kolmogorov complexity. However, it is well known that the task of approximating the conditional Kolmogorov complexity is provably intractable in the worst case, so such a characterization would imply $\mathsf{NP} \not\subseteq \mathsf{BPP}$ unconditionally.

Consider a polynomial-time samplable distribution $\mathcal{D}$ over $\{0,1\}^n \times \{0,1\}^n$. Also, let $\mathcal{D}^{(2)}$ be the marginal distribution of $\mathcal{D}$ on the second half, and let $\mathcal{D}(\cdot \mid y)$ denote the conditional distribution of $\mathcal{D}$ on the first half given that the second half is y. Now, observe the following equivalent way of sampling a pair of strings (x,y) from $\mathcal{D}$: We first sample y from $\mathcal{D}^{(2)}$ and then x from $\mathcal{D}(\cdot \mid y)$.

Note that Theorem 6 essentially says that one-way functions do not exist if and only if, for every polynomial-time samplable distribution $\mathcal{D}$, one can approximate $\mathsf{K}(x \mid y)$ on average over (x,y), where we sample y from $\mathcal{D}^{(2)}$ and then x from $\mathcal{D}(\cdot \mid y)$. In order to characterize $\mathsf{NP} \subseteq \mathsf{BPP}$, we consider the tractability of approximating conditional Kolmogorov complexity in the *semi-worst case*, meaning that we can approximate $\mathsf{K}(x \mid y)$ on average over x sampled from $\mathcal{D}(\cdot \mid y)$ for *all* $y \in \{0,1\}^n$ (instead of an average y from $\mathcal{D}^{(2)}$). Our first result is a characterization of $\mathsf{NP} \subseteq \mathsf{BPP}$ by the tractability of approximating conditional Kolmogorov complexity in this semi-worst case. Formally, we show the following.

Theorem 7. *The following are equivalent.*

1. $\mathsf{NP} \subseteq \mathsf{BPP}$.

2. **(Approximating conditional Kolmogorov complexity is easy in the semi-worst case.)**

   For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$, where each $\mathcal{D}_n$ is over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$ and $y \in \{0,1\}^n$,

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

Theorem 7 shows that $\mathsf{NP} \subseteq \mathsf{BPP}$ if and only if for every polynomial-time samplable distribution $\mathcal{D}$, approximating $\mathsf{K}(x \mid y)$ is easy on average over x sampled from $\mathcal{D}(\cdot \mid y)$ for *every* $y \in \{0,1\}^n$. Now, instead of considering every $y \in \{0,1\}^n$ (a worst-case notion), it is also natural to consider an average y sampled from some polynomial-time samplable distribution $\mathcal{C}$ (an average-case notion). However, the distribution $\mathcal{C}$ here can be independent of $\mathcal{D}$. In particular, it does not necessarily have to be $\mathcal{D}^{(2)}$.

Next, we show that the average-case tractability of approximating conditional Kolmogorov complexity over such *independent polynomial-time samplable distributions*, in fact characterizes the *average-case* easiness of NP (i.e., DistNP $\subseteq$ HeurBPP). We first state formally the definition of independent polynomial-time samplable distributions.

Definition 8 (Independent Polynomial-Time Samplable [HIL+23])**.** We say that a distribution family $\{\mathcal{D}_n\}_n$, where each $\mathcal{D}_n$ is over $\{0,1\}^n \times \{0,1\}^n$, is *independent polynomial-time samplable* if there exist two polynomial-time samplable distribution families $\{\mathcal{A}_n\}_n$ and $\{\mathcal{B}_n\}_n$, where each $\mathcal{A}_n$ is over $\{0,1\}^n$ and each $\mathcal{B}_n$ is over $\{0,1\}^n \times \{0,1\}^n$, such that $\mathcal{D}_n$ can be equivalently sampled as follows: sample $y \sim \mathcal{A}_n$, sample $x \sim \mathcal{B}_n(\cdot \mid y)$, and then output (x, y).

It is easy to see that every polynomial-time samplable distribution is also independent polynomial-time samplable, by letting $\mathcal{A}$ be the marginal distribution of $\mathcal{D}$ on the second half and letting $\mathcal{B}$ be $\mathcal{D}$. However, the converse is not necessarily true. Nevertheless, Theorem 6 and Theorem 9 (which we state below) imply that the task of ruling out Pessiland is equivalent to showing that the hardness of approximating conditional Kolmogorov complexity remains unchanged over these two classes of distributions.

Theorem 9. *The following are equivalent.*

1. DistNP $\subseteq$ HeurBPP.
2. **(Approximating conditional Kolmogorov complexity is easy-on-average over independent polynomial-time samplable distributions.)**

   For every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,

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

Finally, we extend Theorem 6 to characterize the non-existence of *auxiliary-input one-way functions* by the tractability of approximating conditional Kolmogorov complexity over P/poly-*samplable distributions*.

Theorem 10. *The following are equivalent.*

1. *Auxiliary-input one-way functions do not exist.*
2. *For every sequence of strings $\{y_n\}_n$ where each $y_n \in \{0,1\}^n$, every distribution family $\{\mathcal{D}_n\}_n$ samplable in polynomial time using $\{y_n\}_n$ as advice, where each $\mathcal{D}_n$ is over $\{0,1\}^n$, and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,*

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

The results above characterize the non-existence of one-way functions, DistNP $\subseteq$ HeurBPP, and NP $\subseteq$ BPP by the *distributional* tractability of approximating the conditional Kolmogorov complexity. They imply that the tasks of ruling out Impagliazzo's certain worlds are equivalent to showing that the hardness of this problem is the same with respect to different classes of distributions. For example, Theorem 6 and Theorem 9 imply that basing one-way functions on DistNP $\not\subseteq$ HeurBPP (a.k.a., ruling out Pessiland) is equivalent to showing that the hardness of approximating conditional Kolmogorov complexity over polynomial-time samplable distributions is the same as the hardness over independent polynomial-time samplable distributions.

Equivalences between Tractability of Time-Unbounded and Time-Bounded Kolmogorov Complexity. We first recall the definition of time-bounded Kolmogorov complexity. For $x, y \in \{0,1\}^*$ and $t \in \mathbb{N}$, we define $\mathsf{K}^t(x \mid y)$ to be the minimum length of a program $p \in \{0,1\}^*$ such that $U^y(p)$ outputs x within t steps. Here, U is a fixed time-optimal universal Turing machine and has oracle access to the string y.

For $\tau\colon \mathbb{N} \to \mathbb{N}$ and $\kappa\colon \mathbb{N} \to \mathbb{N}$, let $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ be the problem where we are given input $x \in \{0,1\}^n$ and $y \in \{0,1\}^{\kappa(n)}$, and we are asked to compute $\mathsf{K}^{\tau(|x|)}(x \mid y)$. Given a polynomial τ and a polynomial κ, we say that:

- $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is *easy in the worst case* if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ can be solved in polynomial time.
- $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is *easy-on-average over polynomial-time samplable distributions* if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ admits a heuristic scheme. That is for any polynomial-time samplable distribution $\mathcal{D} = \{D_n\}_n$, where each $\mathcal{D}_n$ samples (x, z) with $x \in \{0,1\}^n$ and $y \in \{0,1\}^{\kappa(n)}$, there exists a probabilistic polynomial-time algorithm A such that for all $n, k \in \mathbb{N}$,
$$\Pr_{x,y\sim\mathcal{D}_n}\left[A(x, y; 1^n, 1^k) = \mathsf{K}^{\tau(|x|)}(x \mid y)\right] \geq 1 - \frac{1}{k}.$$
- $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is *easy-on-average over the uniform distribution* if for every polynomial p, there exists a probabilistic polynomial-time algorithm A such that for all $n \in \mathbb{N}$,
$$\Pr_{x\sim\{0,1\}^n,y\sim\{0,1\}^{\kappa(n)}}\left[A(x, y) = \mathsf{K}^{\tau(|x|)}(x \mid y)\right] \geq 1 - \frac{1}{p(n)}.$$

Theorem 11 (Implicit in [LP22])**.** *The following hold.*

- *For all polynomial $\tau(n) \geq n^2$, there exists a polynomial κ such that $\mathsf{NP} \subseteq \mathsf{BPP}$ if and only if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is easy in the worst-case.*
- *For all polynomial $\tau(n) \geq n^2$, there exists a polynomial κ such that $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ if and only if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is easy-on-average over polynomial-time samplable distributions.*
- *For every polynomial $\tau(n) \geq 1.1n$ and polynomial κ, infinitely-often one-way functions do not exist if and only if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is easy-on-average over the uniform distribution.*

As a corollary, we get the following equivalences between the tractability of conditional Kolmogorov complexity and that of conditional time-bounded Kolmogorov complexity.

Corollary 12 (Informal)**.** *The following hold.*

- *For all polynomial $\tau(n) \geq n^2$, there exists a polynomial κ such that approximating conditional Kolmogorov complexity is easy in the semi-worst case if and only if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is easy in the worst-case case.*
- *For all polynomial $\tau(n) \geq n^2$, there exists a polynomial κ such that approximating conditional Kolmogorov complexity is easy-on-average over independent polynomial-time samplable distributions if and only if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is easy-on-average over polynomial-time samplable distributions.*
- *For every polynomial $\tau(n) \geq 1.1n$ and polynomial κ, approximating conditional Kolmogorov complexity is easy-on-average over polynomial-time samplable distributions if and only if $\mathsf{McK}^\tau\mathsf{P}[\kappa]$ is easy-on-average over the uniform distribution.*

Proof. This follows directly from Theorem 6, Theorem 7, Theorem 9, and Theorem 11. □

1.2 Techniques

In this section, we explain the main ideas behind our proofs.

Characterizing Non-Existence of One-Way Functions by Average-Case Easiness of Conditional pK^t. A recent result by Liu and Pass [LP23] characterized the non-existence of (infinitely-often) one-way functions by the average-case easiness of computing pK^t over polynomial-time samplable distributions. Here, we describe a proof of this result that is slightly different than the original one and show how to generalize it to *conditional* pK^t.

It will be convenient to think of the pK^t complexity of a string as its K^t complexity *conditioning on a random string* r (see Proposition 17).

First of all, by employing ideas from [LP20, LP23], one can construct a function, which outputs the string x produced by a randomly selected (time-bounded) program (resp. conditioning on a random string r), and show that if this function can be inverted, then we can obtain a shortest program for x (resp. conditioning on r) "on average". In particular, it can be shown that if infinitely-often one-way functions do not exist, then for every time bound function $\tau(n) = n^{O(1)}$, there exists an efficient algorithm A (for simplicity, think of it as being deterministic) such that with high probability over a uniformly random string r, $A(x; r)$ computes $\mathsf{K}^\tau(x \mid r)$ for an average x sampled from some distribution $\mathcal{E}^\tau_r$, defined as $\mathcal{E}^\tau_r(x) := 2^{-\mathsf{K}^\tau(x|r)}$.

Next, we want to say that, for almost all r, the algorithm $A(-; r)$, which works for the distribution $\mathcal{E}^\tau_r$, also works for a given polynomial-time samplable distribution $\mathcal{D}$ (provided that τ is a sufficiently large polynomial). To get this, it suffices to show that $\mathcal{E}^\tau_r$ *dominates*[7] $\mathcal{D}$, i.e., $2^{-\mathsf{K}^\tau(x|r)} \gtrsim \mathcal{D}(x)$ for *every* x. The observation here is that this follows from the recently discovered coding theorem for $\mathsf{pK}^{\mathsf{poly}}$ [LOZ22], which asserts that for *every* string x, $\mathsf{pK}^\tau(x) \lesssim \log(1/\mathcal{D}(x))$ (again, provided that τ is a sufficiently large polynomial). To see this, note that by the definition of pK^t, we have for a uniform random r, $\mathsf{K}^\tau(x \mid r) \leq \mathsf{pK}^\tau(x)$.

Given the above, we have that with high probability over a uniformly random r, $A(x; r) = \mathsf{K}^\tau(x \mid r)$ for an average x sampled from $\mathcal{D}$. By an averaging argument, we get that with high probability over $x \sim \mathcal{D}$, $A(x; r) = \mathsf{K}^\tau(x \mid r)$ with high probability over a uniformly random r. For any such *good* x, if $\mathsf{pK}^\tau_{2/3}(x) \leq s$ (resp. $\mathsf{pK}^\tau_{1/3}(x) > s$), which means $\mathbf{Pr}_r[\mathsf{K}^\tau(x \mid r) \leq s] \geq 2/3$ (resp. $\mathbf{Pr}_r[\mathsf{K}^\tau(x \mid r) > s] \geq 2/3$), then $A(x, r) \leq s$ (resp. $A(x, r) > s$) with high probability over r. This allows us to solve the problem of computing pK^τ on average over the distribution $\mathcal{D}$.

Now we describe how to generalize the above to *conditional* pK^t.

Suppose we want to compute $\mathsf{pK}^\tau(x \mid y)$ over (x, y) sampled from some polynomial-time distribution $\mathcal{D}$. It will be convenient to consider the following equivalent way of sampling $\mathcal{D}$: We first sample $y \sim \mathcal{D}^{(2)}$, where $\mathcal{D}^{(2)}$ is the marginal distribution of $\mathcal{D}$ on the second half, and then sample $x \sim \mathcal{D}(\cdot \mid y)$, where $\mathcal{D}(\cdot \mid y)$ is the conditional distribution of $\mathcal{D}_n$ on the first half given that the second half is y. Finally, we output (x, y).

First of all, by modifying the construction of the candidate one-way function described above (e.g., by incorporating the distribution $\mathcal{D}^{(2)}$ into the construction), we can show that if infinitely-often one-way functions do not exist, then there exists an efficient algorithm A such that with high probability over a uniformly random string r and over y sampled from $\mathcal{D}^{(2)}$, $A(x; y, r)$ computes $\mathsf{K}^\tau(x \mid y, r)$ for an average x sampled from some distribution $\mathcal{E}^\tau_{y,r}$, where $\mathcal{E}^\tau_{y,r}(x) := 2^{-\mathsf{K}^\tau(x|y,r)}$.

Now similar to the previous case, we want to say that, with high probability over r and $y \sim \mathcal{D}^{(2)}$, the algorithm $A(-; y, r)$, which works for the distribution $\mathcal{E}^\tau_{y,r}$, also works for the distribution $\mathcal{D}(\cdot \mid y)$. Again, it suffices to show that $\mathcal{E}^\tau_{y,r}(x) = 2^{-\mathsf{K}^\tau(x|y,r)} \gtrsim \mathcal{D}(x \mid y)$ for every x. However, this

[7] 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.

would require a conditional version of the coding theorem for $\mathsf{pK}^{\mathsf{poly}}$ applying to the distribution $\mathcal{D}(\cdot \mid y)$ (which is not necessarily efficiently samplable given y). Such a coding theorem is not known (in fact, is unlikely to hold).

The key observation is that in order to show that the algorithm $A(-; y, r)$, which works on average over the distribution $\mathcal{E}^{\tau}_{y,r}$, also works for $\mathcal{D}(\cdot \mid y)$, it suffices to have that $\mathcal{E}^{\tau}_{y,r}(x)$ dominates $\mathcal{D}(x \mid y)$ *on almost all* x, instead of every x. Then this weaker condition can be obtained from an *average-case* coding theorem for $\mathsf{pK}^{\mathsf{poly}}$, which has been shown under the assumption that infinitely-often one-way functions do not exist [HIL+23] (see Theorem 29).

More specifically, [HIL+23] showed that if infinitely-often one-way functions do not exist, then with high probability over $y \sim \mathcal{D}^{(2)}$ and $x \sim \mathcal{D}(\cdot \mid y)$, it holds that

$$\mathsf{pK}^{\tau}(x \mid y) \lesssim \log \frac{1}{\mathcal{D}(x \mid y)}.$$

Again, by the definition of pK^t and an averaging argument, this yields that with high probability over a uniformly random r and $y \sim \mathcal{D}^{(2)}$,

$$\mathsf{K}^{\tau}(x \mid y, r) \leq \log \frac{1}{\mathcal{D}(x \mid y)}$$

holds for almost all $x \sim \mathcal{D}(\cdot \mid y)$. This allows us to say that with high probability over r and $y \sim \mathcal{D}^{(2)}$, the distribution $\mathcal{E}^{\tau}_{y,r}$ dominates $\mathcal{D}(\cdot \mid y)$ *on average*, so the algorithm $A(-; y, r)$, which works for $\mathcal{E}^{\tau}_{y,r}$, also works for $\mathcal{D}(\cdot \mid y)$.

At this point, we get that with high probability over $(x, y) \sim \mathcal{D}$ and over a uniformly random r, $A(x; y, r) = \mathsf{K}^{\tau}(x \mid y, r)$. By the same argument as described above, this allows us to compute $\mathsf{pK}^{\tau}(x \mid y)$ on average over (x, y) sampled from $\mathcal{D}$.

The converse direction, i.e., that computing conditional pK^t on average allows us to break one-way functions, follows from the standard observation that computing pK^t on average over samplable distributions allows us to distinguish pseudo-random distributions (which are supported on strings of low pK^t complexity) from random strings (which have high pK^t complexity).

Characterizing $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ by Average-Case Easiness of Conditional pK^t in Sublinear-Time Regime. To show that the average-case easiness of computing conditional pK^t (in the sublinear-time regime) implies the average-case easiness of NP (both with respect to polynomial-time samplable distributions), we first show that it is NP-hard to compute conditional pK^t (again, in the sublinear-time regime). Recently, Liu and Pass [LP22] and Hirahara [Hir22] showed that the problem of computing the conditional K^t in the sublinear-time regime is NP-hard. We generalize this result to pK^t.

At a high level, our proof follows a similar approach but also requires some crucial observations to address the more complex notion of pK^t and to make it applicable to show Theorem 2. In particular, we adapt the proof in [Hir22] which relies on the use of a *secret sharing scheme* (see [Hir22, Section 2.3] for an exposition). More specifically, it reduces the problem of approximating the hamming weight of a *minimum satisfying assignment* of a given monotone formula, which is known to be NP-hard, to that of computing conditional K^t in the sublinear-time regime. That is, for every constant $c > 1$ and time bound function $\tau(n, m) := n^c \cdot m^{1-1/c}$, there is a randomized reduction R such that if a given monotone formula ψ has a satisfying assignment of hamming weight at most ζ (resp. much larger than ζ), then with high probability, R produces a pair of strings (x, y) and ρ such that $\mathsf{K}^{\tau(|x|,|y|)}(x \mid y) \leq \rho$ (resp. $\mathsf{K}^{\tau(|x|,|y|)}(x \mid y) > \rho$).

Our key observation is that this reduction still works in the presence of any fixed string r. Roughly put, the reason for this is that a secret sharing scheme remains secure even if an adversary

has access to some fixed string. More specifically, we can show that with respect to any string r, if a given monotone formula ψ has a satisfying assignment of hamming weight much larger than ζ, then with high probability the algorithm R produces a pair of strings (x, y) and ρ such that $\mathsf{K}^{\tau(|x|,|y|)}(x \mid y, r) > \rho$. This allows us to say that if the minimum weight of ψ is much larger than ζ, then with high probability over a random string r and over the internal randomness of R, $\mathsf{K}^{\tau(|x|,|y|)}(x \mid y, r) > \rho$. By an averaging argument, this gives that with high probability over the internal randomness of R, $\mathsf{K}^{\tau(|x|,|y|)}(x \mid y, r) > \rho$ for more than $2/3$ of the r's, which essentially means $\mathsf{pK}^{\tau(|x|,|y|)}_{1/3}(x \mid y) > \rho$.

Now we have showed that computing conditional pK^t (in the sublinear-time regime) is NP-hard. To solve an NP problem L over a given polynomial-time samplable distribution $\mathcal{D}$, we can compose $\mathcal{D}$ with the reduction R to obtain a new distribution $\mathcal{D}'$. Then we can show that computing conditional pK^t on average over $\mathcal{D}'$ will allow us to solve L on average over $\mathcal{D}$. However, there is an additional subtle issue here, the original reduction R depends on the time bound function (i.e., for every sublinear time bound τ, there is a reduction R that will work). On the other hand, to show Theorem 2 (Item 2 $\implies$ Item 1), it is required that the reduction works for all time bound functions τ of the form $\tau(n, m) = n^c \cdot m^{1-1/c}$. We will then need to further modify the reduction to achieve this. (See Lemma 45 for the details.)

Now we need to show the other direction saying that the average-case easiness of NP implies the average-case easiness of computing conditional pK^t. Unlike the problem of computing (conditional) K^t, computing (conditional) pK^t is not known to be in NP, so we can not get the desired implication directly. However, it is not hard to see that the problem of computing conditional pK^t is in fact in (promise) AM.[8] If we can solve NP, then we can also solve AM (in the randomized setting), by a standard trick that combines the instance of an AM problem with a random string to produce an instance for an NP problem. (See Lemma 53 for the details.)

Characterizing $\mathsf{DistNP} \subseteq \mathsf{BPP}$ and $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ by Tractability of Time-Unbounded Kolmogorov Complexity. First, we recap the proof of Theorem 6 as presented in [IRS22]. We will ignore the issue of "infinitely often" in this subsection.

To show that the non-existence of one-way functions implies efficient algorithms for approximating conditional Kolmogorov complexity on average over polynomial-time samplable distributions, we use a powerful result from [IL90], which asserts that if one-way functions do not exist, then for any polynomial-time samplable distribution $\mathcal{D}$ over $\{0,1\}^n \times \{0,1\}^n$, one can efficiently estimate $\mathcal{D}(x \mid y)$ on average over $(x, y) \sim \mathcal{D}$. In addition, we combine two fundamental properties related to time-unbounded Kolmogorov complexity: The first is called the coding theorem, which roughly says that for every $(x, y) \in \mathsf{Support}(\mathcal{D})$,

$$\mathsf{K}(x \mid y) \lesssim \log \frac{1}{\mathcal{D}(x \mid y)},$$

and the second is the incompressibility property, which states that all $y \in \{0,1\}^n$ and for almost all $x \sim \mathcal{D}(\cdot \mid y)$,

$$\mathsf{K}(x \mid y) \gtrsim \log \frac{1}{\mathcal{D}(x \mid y)}.$$

It follows that for almost all $(x, y) \sim \mathcal{D}$,

$$\mathsf{K}(x \mid y) \approx \log \frac{1}{\mathcal{D}(x \mid y)}.$$

[8] Here, we refer to the problem $\mathsf{Cond\text{-}pK}$ instead of the one that asks to decide whether $\mathsf{pK}^{\tau(|x|,|y|)}(x \mid y) \leq s$ for a given input $(x, y, 1^s)$ and time bound τ.

This allows us to approximate $\mathsf{K}(x \mid y)$ by estimating $\mathcal{D}(x \mid y)$, and the latter be done efficiently if one-way functions do not exist.

For the other direction, the idea is that an efficient algorithm for approximating Kolmogorov complexity on average can be used to construct a function that distinguishes the output distribution of a cryptographic pseudorandom generator from the uniform distribution. Intuitively, this is because the outputs of such a generator have low $\mathsf{K}^{\mathsf{poly}}$ complexity while a random string has high Kolmogorov complexity. Then such an algorithm implies the non-existence of pseudorandom generators and hence of one-way functions [HILL99].

Now, let us try to see if we can adapt the above proof paradigm to show Theorem 9, which characterizes $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ by the tractability of approximating conditional Kolmogorov complexity on average over *independent polynomial-time samplable distributions*.

One direction is in fact easy by using tools developed in [HIL+23]. In particular, it is observed in [HIL+23] that if $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$, then every independent polynomial-time samplable distribution can be simulated by some polynomial-time samplable distribution (see Lemma 26). Consequently, if $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ (which also implies that one-way functions do not exist), then we can reduce the task of approximating conditional Kolmogorov complexity over independent polynomial-time samplable distributions to that of approximating conditional Kolmogorov complexity over polynomial-time samplable distributions, which is tractable if one-way functions do not exist.

However, for the other direction, it is unclear how we can get $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ from the tractability of approximating conditional Kolmogorov complexity over independent polynomial-time samplable distributions, by using ideas from the proof of the characterization for one-way functions. In that scenario, we use the algorithm for approximating conditional Kolmogorov complexity as a distinguisher to break the security of a cryptographic pseudorandom generator.

Here, we will use a different approach. Specifically, we rely on a recently discovered characterization of $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ by the validity of a certain property called *conditional coding* for pK^t. More precisely, the authors of [HIL+23] showed that $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ if and only if conditional coding property for $\mathsf{pK}^{\mathsf{poly}}$ holds on average over pairs of strings drawn from independent polynomial-time samplable distributions, i.e., for any independent polynomial-time samplable distribution $\mathcal{D}$ over $\{0,1\}^n \times \{0,1\}^n$ and for almost all $(x, y) \sim \mathcal{D}$,

$$\mathsf{pK}^{\mathsf{poly}(n)}(x \mid y) \lesssim \log \frac{1}{\mathcal{D}(x \mid y)}$$

(see Theorem 30).

Now given this characterization of $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ using conditional coding, it suffices to show that conditional coding property for $\mathsf{pK}^{\mathsf{poly}}$ over independent polynomial-time samplable distributions follows from the tractability of approximating conditional Kolmogorov complexity over the same class of distributions.

How can we show this? First of all, note that by the coding theorem for *time-unbounded* Kolmogorov complexity, we have that for every $(x, y) \in \mathsf{Support}(\mathcal{D})$,

$$\mathsf{K}(x \mid y) \lesssim \log \frac{1}{\mathcal{D}(x \mid y)}.$$

Then to get the desired conditional coding property for $\mathsf{pK}^{\mathsf{poly}}$, it suffices to show that for almost all $(x, y) \sim \mathcal{D}$,

$$\mathsf{pK}^{\mathsf{poly}(n)}(x \mid y) \leq \mathsf{K}(x \mid y) + O(\log n). \tag{1}$$

Now, let us describe how to show the above, assuming efficient algorithms for approximating conditional Kolmogorov complexity over independent polynomial-time samplable distributions.

The key ingredient here is a pseudorandom generator construction with reconstruction property. Such a generator is instantiated with a target string, it then takes as input a random seed and outputs a string that is longer than the seed. The reconstruction property allows us to say that if there exists a function that can distinguish the output distribution of the generator from the uniform distribution, then it can be used to recover the target string, using an additional advice string. This enables us to say that given a distinguisher, the target string has poly-time-bounded Kolmogorov complexity bounded by the length of the advice string. An algorithm for approximating Kolmogorov complexity can naturally be used as such a distinguisher, since the outputs of the generator have low Kolmogorov complexity while a random string has high Kolmogorov complexity. By appropriately configuring the parameters of the generator, we can ensure that the length of the advice string is comparable to the Kolmogorov complexity of the target string. This allows us to upper bound the poly-time-bounded Kolmogorov complexity of the target string by its Kolmogorov complexity.

Using this approach, the authors of [HIL+23] showed that if efficient algorithms exist for approximating conditional Kolmogorov complexity over polynomial-time samplable distributions, then for every polynomial-time samplable distribution $\mathcal{D}$ over $\{0,1\}^n \times \{0,1\}^n$ and almost all $(x, y) \sim \mathcal{D}$,

$$\mathsf{rK}^{\mathsf{poly}(n)}(x \mid y) \leq \mathsf{K}(x \mid y) + O(\log^3 n). \tag{2}$$

Here, rK^t is a certain randomized variant of time-bounded Kolmogorov complexity measure [BLvM05, LOS21].

The $O(\log^3 n)$ additive term in Equation (2) results from the use of a specific pseudorandom generator construction with an rK^t-style reconstruction property (as they need to upper bound $\mathsf{rK}^{\mathsf{poly}}$ by K), and such a generator has sub-optimal "advice complexity" in its reconstruction. In our case, we need to upper bound $\mathsf{pK}^{\mathsf{poly}}$ by K, and we can use a different pseudorandom generator construction with a pK^t-style reconstruction property that is known to have optimal "advice complexity" (see Section 2.7). This results in only an $O(\log n)$ additive term instead of $O(\log^3 n)$ as in the previous case.

The description provided above does not address an important technical distinction between showing Equation (1) and showing Equation (2) in [HIL+23]. In our case, we need to show Equation (1) over *independent polynomial-time samplable distributions*, whereas the other case involves the simpler class of *polynomial-time samplable samplable distributions*. In fact, in the proof of Equation (2), a crucial fact used is that the uniform mixture of two polynomial-time samplable distributions is also polynomial-time samplable. Intuitively, the reason why this is needed is that we need to obtain a function that can distinguish the output distribution of a pseudorandom generator (induced by a polynomial-time samplable distribution) and the uniform distribution (also combined with a polynomial-time samplable distribution), so we need to apply an algorithm to approximate Kolmogorov complexity over the mixture uniform of those two distributions.

However, in our case, we are dealing with independent polynomial-time samplable distributions, and the uniform mixture of two independent polynomial-time samplable distributions is not necessarily independent polynomial-time samplable. The key insight here is that we don't really need to be concerned with the uniform mixture of two *generic* independently polynomial-time samplable distributions. Instead, the two distributions have the property that they are identical when restricted to the second half. We then show that the uniform mixture of such two distributions remains independently polynomial-time samplable. (See the proofs of Lemma 59 and Lemma 63 for details.)

We now describe the proof of Theorem 7. Again, the direction of showing the tractability of approximating conditional Kolmogorov complexity in the semi-worst case from $\mathsf{NP} \subseteq \mathsf{BPP}$ can be done in a way similar to that of Theorem 6 (as described earlier in this subsection). This is because

if $\mathsf{NP} \subseteq \mathsf{BPP}$, then one can estimate $\mathcal{D}(x \mid y)$ for every polynomial-time samplable distribution $\mathcal{D}$ and $(x, y) \in \mathsf{Support}(\mathcal{D})$, a result due to [Sto85] (see also Lemma 27).

For the other direction, we will employ the same approach as used to to show Theorem 9. In this case, we will use a similar characterization of $\mathsf{NP} \subseteq \mathsf{BPP}$ through conditional coding. Specifically, it has been shown in [HIL+23] that $\mathsf{NP} \subseteq \mathsf{BPP}$ if and only if *worst-case* conditional coding for $\mathsf{pK}^{\mathsf{poly}}$ holds, i.e., for every polynomial-time samplable distribution $\mathcal{D}$ over $\{0,1\}^n \times \{0,1\}^n$ and *every* $(x, y) \in \mathsf{Support}(D)$,

$$\mathsf{pK}^{\mathsf{poly}(n)}(x \mid y) \lesssim \log \frac{1}{\mathcal{D}(x \mid y)}. \tag{3}$$

Unfortunately, it is unclear how we can obtain the above worst-case conditional coding property from the tractability of approximating conditional Kolmogorov complexity in the *semi-worst case* by following the same approach. To overcome this, we observe that we can modify the original proof in [HIL+23] to obtain a characterization of $\mathsf{NP} \subseteq \mathsf{BPP}$ by *semi-worst-case* conditional coding, which only requires Equation (3) to hold for almost all $x \sim \mathcal{D}(\cdot \mid y)$ and for all $y \in \{0,1\}^n$ (see Lemma 64).

By using this alternative characterization and addressing a similar issue that arises when transitioning from polynomial-time samplable distributions to semi-worst-case input distributions, as described above in the case of showing Theorem 9, we can now use efficient algorithms for approximating conditional Kolmogorov complexity in the semi-worst case to obtain the desired semi-worst-case conditional coding property, which then yields $\mathsf{NP} \subseteq \mathsf{BPP}$.

1.3 Open Problems

Can we show NP-hardness of computing conditional pK^t in the polynomial-time regime? By Corollary 4, this would imply that Pessiland does not exist. Are there any barriers to showing such an NP-hardness result?

Theorem 9 characterizes the *error-prone* average-case easiness of NP (i.e., $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$) by the tractability of approximating conditional Kolmogorov complexity over independent polynomial-time samplable distributions. Can we obtain a similar characterization for the *errorless* average-case easiness of NP (i.e., $\mathsf{DistNP} \subseteq \mathsf{AvgBPP}$)?

2 Preliminaries

2.1 Notation

We will primarily consider distributions supported over pairs of strings. Unless specified otherwise, we use $\mathcal{D} = \{\mathcal{D}_n\}_{n \in \mathbb{N}}$ to denote a family of polynomial-time samplable distributions[9], where each $\mathcal{D}_n$ is supported over $\{0,1\}^n \times \{0,1\}^n$. We let PSamp be the collection of distribution families that can be sampled in polynomial time. When n is clear from context, we might simply write $\mathcal{D}$ instead of $\mathcal{D}_n$. For $y \in \{0,1\}^n$, we denote by $\mathcal{D}_n(\cdot \mid y)$ the conditional distribution of $\mathcal{D}_n$ on the first half given that the second half is y. Also, we use $\mathcal{D}^{(2)}$ to refer to the marginal distribution of the second half of $\mathcal{D}$.

We use $\mathcal{D}_n(x, y)$ to denote the probability that the pair (x, y) is sampled from $\mathcal{D}_n$. Similarly, $\mathcal{D}_n(x \mid y)$ denotes the probability that x is sampled from the conditional distribution $\mathcal{D}_n(\cdot \mid y)$.

[9] Recall that $\mathcal{D}$ can be sampled in polynomial time if there is a polynomial-time algorithm Samp such that $\mathsf{Samp}(1^n, r)$ is distributed according to $\mathcal{D}_n$ when r is a uniformly random string of length $\mathsf{poly}(n)$.

2.2 Average-Case Complexity

Recall that a pair $(L, \mathcal{D})$ is a *distributional problem* if $L \subseteq \{0,1\}^*$ and $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ is a distribution family, where each $\mathcal{D}_n$ is over $\{0,1\}^*$.

We let DistNP denote the set of distributional problems $(L, \mathcal{D})$ with $L \in \mathsf{NP}$ and $\mathcal{D} \in \mathsf{PSamp}$.

A distributional problem $(L, \mathcal{D})$ is said to admit a (error-prone) *heuristic scheme* if there exists a probabilistic polynomial-time algorithm A such that for every $n, k \in \mathbb{N}$,

$$\Pr_{x\sim\mathcal{D}_n, A}\Big[A(x; 1^n, 1^k) \neq L(x)\Big] \leq 1/k.$$

We let $\mathsf{HeurBPP}$ denote the set of distributional problems that admit a heuristic scheme. For more information about average-case complexity, we refer to [BT06].

Lemma 13 ([Imp95, Proposition 3])**.** *If every distributional* NP *problem has a probabilistic polynomial-time heuristic algorithm of failure probability at most* n^{-2} *over the choice of instances (where* n *is an instance size), then* $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$*.*

2.3 Kolmogorov Complexity

We recall the definition of Kolmogorov complexity.

Definition 14 (Kolmogorov Complexity)**.** Let $x, y \in \{0,1\}^*$. We define

$$\mathsf{K}(x \mid y) = \min_{p\in\{0,1\}^*}\{|p| \mid U^y(p) \text{ halts and outputs } x.\}$$

Here, U is a fixed time-optimal universal Turing machine and has oracle access to the string y.

Definition 15 (Time-Bounded Kolmogorov Complexity)**.** Let $x, y \in \{0,1\}^*$ and $t \in \mathbb{N}$. The *t-time-bounded Kolmogorov complexity of x* is defined as

$$\mathsf{K}^t(x \mid y) = \min_{p\in\{0,1\}^*}\{|p| \mid U^y(p) \text{ outputs } x \text{ within } t \text{ steps.}\}$$

We also need the following probabilistic variant of time-bounded Kolmogorov complexity.

Definition 16 ([GKLO22])**.** Let $x, y \in \{0,1\}^*$, $t \in \mathbb{N}$, and $\lambda \in [0,1]$. The *probabilistic t-time-bounded Kolmogorov complexity of x given y* is defined as

$$\mathsf{pK}^t_\lambda(x \mid y) = \min\left\{k \in \mathbb{N} \;\middle|\; \Pr_{r\sim\{0,1\}^t}\Big[\exists p \in \{0,1\}^k \text{ s.t. } U^y(p, r) \text{ outputs } x \text{ within } t \text{ steps}\Big] \geq \lambda\right\}.$$

We omit the subscript λ when $\lambda = 2/3$.

The following follows easily from the definition of pK^t.

Proposition 17. *For any sufficiently large $x, y \in \{0,1\}^*$, any $t \in \mathbb{N}$, and $\lambda \in [0,1]$,*

$$\Pr_{r\sim\{0,1\}^{\mathsf{poly}(t)}}\Big[\mathsf{K}^{\mathsf{poly}(t)}(x \mid y, r) \leq \mathsf{pK}^t_\lambda(x \mid y) + O(\log t)\Big] \geq \lambda.$$

Theorem 18 (Coding Theorem)**.** *There exists a universal constant $b > 0$ such that for every computable distribution family $\{\mathcal{E}_n\}_n$, where each $\mathcal{E}_n$ is over $\{0,1\}^n$, every $n \in \mathbb{N}$ and $x \in \mathsf{Support}(\mathcal{E}_n)$,*

$$\mathsf{K}(x \mid \mathcal{E}_n) \leq \log\frac{1}{\mathcal{E}_n(x)} + b \cdot \log n.$$

Lemma 19 (See, e.g., [HIL+23, Lemma 9]). *There exists a universal constant $b > 0$ such that for every distribution family $\{\mathcal{E}_n\}_n$, where each $\mathcal{E}_n$ is over $\{0,1\}^n$, and every $\{y_m\}_m$, where each $y_m \in \{0,1\}^m$,*

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

Lemma 20 ([GKLO22]). *For any $x, y \in \{0,1\}^*$ and time bound $t \in \mathbb{N}$, we have*

$$\mathsf{K}(x \mid y, t) \le \mathsf{pK}^t(x \mid y) + O(\log(|x| \cdot |y|)).$$

Lemma 21 (Success Amplification [GKLO22]). *For any $x, y \in \{0,1\}^*$, time bound $t \in \mathbb{N}$, and $0 \le \alpha < \beta \le 1$, we have*

$$\mathsf{pK}_\beta^{O(qt/\alpha)}(x \mid y) \le \mathsf{pK}_\alpha^t(x \mid y) + O(\log(q/\alpha)),$$

where $q = \ln(1/(1-\beta))$.

2.4 Cryptography

We recall the definitions of one-way functions and auxiliary-input one-way functions.

Definition 22 (One-Way Functions). A polynomial-time computable function $f\colon \{0,1\}^* \to \{0,1\}^*$ is said to be a *one-way function* if for every probabilistic polynomial-time algorithm A and polynomial p, and for every sufficiently large n, we have

$$\Pr_{x\sim\{0,1\}^n}[f(A(1^n, f(x))) = f(x)] < \frac{1}{p(n)}. \tag{4}$$

Also, we say that f is an *infinitely-often one-way function* if for every probabilistic polynomial-time algorithm A and polynomial p, there are infinitely many values of n such that Equation (4) holds.

Definition 23 (Auxiliary-Input One-Way Functions). Let s, ℓ be polynomials. We say that a collection of functions $\left\{f_z\colon \{0,1\}^{s(|z|)} \to \{0,1\}^{\ell(|z|)}\right\}_{z\in\{0,1\}^*}$ is an *auxiliary-input one-way function* if

1. f_z is polynomial-time computable given z, and
2. for every probabilistic polynomial-time algorithm A and every polynomial p, there exist infinitely many $z \in \{0,1\}^n$ such that

$$\Pr_{x\sim\{0,1\}^{s(n)},A}[f_z(A(z, f_z(x))) = f_z(x)] < \frac{1}{p(n)}.$$

We will also need the notion of auxiliary-input pseudorandom generators.

Definition 24 (Auxiliary-Input Pseudorandom Generators). Let s, ℓ be polynomials such that $s(n) < \ell(n)$. We say that $\left\{G_z\colon \{0,1\}^{s(|z|)} \to \{0,1\}^{\ell(|z|)}\right\}_{z\in\{0,1\}^*}$ is an *auxiliary-input pseudorandom generator* if

1. G_z is polynomial-time computable given z, and

2. for every probabilistic polynomial-time algorithm D and every polynomial p, there exist infinitely many $z \in \{0,1\}^n$ such that

$$\left| \Pr_{r\sim\{0,1\}^{s(n)},D}[D(z,G_z(r))=1] - \Pr_{u\sim\{0,1\}^{\ell(n)},D}[D(z,u)=1] \right| < \frac{1}{p(n)}.$$

It is well known in cryptography that the existence of auxiliary-input one-way functions and auxiliary-input pseudorandom generators is equivalent [HILL99].

We also need the following technical theorem regarding inverting auxiliary-input one-way functions and estimating the probability of P/poly-samplable distributions.

Theorem 25 (Following [IL90]. See also [Nan21, Lemma 15])**.** *Suppose auxiliary-input one-way functions do not exist. Let s be a polynomial and $\{\mathcal{D}_n\}_n$ be a family distributions samplable by a family of s-size circuits $\{C_n\colon \{0,1\}^{s(n)} \to \{0,1\}^n\}_n$. Let $\alpha, \beta \geq 1$ be constants. There exists a probabilistic polynomial-time algorithm B such that for all $n \in \mathbb{N}$,*

$$\Pr_{z\sim\mathcal{D}_n,B}\left[\frac{\mathcal{D}_n(z)}{\beta} \leq B(C_n,z) \leq \mathcal{D}_n(z)\right] \geq 1 - \frac{1}{n^\alpha}.$$

2.5 Probability Distributions

Lemma 26 (Implicit in [HIL+23, Section 5.1])**.** *Assume $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$. Then for every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, where each $\mathcal{D}_n$ is over $\{0,1\}^n \times \{0,1\}^n$, and for every polynomial q, there exists a polynomial-time samplable distribution family $\{\mathcal{D}'_n\}_{n\in\mathbb{N}}$ such that for every $n \in \mathbb{N}$, $\mathsf{L}_1(\mathcal{D}_n, \mathcal{D}'_n) \leq 1/q(n)$.*

Lemma 27 ([HIL+23, Lemma 49])**.** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$, where each $\mathcal{D}_n$ is over $\{0,1\}^n \times \{0,1\}^n$, there exists a polynomial-time deterministic algorithm B with access to a Σ_2^P-oracle such that for input $(x,y) \in \mathsf{Support}(\mathcal{D}_n)$,*

$$\mathcal{D}_n(x \mid y)/2 \;\leq\; B(x,y) \;\leq\; 2 \cdot \mathcal{D}_n(x \mid y).$$

Theorem 28 ([IL90, IL89]. See also [IRS21, Theorem 20])**.** *Assume infinitely-often one-way functions do not exist. Let $\{\mathcal{D}_\ell\}_\ell$ be a family of polynomial-time samplable distributions, and let $\alpha \geq 1$ be any constant. There exists a probabilistic polynomial-time algorithm A such that for all ℓ,*

$$\Pr_{z\sim\mathcal{D}_\ell,A}\left[\frac{\mathcal{D}_\ell(z)}{2} \leq A(1^\ell,z) \leq \mathcal{D}_\ell(z)\right] \geq 1 - \frac{1}{\ell^\alpha},$$

2.6 Characterizations through Conditional Coding

We will make use of recently discovered characterizations of the (worst-case and average-case) easiness of NP by the validity of conditional coding properties on probabilistic time-bounded Kolmogorov complexity [HIL+23].

Theorem 29 ([HIL+23])**.** *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*
2. **(Average-Case Conditional Coding)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$ and every polynomial q, there exists a polynomial p such that for all $n \in \mathbb{N}$,*

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

Theorem 30 ([HIL+23])**.** *The following are equivalent.*

1. $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$.
2. **(Independent Average-Case Conditional Coding)** *For every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ and every polynomial q, there exists a polynomial p such that for all $n \in \mathbb{N}$,*

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

Theorem 31 ([HIL+23])**.** *The following are equivalent.*

1. $\mathsf{NP} \subseteq \mathsf{BPP}$.
2. **(Worst-Case Conditional Coding)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and every polynomial q, there exists a polynomial p such that for all $n \in \mathbb{N}$ and all $(x, y) \in \mathsf{Support}(\mathcal{D}_n)$,*

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

2.7 Direct Product Generator

For $x, z \in \{0,1\}^n$, we let $x \cdot z := \sum_{i=1}^n x_i z_i \pmod 2$ denote their inner product modulo 2.

Definition 32 (Direct Product Generator [Hir21, Definiton 3.10])**.** For $k, n \in \mathbb{N}$, we define the *k-wise direct product generator* to be the function

$$\mathsf{DP}_k \colon \{0,1\}^n \times \{0,1\}^{nk} \to \{0,1\}^{nk+k}$$

such that

$$\mathsf{DP}_k(x; z^1, \ldots, z^k) := (z^1, \ldots, z^k, x \cdot z^1, \ldots, x \cdot z^k).$$

Lemma 33 (pK^t Reconstruction Lemma [GKLO22])**.** *For every $\varepsilon > 0$, $x \in \{0,1\}^n$, $s \in \mathbb{N}$, and $k \in \mathbb{N}$ satisfying $k \leq 2n$, let D be a randomized algorithm that takes an advice string β and runs in time t_D such that D ε-distinguishes $\mathsf{DP}_k(x;\mathcal{U}_{nk})$ from $\mathcal{U}_{nk+k}$. Then there is a polynomial p_{DP} such that*

$$\mathsf{pK}^{\widetilde{O}(t_D)\cdot p_{\mathsf{DP}}(n/\varepsilon)}(x \mid \beta) \leq k + \log p_{\mathsf{DP}}(nt_D/\varepsilon).$$

3 Characterizing Non-Existence of One-Way Functions by Average-Case Easiness of Conditional pK^t

In this section, we prove Theorem 1, which follows directly from Lemma 34 and Lemma 38, stated and proved in the following two subsections.

3.1 Computing Conditional pK^t from Inverting One-Way Functions

Lemma 34. (Item 1 $\implies$ Item 2 in Theorem 1). *If infinitely-often one-way functions do not exist, then for every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$, every polynomial q, and for all large enough constant c, there exists a probabilistic polynomial-time algorithm A such that for all $n, m, s \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}}[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x, y, 1^s)] \geq 1 - \frac{1}{q(n,m)},$$

where $\tau(n,m) := n^c \cdot m^c$.

The proof of Lemma 34 is inspired by a concurrent work [HKLO24], which extends a related result of [LP23] on computing K^t to that of conditional K^t, assuming the non-existence of one-way functions and an improved derandomization assumption.

We will need a few technical lemmas.

Lemma 35. *If infinitely-often one-way functions do not exist, then for every polynomial-time samplable distribution family $\{\mathcal{C}_{\langle n,m\rangle}\}$ supported over $\{0,1\}^m$, every polynomial q, and every constant $c > 1$, there exists a probabilistic polynomial-time algorithm A such that for all $n, m \in \mathbb{N}$, with probability at least $1 - 1/q(n,m)$ over $y \sim \mathcal{C}_{\langle n,m\rangle}$, the internal randomness of A, and $r \sim \{0,1\}^{\tau(n,m)}$,*

$$\sum_{x\in\{0,1\}^n} 2^{-\mathsf{K}^{\tau(n,m)}(x|y,r)} \cdot 1_{\left[A(x,y,r)=\mathsf{K}^{\tau(n,m)}(x|y,r)\right]} \leq \frac{1}{q(n,m)}, \tag{5}$$

where $\tau(n,m) := n^c \cdot m^c$.

Proof. Let $d > 0$ be a constant so that $\mathsf{K}^t(x) \leq n + d$ for every $x \in \{0,1\}^n$ and $t \geq O(n)$. Let $\tau := \tau(n,m) = n^c \cdot m^c$. Let S be the sampler for $\{\mathcal{C}_{\langle n,m\rangle}\}$ that uses $u := \mathsf{poly}(n,m)$ random bits.

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

> On input $(\ell, \Pi, r_{\mathsf{S}}, r)$, where $\ell \in \{0,1\}^{\log(n+d)}$, $\Pi \in \{0,1\}^{n+d}$, $r_{\mathsf{S}} \in \{0,1\}^u$, and $r \in \{0,1\}^\tau$, we first obtain $y := \mathsf{S}(r)$. We then run $U^{y,r}(\Pi_{[\ell]})$ for τ steps and obtain a string x. If x is of length n, we output (ℓ, x, y, r); otherwise output $(\ell, 0^n, y, r)$.

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

$$\mathbf{Pr}\left[A'(\ell, x, y, r; r_{_A}) \text{ succeeds}\right] \geq 1 - \frac{1}{q^2(n,m) \cdot n^b},$$

where r_A denotes the internal randomness of A, $b > 0$ is a constant specified later, (ℓ, x, y, r) is sampled according to f, and "$A'(\ell, x, y, r)$ succeeds" means $A'(\ell, x, y, r)$ outputs a pre-image of (ℓ, x, y, r).

By an averaging argument, we get that with probability at least $1 - 1/q(n,m)$ over $y \sim \mathcal{C}_{\langle n,m\rangle}$ (i.e., over $r_{\mathsf{S}} \sim \{0,1\}^u$), $r_{_A}$, and $r \sim \{0,1\}^\tau$, it holds that

$$\mathbf{Pr}\left[A'(\ell, x, y, r; r_{_A}) \text{ succeeds}\right] \geq 1 - \frac{1}{q(n,m) \cdot n^b}, \tag{6}$$

where the above probability is only over ℓ and x. In what follows, fix any *good* y, $r_{_A}$, and r such that Equation (6) holds.

By a union bound, Equation (6) yields that for all $\ell \in \{0,1\}^{\log(n+d)}$,

$$\mathbf{Pr}\big[A'(\ell, x, y, r) \text{ succeeds}\big] \geq 1 - \frac{n+d}{q(n,m) \cdot n^b}, \tag{7}$$

where now the probability is over x.

Next, for any fixed ℓ, consider the following distribution $\mathcal{D}_{(\ell,y,r)}$:

1. Pick $\Pi \sim \{0,1\}^{n+d}$.
2. Run $U^{y,r}(\Pi_{[\ell]})$ for τ steps and obtain a string x. If x is of length n, output (x); otherwise output (0^n).

Then Equation (7) implies that for all $\ell \in \{0,1\}^{\log(n+c)}$,

$$\mathop{\mathbf{Pr}}_{x \sim \mathcal{D}_{(\ell,y,r)}}\big[A'(\ell, x, y, r; r_{_A}) \text{ fails}\big] < \frac{n+d}{q(n,m) \cdot n^b}. \tag{8}$$

Now consider the following algorithm A:

> On input $(x, y, r; r_{_A})$, output the smallest $\ell \in [n+d]$ such that $A'(\ell, x, y, r, r_{_A})$ returns some $(\ell, \Pi, r_{\mathsf{S}}, r)$ for which $y = \mathsf{S}(r_{\mathsf{S}})$ and $U^{y,r}(\Pi_{[\ell]})$ outputs x within τ steps.

We will show that for all good y, $r_{_A}$ and r, the algorithm A satisfies the condition stated in Equation (5). For the sake of contradiction, suppose there exists some good y, $r_{_A}$ and r such that

$$\sum_{x \in \{0,1\}^n} 2^{-\mathsf{K}^{\tau(n,m)}(x|y,r)} \cdot 1_{\left[A(x,y) = \mathsf{K}^{\tau(n,m)}(x|y,r)\right]} \leq \frac{1}{q(n,m)}. \tag{9}$$

Note that for every fixed ℓ, y, r, and for every $x \in \{0,1\}^n$ with $\mathsf{K}^\tau(x \mid y, r) = \ell$, $\mathcal{D}_{(\ell,y,r)}$ outputs x with probability at least $2^{-\mathsf{K}^\tau(x|y,r)}$. In other words, for every such x, we have

$$2^{-\mathsf{K}^\tau(x|y,r)} \leq \mathcal{D}_{(\ell,y,r)}(x). \tag{10}$$

Also, for every $x \in \{0,1\}^n$ with $\mathsf{K}^t(x \mid y, r) = \ell$, if $A(x, y, r; r_{_A}) \neq \ell$, then it means $A'(\ell, x, y, r; r_{_A})$ fails. (To see this, consider the contrapositive.)

Then we have

$$\begin{aligned}
\frac{1}{q(n,m)} &\leq \sum_{\ell \in [n+d]} \sum_{\substack{x \in \{0,1\}^n: \\ \mathsf{K}^\tau(x|y,r) = \ell}} 2^{-\mathsf{K}^\tau(x|y,r)} \cdot 1_{[A(x,y,r;r_{_A}) \neq \mathsf{K}^\tau(x \mid y, r)]} && \text{(by Equation (9))} \\
&\leq \sum_{\ell} \sum_{x : \mathsf{K}^\tau(x|y,r) = \ell} \mathcal{D}_{(\ell,y,r)}(x) \cdot 1_{[A(x,y,r;r_{_A}) \neq \mathsf{K}^\tau(x \mid y, r)]} && \text{(by Equation (10))} \\
&\leq \sum_{\ell} \sum_{x : \mathsf{K}^\tau(x|y,r) = \ell} \mathcal{D}_{(\ell,y,r)}(x) \cdot 1_{[A'(\ell,x,y,r;r_{_A}) \text{ fails}]}.
\end{aligned}$$

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

$$\sum_{x : \mathsf{K}^\tau(x|y,r) = \ell} \mathcal{D}_{(\ell,y,r)}(x) \cdot 1_{[A'(\ell,x,y,r;r_{_A}) \text{ fails}]} \geq \frac{1}{(n+c) \cdot q(n,m)},$$

which contradicts Equation (8) by letting b be a sufficiently large constant. □

Lemma 36. *If infinitely-often one-way functions do not exist, then for every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}$ supported over $\{0,1\}^n \times \{0,1\}^m$, every polynomial q, there exists a polynomial p such that for all $n, m \in \mathbb{N}$, with probability at least $1 - 1/q(n,m)$ over $y \sim \mathcal{D}^{(2)}_{\langle n,m\rangle}$ and $r \sim \{0,1\}^{p(n,m)}$,*

$$\Pr_{x\sim\mathcal{D}_{\langle n,m\rangle}(\cdot|y)}\left[\mathsf{K}^{p(n,m)}(x \mid y, r) \leq \log \frac{1}{\mathcal{D}_{\langle n,m\rangle}(x \mid y)} + \log p(n,m)\right] \geq 1 - \frac{1}{q(n,m)},$$

where $\mathcal{D}^{(2)}_{\langle n,m\rangle}$ denotes the marginal distribution of $\mathcal{D}_{\langle n,m\rangle}$ on the second half.

Proof. Let $\{\mathcal{D}_{\langle n,m\rangle}\}$ be a family of polynomial-time samplable distribution, q be a polynomial. Fix any $n, m \in \mathbb{N}$ and let p be a polynomial specified later.

By Theorem 29, we get that there exists a polynomial p' such that

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}}\left[\mathsf{pK}^{p'(n,m)}(x \mid y) \leq \log \frac{1}{\mathcal{D}_{\langle n,m\rangle}(x \mid y)} + \log p'(n,m)\right] \geq 1 - \frac{1}{q(n,m)^4}. \tag{11}$$

Also, by Lemma 21, we get that for all $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$,

$$\mathsf{pK}^{p''(n,m)}_{1-1/q(n,m)^4}(x \mid y) \leq \mathsf{pK}^{p'(n,m)}(x \mid y) + O(\log q(n,m)), \tag{12}$$

where p'' is a sufficiently large polynomial.

Now note that by Proposition 17, we have that for all $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$,

$$\Pr_{r\sim\{0,1\}^{p(n,m)}}\left[\mathsf{K}^{p(n,m)}(x \mid y, r) \leq \mathsf{pK}^{p''(n,m)}_{1-1/q(n,m)^4}(x \mid y) + O(\log p''(n,m))\right] \geq 1 - \frac{1}{q(n,m)^4}, \tag{13}$$

provided that p is a sufficiently large polynomial.

By combining Equation (12) and Equation (13), we get that for *all* $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$,

$$\Pr_{r\sim\{0,1\}^{p(n,m)}}\left[\mathsf{K}^{p(n,m)}(x \mid y, r) \leq \mathsf{pK}^{p'(n,m)}(x \mid y) + O(\log q(n,m) + \log p''(n,m))\right] \geq 1 - \frac{1}{q(n,m)^4}. \tag{14}$$

By a union bound, Equation (11) and Equation (14) imply that

$$\Pr_{\substack{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}\\ r\sim\{0,1\}^{p(n,m)}}}\left[\mathsf{K}^{p(n,m)}(x \mid y, r) \leq \log \frac{1}{\mathcal{D}_{\langle n,m\rangle}(x \mid y)} + \log p'(n,m) + O(\log q(n,m) + \log p''(n,m))\right]$$
$$\geq 1 - \frac{1}{q(n,m)^2}.$$

Finally, by an averaging argument, the above yields that with probability at least $1 - 1/q(n,m)$ over $y \sim \mathcal{D}^{(2)}_{\langle n,m\rangle}$ and $r \sim \{0,1\}^{p(n,m)}$,

$$\Pr_{x\sim\mathcal{D}_{\langle n,m\rangle}(\cdot|y)}\left[\mathsf{K}^{p(n,m)}(x \mid y, r) \leq \log \frac{1}{\mathcal{D}_{\langle n,m\rangle}(x \mid y)} + \log p(n,m)\right] \geq 1 - \frac{1}{q(n,m)},$$

again, provided that p is a sufficiently large polynomial. This completes the proof of the lemma. □

Using Lemma 36 Lemma 35, we show the following.

Lemma 37. *If infinitely-often one-way functions do not exist, then for every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$, every polynomial q, and for all large enough constant c, there exists a probabilistic polynomial-time algorithm B such that for all $n, m, s \in \mathbb{N}$,*

$$\Pr_{\substack{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}\\ r\sim\{0,1\}^{\tau(n,m)}}}\left[B(x,y,r) = \mathsf{K}^{\tau(n,m)}(x \mid y,r)\right] \geq 1 - \frac{1}{q(n,m)}.$$

Proof. Let $\{\mathcal{D}_{\langle n,m\rangle}\}$ be a polynomial-time samplable distribution family. Let $\{\mathcal{D}^{(2)}_{\langle n,m\rangle}\}$ be the family of marginal distributions of $\{\mathcal{D}_{\langle n,m\rangle}\}$ on the second part. Note that $\{\mathcal{D}^{(2)}_{\langle n,m\rangle}\}$ is polynomial-time samplable and is supported over $\{0,1\}^m$. Let q be a polynomial. Let $c > 0$ be a sufficiently large constant to be specified later. Also, let $\tau(n,m) := n^c \cdot m^c$.

First of all, by Lemma 36, there exists some polynomial p such that for all $n, m \in \mathbb{N}$, with probability at least $1 - 1/q(n,m)^2$ over $y \sim \mathcal{D}^{(2)}_{\langle n,m\rangle}$ and $r' \sim \{0,1\}^{p(n,m)}$,

$$\Pr_{x\sim\mathcal{D}_{\langle n,m\rangle}(\cdot|y)}\left[\mathsf{K}^{p(n,m)}(x \mid y,r') \leq \log\frac{1}{\mathcal{D}_{\langle n,m\rangle}(x \mid y)} + \log p(n,m)\right] \geq 1 - \frac{1}{q(n,m)^2}.$$

Note that by letting c be a sufficiently large constant (in particular, $\tau > p$), the above implies that with probability at least $1 - 1/q(n,m)^2$ over $y \sim \mathcal{D}^{(2)}_{\langle n,m\rangle}$ and $r \sim \{0,1\}^{\tau(n,m)}$,

$$\Pr_{x\sim\mathcal{D}_{\langle n,m\rangle}(\cdot|y)}\left[\mathsf{K}^{\tau(n,m)}(x \mid y,r) \leq \log\frac{1}{\mathcal{D}_{\langle n,m\rangle}(x \mid y)} + O(\log p(n,m))\right] \geq 1 - \frac{1}{q(n,m)^2}. \tag{15}$$

Also, by Lemma 35, there exists a polynomial-time algorithm B such that for all $n, m \in \mathbb{N}$, with probability at least

$$1 - \frac{1}{q(n,m)^2 \cdot p(n,m)^b} \geq 1 - \frac{1}{q(n,m)^2}$$

over $y \sim \mathcal{C}_{\langle n,m\rangle}$, $r \sim \{0,1\}^{\tau(n,m)}$ and $r_{_B} \sim \{0,1\}^{\mathsf{poly}(n,m)}$,

$$\sum_{x\in\{0,1\}^n} 2^{-\mathsf{K}^{\tau(n,m)}(x|y,r)} \cdot 1_{\left[B(x,y,r;r_{_B})=\mathsf{K}^{\tau(n,m)}(x|y,r)\right]} \leq \frac{1}{q(n,m)^2 \cdot p(n,m)^b}, \tag{16}$$

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

Consider any *good* y, r and $r_{_B}$ such that both Equation (15) and Equation (16) hold. (Note that y, r and $r_{_B}$ are good with probability at least $1 - 2/q(n,m)^2$.) We claim that

$$\Pr_{x\sim\mathcal{D}_{\langle n,m\rangle}(\cdot|y)}\left[B(x,y,r;r_{_B}) = \mathsf{K}^{\tau(n,m)}(x \mid y,r)\right] \geq 1 - \frac{1}{2q(n,m)}. \tag{17}$$

Suppose, for the sake of contradiction, Equation (17) is not true. Then for some good y, r and $r_{_B}$, we have

$$\Pr_{x\sim\mathcal{D}_{\langle n,m\rangle}(\cdot|y)}\left[B(x,y,r;r_{_B}) \neq \mathsf{K}^{\tau(n,m)}(x \mid y,r)\right] > \frac{1}{2q(n,m)}. \tag{18}$$

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

- $B(x,y,r;r_{_B}) \neq \mathsf{K}^{\tau(n,m)}(x \mid y,r)$

- $\mathsf{K}^{\tau(n,m)}(x \mid y, r) \leq \log \frac{1}{\mathcal{D}_{\langle n,m\rangle}(x|y)} + O(\log p(n,m))$.

Note that by Equation (18) and Equation (15), we get that

$$\sum_{x\in\{0,1\}^n} \mathcal{D}_{\langle n,m\rangle}(x \mid y) \cdot 1_{\mathcal{E}(x)} \geq 1 - \left(1 - \frac{1}{2q(n,m)}\right) - \frac{1}{q(n,m)^2} \geq \frac{1}{4q(n,m)}. \tag{19}$$

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

$$\mathcal{D}_{\langle n,m\rangle}(x \mid y) \leq \frac{p(n,m)^{O(1)}}{2^{\mathsf{K}^{\tau(n,m)}(x|y,r)}}. \tag{20}$$

Now we have

$$\begin{aligned}
\frac{1}{4q(n,m)} &\leq \sum_{x\in\{0,1\}^n} \mathcal{D}_{\langle n,m\rangle}(x \mid y) \cdot 1_{\mathcal{E}(x)} && \text{(by Equation (19))}\\
&\leq \sum_{x\in\{0,1\}^n} \frac{\tau(n,m)}{2^{\mathsf{K}^{p(n,m)^{O(1)}}(x|y,r)}} \cdot 1_{\mathcal{E}(x)} && \text{(by Equation (20))}\\
&\leq p(n,m)^{O(1)} \cdot \sum_{x\in\{0,1\}^n} 2^{-\mathsf{K}^{\tau(n,m)}(x|y,r)} \cdot 1_{\mathcal{E}(x)}\\
&\leq p(n,m)^{O(1)} \cdot \sum_{x\in\{0,1\}^n} 2^{-\mathsf{K}^{\tau(n,m)}(x|y,r)} \cdot 1_{[B(x,y,r;r_{_B})\neq \mathsf{K}^{\tau(n,m)}(x|y,r)]}.
\end{aligned}$$

By rearranging, we get

$$\sum_{x\in\{0,1\}^n} 2^{-\mathsf{K}^{\tau(n,m)}(x|y,r)} \cdot 1_{[B(x,y,r;r_{_B})\neq \mathsf{K}^{\tau(n,m)}(x|y,r)]} \geq \frac{2}{p(n,m)^{O(1)} \cdot 4 \cdot q(n,m)}.$$

However, this contradicts Equation (16) by letting b be a sufficiently large constant.

Now we have concluded Equation (17). Note that this means, with probability at least $1 - 2/q(n,m)^2$ over $y \sim \mathcal{C}_{\langle n,m\rangle}$, $r \sim \{0,1\}^{\tau(n,m)}$ and $r_{_B} \sim \{0,1\}^{\mathsf{poly}(n,m)}$,

$$\mathop{\mathbf{Pr}}_{x\sim\mathcal{D}_{\langle n,m\rangle}(\cdot|y)}\left[B(x,y,r;r_{_B}) = \mathsf{K}^{\tau(n,m)}(x \mid y, r)\right] \geq 1 - \frac{1}{2q(n,m)}.$$

By a union bound, we get

$$\mathop{\mathbf{Pr}}_{\substack{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}\\ r\sim\{0,1\}^{\tau(n,m)}\\ r_{_B}}}\left[B(x,y,r;r_{_B}) = \mathsf{K}^{\tau(n,m)}(x \mid y, r)\right] \geq 1 - \frac{2}{q(n,m)^2} - \frac{1}{2q(n,m)} \geq 1 - \frac{1}{q(n,m)}.$$

This completes the proof of Lemma 37. □

We are now ready to show Lemma 34.

Proof of Lemma 34. Let $\{\mathcal{D}_{\langle n,m\rangle}\}$ be a polynomial-time samplable distribution family. Let $c > 0$ be any sufficiently large constant, and let $\tau(n,m) := n^c \cdot m^c$.

By Lemma 37, there exists a probabilistic polynomial-time algorithm B such that for all $n, m, s \in \mathbb{N}$,

$$\mathop{\mathbf{Pr}}_{\substack{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}\\ r\sim\{0,1\}^{\tau(n,m)}}}\left[B(x,y,r) = \mathsf{K}^{\tau(n,m)}(x \mid y, r)\right] \geq 1 - \frac{1}{q(n,m)^2}.$$

By an averaging argument, we get that with probability at least $1 - 1/(2q(n,m))$ over $(x,y) \sim \mathcal{D}_{\langle n,m\rangle}$,

$$\Pr_{r\sim\{0,1\}^{\tau(n,m)},B}\left[B(x,y,r) = \mathsf{K}^{\tau(n,m)}(x \mid y,r)\right] \geq 1 - \frac{2}{q(n,m)}. \tag{21}$$

Consider the following algorithm A.

On input $(x,y,1^s)$, A picks $r \sim \{0,1\}^{\tau(n,m)}$ and accepts if and only if $B(x,y,r) \leq s$.

We now argue the correctness of A. Consider any *good* (x,y) such that Equation (21) holds. If $(x,y,1^s)$ is a YES instance of $\mathsf{Cond\text{-}pK}[\tau]$, i.e., $\mathsf{pK}^{\tau(n,m)}_{2/3}(x \mid y) \leq s$, which means

$$\Pr_{r\sim\{0,1\}^{\tau(n,m)}}\left[\mathsf{K}^{\tau(n,m)}(x \mid y,r) \leq s\right] \geq \frac{2}{3}. \tag{22}$$

By combining Equation (22) and Equation (21), we get that

$$\Pr_{r\sim\{0,1\}^{\tau(n,m)},B}[B(x,y,r) \leq s] \geq 1 - \frac{2}{q(n,m)} - \frac{1}{3} \geq \frac{3}{5}. \tag{23}$$

In this case , $A(x,y,1^s) = 1$ with probability at least $3/5$.

Similarly, if $(x,y,1^s)$ is a NO instance of $\mathsf{Cond\text{-}pK}[\tau]$, i.e., $\mathsf{pK}^{\tau(n,m)}_{1/3}(x \mid y) > s$, which means

$$\Pr_{r\sim\{0,1\}^{\tau(n,m)}}\left[\mathsf{K}^{\tau(n,m)}(x \mid y,r) > s\right] \geq \frac{2}{3}. \tag{24}$$

By combining Equation (21) and Equation (24), we get that

$$\Pr_{r\sim\{0,1\}^{\tau(n,m)},B}[B(x,y,r) > s] \geq \frac{3}{5}. \tag{25}$$

In this case, $A(x,y,1^s) = 0$ with probability at least $3/5$.

The above implies that for every s, with probability at least $1-1/(2q(n,m))$ over $(x,y) \sim \mathcal{D}_{\langle n,m\rangle}$, A decides $\mathsf{Cond\text{-}pK}[\tau]$ on $(x,y,1^s)$. The lemma now follows by amplifying the success probability of A using standard amplification techniques. □

3.2 Inverting One-Way Functions from Computing Conditional pK^t

Lemma 38. (Item 2 $\implies$ Item 1 in Theorem 1). *Suppose for every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$, every polynomial q, and for all large enough constant c, there exists a probabilistic polynomial-time algorithm A such that for all $n,m,s \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}}[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x,y,1^s)] \geq 1 - \frac{1}{q(n,m)},$$

where $\tau(n,m) := n^c \cdot m^c$. Then infinitely-often one-way functions do not exist.

Proof Sketch. The proof is similar to that of Lemma 68, which uses ideas from that of Theorem 6, as described in Section 1.2. If we have an efficient algorithm for computing (conditional) $\mathsf{pK}^{\mathsf{poly}}$ on average as specified in the assumption of the lemma, then we can construct a function that distinguishes the output distribution of a cryptographic pseudorandom generator from the uniform distribution, since the outputs of such a generator have low $\mathsf{pK}^{\mathsf{poly}}$ complexity while a random string has high $\mathsf{pK}^{\mathsf{poly}}$ complexity. This then implies that infinitely-often one-way functions do not exist. □

3.3 Equivalences between Average-Case Easiness of Approximating and Computing (Conditional) pK^t

Theorem 5 (Informal)**.** *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*
2. *Approximating* pK^t *is easy-on-average over samplable distributions.*
3. *Approximating conditional* pK^t *is easy-on-average over samplable distributions.*
4. *Computing* pK^t *is easy-on-average over samplable distributions.*
5. *Computing conditional* pK^t *is easy-on-average over samplable distributions.*

Proof Sketch. It can be shown that each of last 4 items is equivalent to the non-existence of infinitely-often one-way functions.

A result of [IRS22] characterized the non-existence of (infinitely-often) one-way functions by the tractability of approximating Kolmogorov complexity over polynomial-time samplable distributions. (See also Section 1.2 for an exposition of the proof in [IRS22].)

By employing ideas in its proof, one can easily show that the non-existence of infinitely-often one-way functions is equivalent to both the first and second items.

More specifically, the proof in [IRS22] uses the coding theorem for time-unbounded Kolmogorov complexity. To characterize the non-existence of one-way functions by the first item, one can instead use the efficient coding theorem for pK^t obtained by [LOZ22]. For the second item, one can also use an average-case conditional coding theorem for pK^t (Theorem 29), which can be obtained assuming the non-existence of infinitely-often one-way functions.

The equivalence between the fourth item and the non-existence of infinitely-often one-way functions was shown in [LP23].

For the last item, the equivalence is given by Theorem 1. □

4 Characterizing $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ by Average-Case Easiness of Conditional pK^t in Sublinear-Time Regime

In this subsection, we prove Theorem 2. We first review the notion of secret sharing scheme and some technical tools.

4.1 Technical Tools

Definition 39 (Access Structure. See also [Hir22, Definition 6.7])**.** An access structure $\mathcal{A} \subseteq 2^{[n]}$ is a monotone subset. i.e., for every $S \in \mathcal{A}$, if $T \supseteq S$, we have $T \in \mathcal{A}$. The *minimum weight* of an access structure $\mathcal{A}$ is defined as $w(\mathcal{A}) := \min\{|T| \mid T \in \mathcal{A}\}$.

Definition 40 (Secret Sharing Scheme. See also [Hir22, Definitions 6.8 and 6.10])**.** For a family of access structure $\{\mathcal{A}_\psi\}_{\psi\in\{0,1\}^*}$, where each $\mathcal{A}_\psi \subseteq 2^{[n]}$, a *secret sharing scheme* for $\{\mathcal{A}_\psi\}$ is a pair $(\mathsf{Share}, \mathsf{Rec})$ of a polynomial-time randomized algorithm Share and a deterministic polynomial-time algorithm Rec with the following properties. For every ψ and every $\ell \in \mathbb{N}$,

1. **(Correctness.)** For every $T \in \mathcal{A}_\psi$ and every string $x \in \{0,1\}^\ell$, $\mathsf{Share}(\psi, x)$ outputs a sequence $(y_1, \cdots, y_n)$ of n strings and $\mathsf{Rec}(\psi, y_T) = x$, where $y_t := \{(i, y_i) \mid y \in T\}$.

2. **(Privacy.)** For every $T \in \mathcal{A}_\psi$ and random variable X on $\{0,1\}^\ell$, the random variable X and $\mathsf{Share}(X)_T$ are statistically independent.

Lemma 41 ([Hir22, Lemma 6.9])**.** *Let $\{\mathcal{A}_\psi\}_{\psi\in\{0,1\}^*}$ be a family of access structure, where each $\mathcal{A}_\psi \subseteq 2^{[n]}$. Let $(\mathsf{Share}, \mathsf{Rec})$ be a secret sharing scheme for $\{\mathcal{A}_\psi\}$. Then, for every ψ, $\ell, k \in \mathbb{N}$, and $z \in \{0,1\}^*$ it holds that*

$$\Pr_{X\sim\{0,1\}^\ell,\mathsf{Share}}\left[\min_{T\notin\mathcal{A}_\psi} \mathsf{K}(X \mid \mathsf{Share}(\psi, X)_T, z) \geq \ell - n - k\right] \geq 1 - 2^{-k}.$$

For $T \in [n]$, we identify T by its characteristic vector, which is the string $x \in \{0,1\}^n$ such that $x_i = 1$ iff $i \in T$ for all $i \in [n]$.

Lemma 42 ([BL88])**.** *Let $\{\mathcal{A}_\psi\}_{\psi\in\{0,1\}^*}$ be a family of access structure, where each ψ is a monotone formula on n variables and $\mathcal{A}_\psi := \{T \subseteq [n] \mid \psi(T) = 1\}$. Then $\{\mathcal{A}_\psi\}$ admits a secret sharing scheme $(\mathsf{Share}, \mathsf{Rec})$, where both Share and Rec have polynomial running time.*

Definition 43 (Minimum Monotone Satisfying Assignment)**.** For a monotone formula ψ on n variables, the *minimum monotone satisfying assignment*, denoted by $\mathsf{MMSA}(\psi)$, is the minimum hamming weight of an assignment $\alpha \in \{0,1\}^n$ such that $\psi(\alpha) = 1$.

Note that for the family of access structure $\{\mathcal{A}_\psi\}$ defined in Lemma 41, we have $\mathsf{MMSA}(\psi) = w(\mathcal{A}_\psi)$ for every ψ.

Lemma 44 (NP-Hardness of GapMMSA [DS04, DHK15]. See also [Hir22, Lemma 6.13])**.** *For some function $g(n) := n^{1/(\log\log n)^{O(1)}}$, it is NP-hard to solved the following promise problem $\mathsf{Gap}_g\mathsf{MMSA}{=}(\mathsf{YES}, \mathsf{NO})$:*

$$\mathsf{YES} := \{(\psi, \zeta) \mid \mathrm{MMSA}(\psi) \leq \zeta\},$$
$$\mathsf{NO} := \{(\psi, \zeta) \mid \mathrm{MMSA}(\psi) > g(|\psi|) \cdot \zeta\}.$$

In other words, for the family of access structure $\{\mathcal{A}_\psi\}$ defined in Lemma 41, it is NP-hard to approximate $w(\mathcal{A}_\psi)$, the minimum weight of $\mathcal{A}_\psi$.

4.2 NP-Hardness of Computing Conditional pK^t in Sublinear-Time Regime

In this subsection, we prove the following lemma which says that one can reduce the problem of GapMMSA to that of computing conditional pK^t in the "sublinear-time" regime.

Lemma 45. *For any constant $\kappa \geq 1$, there exists a randomized polynomial-time algorithm R such that the following holds for every monotone formula ψ on n variables, $\zeta \in [n]$, and $m \in \mathbb{N}$.*

1. *$R(\psi, \zeta, 1^m)$ outputs $(x, y, 1^\rho)$, where $|x| = |\psi|^{15\kappa}$, $|y| = m$, and $\rho \leq O(|x|)$.*
2. *For every constant $c > 1$, there exists a constant $c' > c$ such that if $m = |\psi|^{c'\kappa}$ and ψ is sufficiently large, then the following hold for $\tau(a, b) := a^c \cdot b^{1-1/c}$, and $g(z) := z^{1/(\log\log z)^{O(1)}}$ is the function in Lemma 44.*
   - **(Completeness.)** *If $\mathrm{MMSA}(\psi) \leq \zeta$, then with probability 1 over $(x, y, 1^\rho) \sim R(\psi, \zeta, 1^m)$, $\mathsf{pK}^{\tau(|x|,|y|)}_{2/3}(x \mid y) \leq \rho$.*
   - **(Soundness.)** *If $\mathrm{MMSA}(\psi) > g(|\psi|) \cdot \zeta$, then with probability at least $1 - |\psi|^{-\kappa}$ over $(x, y, 1^\rho) \sim R(\psi, \zeta, 1^m)$, $\mathsf{pK}^{\tau(|x|,|y|)}_{1/3}(x \mid y) > \rho$.*

The rest of this subsection is devoted to prove Lemma 45. The proof is an adaptation of that of [Hir22, Theorem 6.6], which showed the NP-hardness of computing ("sublinear-time") conditional K^t. However, it requires some crucial modifications both to address the more complex notion of pK^t and to be used to showing Theorem 2. We present the detail for completeness.

We assume without loss of generality that $n \leq |\psi|$ Let $\kappa > 1$ be any constant and let $\tau(n, m) := a^c \cdot b^{1-1/c}$. We first describe the algorithm R.

The Algorithm R. Fix a monotone formula ψ on n variables, $\zeta \in [n]$ and $m \in \mathbb{N}$.

The procedure involves $D := |\psi|^{5\kappa}$ iterations. At the i-th iteration, where $i \in [D]$, the following is performed.

1. Pick $x^i \sim \{0,1\}^\ell$, where $\ell := |\psi|^{10\kappa}$.
2. Compute $(s^i_1, \ldots, s^i_n) := \mathsf{Share}(\psi, x)$, where $s^i_j \in \{0,1\}^h$ for each $j \in [n]$ and $h = \mathsf{poly}(|\psi|)$.
3. Find λ, if exists, such that $m = 2^\lambda \cdot h \cdot D$. Also, let $m' := 2^\lambda \cdot h$.
4. Pick $k^i_1, \ldots, k^i_n \sim \{0,1\}^\lambda$.
5. Let $y^i \in \{0,1\}^{m'}$ be viewed as a function $y^i \colon \{0,1\}^\lambda \to \{0,1\}^h$ and define $y^i(q) = s_i$ if $q = k^i_j$ and $y^i(q) = 0^h$ otherwise.
6. Let $k^i := (k^i_1, \ldots, k^i_n)$ and $s^i := (s^i_1, \ldots, s^i_n)$.

After we finish D iterations and obtain $(x^1, y^1, k^1, s^1), \ldots, (x^D, y^D, k^D, s^D)$, let $x := (x^1, \ldots, x^D)$, $y := (y^1, \ldots, y^D)$. The output of $R(\psi, \zeta, 1^m)$ is $(x, y, 1^\rho)$, where $\rho := 2\lambda D\zeta$.

It is easy to verify the the first item in Lemma 45. Next, we show the second item by a sequence of lemma.

Let $c > 1$ be any constant, and let $m = 2^\lambda \cdot h \cdot D$ for some $\lambda = E \cdot c^2 \cdot \kappa \cdot \log |\psi|$, where $E > 0$ is a constant specified later.

Let $k := (k^d_i \mid d \in [D], i \in [n])$ and $s := (s^d_i \mid d \in [D], i \in [n])$.

We first need to make sure that the string y output by $R(\psi, \zeta, 1^m)$ is well-defined in the sense that k is pair-wise distinct with high probability.

Lemma 46. *With probability at least $1 - |\psi|^{-2\kappa}$, k is pair-wise distinct in which case y is well defined.*

Proof. For any $d \in [D]$, any $i, j \in [n]$ with $i \neq j$, the probability that $k^d_i = k^d_j$ is at most $2^{-\kappa}$. By a union bound over all $d \in [d]$ and $i, j \in [n]$ with $i \neq j$, the probability that k is not pair-wise distinct is at most $D \cdot n^2 \cdot 2^{-\lambda} \leq |\psi|^{2\kappa}$, by letting the constant E be sufficiently large. □

In what follows, we will implicitly assume that Lemma 46 holds. We proceed to show the completeness and soundness of the algorithm R.

Completeness. The completeness of the algorithm R is given by the following lemma.

Lemma 47. *If $w(A_\psi) \leq \zeta$, then $\mathsf{pK}^t(x \mid y) \leq \rho$, with probability 1 over $(x, y, 1^\rho) \sim R(\psi, \zeta, 1^m)$.*

Proof. The proof is essentially the same as that of [Hir22, Claim 6.19].

Let $T \in A_\psi$ be a minimum authorized set of parties. Note that $|T| = w(A_\psi)$. Consider the following program M:

M^y first takes T and $\{k_i^d \mid i \in T, d \in [D]\}$, and then obtain s_T^d for every $d \in [D]$ by queering y. Then it takes ψ and runs $\mathsf{Rec}(\psi, s_T^d)$ to obtain x^d for every $d \in [D]$.

This machine can be described using

$$(T \cdot \log n) + (D \cdot |T| \cdot \lambda + |\psi|) + O(\log D) \leq 2\lambda D\zeta = \rho.$$

The above implies that

$$\mathsf{pK}^t(x \mid y) \leq \mathsf{K}^t(x \mid y) \leq \rho,$$

where $t := |\psi|^{O(\kappa)} \cdot \log(m)$. Note that we have

$$\begin{aligned}\tau(|x|, |y|) &= |x|^c \cdot |y|^{1-1/c} \\ &\geq m^{1-1/c} \\ &\geq |\psi|^{E \cdot c \cdot \kappa \cdot (1-1/c)} \\ &\geq t,\end{aligned}$$

where the last inequality holds since $c > 1$ and by letting E be a sufficiently large constant. Therefore, we get that $\mathsf{pK}^{\tau(|x|,|y|)} \leq \rho$, as desired. □

Soundness. We now show the soundness of R, by showing the following.

Lemma 48. *If $w(A_\psi) \geq g(|\psi|) \cdot \zeta$, where $g(z) := z^{1/(\log\log z)^{O(1)}}$ is the function in Lemma 44, then*

$$\mathsf{pK}_{1/3}^{\tau(|x|,|y|)}(x \mid y) > \rho,$$

with probability at least $1 - 2 \cdot |\psi|^{-2\kappa}$ over $(x, y, 1^\rho) \sim R(\psi, \zeta, 1^m)$.

Proof. Let $\tau := \tau(|x|, |y|)$.

We first state the condition under which the statement of Lemma 48 holds. This will be given by the following two claims.

Claim 49. *With probability at least $1 - |\psi|^{-2\kappa}$ over the internal randomness of R, we have*

$$\Pr_{w \sim \{0,1\}^\tau}[\mathsf{K}(k \mid s, w) \geq nD\lambda - 4\kappa \log |\psi|] \geq 1 - o(1). \tag{26}$$

Proof of Claim 49. Note that in the algorithm R, k is chosen uniform from $\{0,1\}^{nD\lambda}$, independent of s. Then for every fixed $w \in \{0,1\}^\tau$ and s sampled by R, we have

$$\Pr_k[\mathsf{K}(k \mid w, s) \geq nD\lambda - 4\kappa \log |\psi|] \geq 1 - \frac{1}{|\psi|^{4\kappa}}.$$

This implies

$$\Pr_{w \sim \{0,1\}^\tau, s, k}[\mathsf{K}(k \mid w, s) \geq nD\lambda - 4\kappa \log |\psi|] \geq 1 - \frac{1}{|\psi|^{4\kappa}}.$$

By averaging, we have that with probability at least $1 - |\psi|^{-2\kappa}$ over k and s sampled by the algorithm R, it holds that

$$\Pr_{w \sim \{0,1\}^\tau}[\mathsf{K}(k \mid s, w) \geq nD\lambda - 4\kappa \log |\psi|] \geq 1 - |\psi|^{-2\kappa},$$

as desired. ◇

For $T \subseteq [n]$ and $V \subseteq [D]$, we define $s_T^V := \left\{ s_i^{d'} \mid i \in T, d' \in V \right\}$.

Claim 50. *With probability at least $1 - |\psi|^{-2\kappa}$ over the internal randomness of R, it holds that*

$$\Pr_{w \sim \{0,1\}^\tau} \left[\forall\ T \notin A_\psi, d \in [D], \mathsf{K}\left(x^d \mid s_T^d, s_{[n]}^{[D]\setminus\{d\}}, k, w \right) \geq \ell - n - 3\log D \right] \geq 1 - o(1). \tag{27}$$

Proof of Claim 50. The proof follows closely to that of [Hir22, Claim 6.21].

Fix $d \in [D]$. Note that with respect tot the algorithm R, the random variable x^d is independent of $s_{[n]}^{[D]\setminus\{d\}}$ and k. Also, $s^d = \mathsf{Share}(\psi, x^d)$. Then by Lemma 41, we get that for every fixed $s_{[n]}^{[D]\setminus\{d\}}$, k sampled by R and $w \in \{0,1\}^\tau$,

$$\Pr_{x^d, s_T^d} \left[\forall\ T \notin A_\psi, \mathsf{K}\left(x^d \mid s_T^d, s_{[n]}^{[D]\setminus\{d\}}, k, w \right) \geq \ell - n - 3\log D \right] \geq 1 - \frac{1}{D^3}.$$

By union bounding all $d \in [D]$, the above implies that

$$\Pr_{x^d, s_T^d, s_{[n]}^{[D]\setminus\{d\}}, k, w \sim \{0,1\}^\tau} \left[\forall\ T \notin A_\psi, d \in [D], \mathsf{K}\left(x^d \mid s_T^d, s_{[n]}^{[D]\setminus\{d\}}, k, w \right) \geq \ell - n - 3\log D \right] \geq 1 - \frac{1}{D^2}$$

By averaging, we have that with probability at least $1 - D$ over $x^d, s_T^d, s_{[n]}^{[D]\setminus\{d\}}, k$ sampled by the algorithm R, it holds that

$$\Pr_{w \sim \{0,1\}^\tau} \left[\forall\ T \notin A_\psi, d \in [D], \mathsf{K}\left(x^d \mid s_T^d, s_{[n]}^{[D]\setminus\{d\}}, k, w \right) \geq \ell - n - 3\log D \right] \geq 1 - \frac{1}{D},$$

as desired. ◇

Let us assume that both Equation (26) and Equation (27) hold, which happens with probability at least $1 - 2 \cdot |\psi|^{-2\kappa}$.

Suppose $w(A_\psi) \geq g(|\psi|) \cdot \zeta$, and for the sake of contradiction, suppose $\mathsf{pK}^\tau_{1/3}(x \mid y) < \rho$. This means with probability at least $1/3$ over $w \sim \{0,1\}^\tau$, there is a machine M_0 of size at most ρ such that $M_0^y(w)$ runs in time τ and outputs x. Let $w \in \{0,1\}^\tau$ be such that all of the following hold.

- There exists a machine M of size at most $\rho + O(\log \ell)$ such that for every $d \in [D]$, $M^y(d; w)$ runs in time 2τ and outputs x^d. Note that such a machine can be obtained from M_0.
- It holds that
$$\mathsf{K}(k \mid s, w) \geq nD\lambda - 4\kappa \log |\psi|. \tag{28}$$
- For every $T \notin A_\psi$ and every $d \in [D]$,
$$\mathsf{K}\left(x^d \mid s_T^d, s_{[n]}^{[D]\setminus\{d\}}, k, w \right) \geq \ell - n - 3\log D. \tag{29}$$

Note that such a w exists since both Equation (26) and Equation (27) hold.

For $d \in [D]$, let $T(d)$ be the set of $i \in [n]$ such that $M^y(d; w)$ queries $y^d(k_i^d)$ during its computation. We claim the following.

Claim 51. *There exists some $d \in [D]$ such that $T(d) \notin A_\psi$.*

Proof of Claim 51. Let $\alpha := \sum_{d\in[D]} T(d)$. We first observe that

$$\mathsf{K}^w(k \mid s) \le |M| + (nD - \alpha)\cdot\lambda + \alpha(\log 2\tau + \log nD) + O(\log nD). \tag{30}$$

To show this, the idea is to learn k by simulating the program M. More specifically, for every d and $i \in T(d)$, there exists a time step $t_{i,d} \in [2\tau]$ such that $M^y(-;w)$ makes a query k_i^d to y on input d. Then to recover k given s, we will first explicitly describe those k_i^d that $M^y(-;w)$ would not query on any input d, which is the set

$$Q := \{k_i^d \in \{0,1\}^\lambda \mid (i,d) \in [n]\times[D], i \notin T(d)\}.$$

Note that this set can be described using a string of length at most

$$(nD - \alpha)\cdot\lambda + O(\log nD).$$

Next, for every d, we keep track of the all the time steps $t_{i,d}$ at which $M^y(-;w)$ queries $y^d(k_i^d)$ on input d, which yields

$$R := \{(i,d,t_{i,d}) \in [n]\times[D]\times[2\tau] \mid i \in T(d)\}.$$

Note that this set can be encoded using a string of length at most

$$\alpha\cdot(\log 2\tau + \log nD) + O(\log nD).$$

Then given s, we can recover k as follows. For every $d \in [D]$, we simulate $M^y(d;w)$. At some time step $t_{i,d}$ such that $(i,d,t_{i,d}) \in R$, $M^y(d;w)$ makes a query k_i^d, we retrieve this query and answer it with s_i^d. By continuing this simulation for 2τ steps, and we will eventually learn all the k_i^d that are not in Q. This concludes Equation (30).

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

$$nD\lambda - 4\kappa\log|\psi| \le |M| + (nD-\alpha)\cdot\lambda + \alpha\cdot(\log 2\tau + \log nD) + O(\log nD).$$

By rearranging and simplifying, we get

$$(\lambda - \log\tau - \log nD - 1)\cdot\alpha \le |M| + O(\log nD) \tag{31}$$

Fist of all, note that

$$\begin{aligned}
\tau &= |x|^c\cdot|y|^{1-1/c}\\
&= |\psi|^{15\cdot c\cdot\kappa}\cdot m^{1-1/c}\\
&= |\psi|^{15\cdot c\cdot\kappa}\cdot(2^\lambda\cdot h\cdot D)^{1-1/c}\\
&\le 2^{(1-1/c)\cdot\lambda}\cdot|\psi|^{O(c\cdot\kappa)}.
\end{aligned}$$

Consider the left hand side of Equation (31), we have

$$\begin{aligned}
\lambda - \log\tau - \log nD - 1 &= \lambda - ((1-1/c)\cdot\lambda + O(c\cdot\kappa\cdot\log|\psi|)) - \log nD - 1\\
&\ge \lambda/c - O(c\cdot\kappa\cdot\log|\psi|)\\
&\ge \lambda/(2c),
\end{aligned} \tag{32}$$

where the last inequality holds if we choose E to be a sufficiently large constant as $\lambda = E\cdot c^2\cdot\kappa\log|\psi|$.

Consider the right hand side of Equation (31), we have

$$\begin{aligned} |M| + O(\log nD) &= \rho + O(\log \ell) + O(\kappa \cdot \log |\psi|) \\ &\leq 2\lambda D\zeta + O(\kappa \cdot \log |\psi|) \\ &\leq 3\lambda D\zeta. \end{aligned} \tag{33}$$

Substitute Equation (32) and Equation (33) to Equation (31), we get

$$\begin{aligned} \alpha &\leq \frac{2c}{\lambda} \cdot 3\lambda D\zeta \\ &\leq 6c \cdot D \cdot \zeta \\ &\leq 6c \cdot D \cdot \frac{w(A_\psi)}{g(|\psi|)} \\ &< D \cdot w(A_\psi). \end{aligned}$$

Where the last inequality holds if ψ is sufficiently large (go that $g(|\psi| > 6c)$).

By averaging, this means that there exists some $d \in [D]$ such that $|T(d)| \leq w(A_\psi)$ and hence for such d, we have $T(d) \notin A_\psi$, as desired. ◇

Now by Claim 51 and Equation (29), we get that for some $d \in [D]$ and $T(d) \notin A_\psi$,

$$\mathsf{K}\left(x^d \mid s^d_{T(d)}, s^{[D]\setminus\{d\}}_{[n]}, k, w\right) \geq \ell - n - 3\log D. \tag{34}$$

Finally, we show the following claim which will give a contradiction.

Claim 52. *For every $d \in [D]$, it holds that*

$$\mathsf{K}\left(x^d \mid s^d_{T(d)}, s^{[D]\setminus\{d\}}_{[n]}, k, w\right) \leq |M| + O(\log D). \tag{35}$$

Proof of Claim 52. The proof follows closely to that of [Hir22, Claim 6.23].

Note that $M^y(d; w)$ outputs x^d. Also, $M^y(d; w)$ does not make any query in $\{k^d_i \mid i \notin T(d)\}$. In other words, $M^y(d; w)$ only makes queries in $\{k^d_i \mid i \in T(d)\}$ and in $\{k^{d'}_i \mid d' \in [D]\setminus\{d\}, i \in [n]\}$. For the former, the answers can be obtained from $s^d_{T(d)}$ and k, and for the latter, the answers can be obtained from $s^{[D]\setminus\{d\}}_{[n]}$ and k. Therefore, given $s^d_{T(d)}$, $s^{[D]\setminus\{d\}}_{[n]}$, k, w, d and M, we can simulate $M^y(d; w)$ and recover x^d. ◇

Note that from Equation (33), we have $|M| + O(\log D) \leq 4\lambda D\zeta \leq 4\lambda Dn$. Then By combining Equation (34) and Equation (35), we get

$$\ell - n - 3\log D \leq 4\lambda Dn.$$

However, this gives a contradiction by our setting of ℓ, λ and D (provided that ψ is sufficiently large). This completes the proof of Lemma 48. □

4.3 Proof of Theorem 2

In this subsection, we prove Theorem 2, which directly follows from Lemma 53 and Lemma 54, stated and proved below.

Lemma 53. (Item 1 $\implies$ Item 2 in Theorem 2). *If* $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$*, then for every polynomial-time samplable distribution family* $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ *supported over* $\{0,1\}^n \times \{0,1\}^m$*, every polynomial* q*, and for all large enough constant* c*, there exists a probabilistic polynomial-time algorithm* A *such that for all* $n, m, s \in \mathbb{N}$*,*

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}}[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x,y,1^s)] \geq 1 - \frac{1}{q(n,m)},$$

where $\tau(n,m) := n^c \cdot m^{1-1/c}$*.*

Proof. Fix a polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m\rangle}\}_{n,m}$ and a polynomial q. Let $c > 0$ be any constant and let τ be defined as $\tau(n,m) := n^c \cdot m^c$

We define the following language $L \in \mathsf{NP}$.

$$L := \left\{(x,y,w,1^s) \mid |w| = |x|^c \cdot |y|^c \text{ and } \exists\, M \in \{0,1\}^{\leq s} \text{ s.t. } U^{w,y}(M) \text{ outputs } x \text{ within } |w| \text{ steps.}\right\}$$

We then define the following polynomial-time samplable distribution family $\{\mathcal{D}'_{\langle n,m,s\rangle}\}$, where each $\mathcal{D}'_{\langle n,m,s\rangle}$ is given by the following sampling procedure.

1. Sample $(x,y) \sim \mathcal{D}_{\langle n,m\rangle}$.
2. Sample $w \sim \{0,1\}^{n^c \cdot m^c}$.
3. Output $(x,y,w,1^s)$.

By assumption, the distributional $\left(L, \{\mathcal{D}'_{\langle n,m,s\rangle}\}\right) \subseteq \mathsf{HeurBPP}$. That is, there is a probabilistic polynomial-time algorithm B such that for all $n, m, s \in \mathbb{N}$,

$$\Pr_{(x,y,w,1^s)\sim\mathcal{D}'_{\langle n,m,s\rangle}}[B(x,y,w,1^s) = L(x,y,w,1^s)] \geq 1 - \frac{1}{q(n,m)^4},$$

which, by the definition of D' and an averaging argument,

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m\rangle}}\left[\Pr_{w\sim\{0,1\}^{(nm)^c}}[B(x,y,w,1^s) = L(x,y,w,1^s)] \geq 1 - \frac{1}{q(n,m)^2}\right] \geq 1 - \frac{1}{q(n,m)^2}.$$

Consider any (x,y) such that the following holds.

$$\Pr_{w}[B(x,y,w,1^s) = L(x,y,w,1^s)] \geq 1 - \frac{1}{q(n,m)^2}. \tag{36}$$

Note that Equation (36) holds with probability at least $1 - 1/q(n,m)^2$ over $(x,y) \sim \mathcal{D}_{\langle n,m\rangle}$.

Suppose $\mathsf{pK}^{\tau(n,m)}_{2/3}(x \mid y) \leq s$. Then by the definition of L, we have that $L(x,y,w,1^s) = 1$ with probability at least $2/3$ over $w \sim \{0,1\}^{(nm)^c}$, which means $B(x,y,w,1^s) = 1$ with probability at least $2/3 - 1/q(n,m) \geq 3/5$ over $w \sim \{0,1\}^{(nm)^c}$.

Suppose $\mathsf{pK}^{\tau(n,m)}_{1/3}(x \mid y) > s$. Again, by the definition of L, we have that $L(x,y,w,1^s) = 1$ with probability less than $1/3$ over $w \sim \{0,1\}^{(nm)^c}$, which means $B(x,y,w,1^s) = 0$ with probability at least $2/3 - 1/q(n,m) \geq 3/5$ over $w \sim \{0,1\}^{(nm)^c}$.

Let A' be the algorithm defined as $A'(x,y,1^s) := B(x,y,U_{|x|^c\cdot|y|^c},1^s)$. We get that for all n, m, s,

$$\Pr_{A'}[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x,y,1^s)] \geq \frac{3}{5}.$$

Using standard amplification techniques, we can obtain from A' a polynomial-time algorithm A such that

$$\Pr_A[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x, y, 1^s)] \geq 1 - \frac{1}{q(n,m)^2}. \tag{37}$$

In other words, with probability at least $1 - 1/q(n,m)^2$ over $(x,y) \sim \mathcal{D}_{\langle n,m \rangle}$, we get Equation (37). This completes the proof. □

Lemma 54. (Item 2 $\implies$ Item 1 in Theorem 2). *Suppose for every polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m \rangle}\}_{n,m}$ supported over $\{0,1\}^n \times \{0,1\}^m$, every polynomial q, and for all large enough constant c, there exists a probabilistic polynomial-time algorithm A such that for all $n, m, s \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m \rangle}}[A \textit{ decides } \mathsf{Cond\text{-}pK}[\tau] \textit{ on } (x, y, 1^s)] \geq 1 - \frac{1}{q(n,m)},$$

where $\tau(n,m) := n^c \cdot m^{1-1/c}$. Then $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$.

Proof. Let $L \in \mathsf{NP}$, $\{\mathcal{C}_v\}_v$ be a polynomial-time samplable distribution family, and $\beta \geq 2$ be a constant.

Fix v, we will show how to solve L in polynomial time with probability at least $1 - v^{-\beta}$ over $\mathcal{C}_v$. Then the lemma follows from Lemma 13.

Let R_0 be a reduction from L to $\mathsf{Gap}_g\text{-}\mathsf{MMSA}$, where g is the function in Lemma 44. That is, for every instance $z \in \{0,1\}^v$, R_0 maps z to (ψ, ζ), where $\psi \in \{0,1\}^{v' := \mathsf{poly}(v)}$ and $\zeta \in \{0,1\}^{\log v'}$, such that

- if $z \in L$, then $w(A_\psi) \leq \zeta$ and
- if $z \in L$, then $w(A_\psi) \leq g(|\psi|) \cdot \zeta$, where $g(v') := (v')^{1/(\log\log z)^{O(1)}}$.

Let R be the randomized algorithm in Lemma 45 instantiated with $\kappa := 4\beta$. Denote by $n'(v)$ the length of the first part of the output of $R(R_0(z), -)$, where $z \in \{0,1\}^v$.

Define the following polynomial-time samplable distribution family $\{\mathcal{D}_{\langle n,m \rangle}\}_{n,m}$ where each $\mathcal{D}_{\langle n,m \rangle}$ is defined by sampling procedure as follows. Given $1^{\langle n,m \rangle}$, the following is performed.

1. Find v such that $n'(v) = n$.
2. Sample $z \sim \mathcal{C}_v$.
3. Compute $(x, y, 1^\rho) := R(R_0(z), 1^m)$.
4. Output $(x, y) \in \{0,1\}^n \times \{0,1\}^m$.

By the assumption of the lemma, there exist a constant $c > 1$ and a probabilistic polynomial-time algorithm A such that for all $n, m, s \in \mathbb{N}$,

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m \rangle}}[A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x, y, 1^s)] \geq 1 - \frac{1}{(nm)^{4\beta}},$$

where $\tau(n,m) := n^c \cdot m^{1-1/c}$. Also, by a union bound, we have

$$\Pr_{(x,y)\sim\mathcal{D}_{\langle n,m \rangle}}[\forall s \leq O(|x|), A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x, y, 1^s)] \geq 1 - \frac{O(n)}{(nm)^{4\beta}}. \tag{38}$$

Now consider when $n := n'(v)$, and $m := |v'|^{c' \cdot 4\beta}$, where $c' > c$ is the constant such that both the completeness and soundness conditions stated in the second item of Lemma 45 hold.

Note that by Equation (38) and the definition of by the distribution of $\mathcal{D}_{\langle n,m \rangle}$, we obtain

$$\Pr_{\substack{z \sim \mathcal{C}_v \\ (x,y,1^\rho) \sim R(R_0(z),1^m)}} [A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x,y,1^\rho)] \geq 1 - \frac{O(n)}{(nm)^{4\beta}}.$$

Also, by averaging, the above yields that with probability at least

$$1 - \frac{O(n)}{(nm)^{2\beta}} \geq 1 - v^{-2\beta}$$

over $z \sim \mathcal{C}_v$, it holds that

$$\Pr_{(x,y,1^\rho) \sim R(R_0(z),1^m)} [A \text{ decides } \mathsf{Cond\text{-}pK}[\tau] \text{ on } (x,y,1^\rho)] \geq 1 - \frac{1}{(nm)^{2\beta}}. \tag{39}$$

Now consider any $z \in \{0,1\}^v$ such that Equation (39) holds.

Suppose $z \in L$. Then we get that $(\psi, \zeta) = R_0(z)$ is a positive instance of $\mathsf{Gap}_g\text{-}\mathsf{MMSA}$, where g is the function in Lemma 44. This means $w(A_\psi) \leq \zeta$. By the completeness condition of the algorithm R, as described in the second item in Lemma 45, we get that $\mathsf{pK}^{\tau(n,m)}(x \mid y) \leq \rho$.

Similarly, if $z \notin L$, then with probability at least $1 - |v|^{-4\beta}$ over $(x,y,1^\rho) \sim R(R_0(z),1^m)$, $\mathsf{pK}^{\tau(n,m)}(x \mid y) > \rho$.

In particular, by Equation (39) and a union bound, the above implies that

$$\Pr_{(x,y,1^\rho) \sim R(R_0(z),1^m)} [A(x,y,1^\rho) = L(z)] \geq 1 - \frac{1}{(nm)^{2\beta}} - \frac{1}{|v|^{4\beta}} \geq 1 - \frac{1}{v^{2\beta}}.$$

Consequently, the following algorithm $A'(z) := A(R(R_0(z),1^m))$, where m is the correct number so that the argument above holds (note that m can be efficiently obtained from the length of z), computes L with probability at least $1 - v^{-\beta}$ over $z \sim \mathcal{C}_v$ and over the internal randomness of A'. This completes the proof. □

4.4 Excluding Pessiland via NP-Hardness of Computing Conditional pK^t

Corollary 55. *Suppose there exists an* NP*-hard problem L such that for every constant $\kappa \geq 1$, there is a randomized polynomial-time reduction R such that for all polynomial τ, there exist polynomials $n, m \colon \mathbb{N} \to \mathbb{N}$ such that for every instance z of L, $R(z, 1^{\langle n(|z|), m(|z|) \rangle})$ outputs $(x,y,1^s)$, where $|x| = n(|z|)$ and $|y| = m(|z|)$, and with probability at least $1 - |z|^{-\kappa}$,*

- *if $z \in L$, then $\mathsf{pK}^{\tau(|x|,|y|)}_{2/3}(x \mid y) \leq s$, and*
- *if $z \notin L$, then $\mathsf{pK}^{\tau(|x|,|y|)}_{1/3}(x \mid y) > s$.*

Then Pessiland does not exist (i.e., $\mathsf{DistNP} \not\subseteq \mathsf{HeurBPP}$ *implies the non-existence of infinitely-often one-way functions).*

Proof Sketch. If computing pK^t in the polynomial-time regime is NP-hard, in the sense that there exist an NP-hard problem L and a randomized polynomial-time reduction R as specified in the corollary, then using ideas from the proof of Lemma 54, it can be shown that the average-case easiness of computing pK^t over polynomial-time samplable distributions allows us to solve NP on

average over any samplable distribution $\mathcal{D}$, and hence $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$. More precisely, this is done by composing $\mathcal{D}$ with the reduction R to obtain a new distribution $\mathcal{D}'$ and then solving the problem of computing pK^t (in the polynomial-time regime) over $\mathcal{D}'$.

Then by Theorem 1, the non-existence of infinitely-often one-way functions, which implies the average-case easiness of computing pK^t over polynomial-time samplable distributions, will also imply $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$. □

5 Characterizing $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ by Approximating Kolmogorov Complexity

In this section, we prove the following theorem which yields Theorem 9.

Theorem 56. *The following are equivalent.*

1. $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$.
2. *For every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}[\mathsf{K}(x \mid y) \le A(x,y) \le \mathsf{K}(x \mid y) + \log p(n)] \ge 1 - \frac{1}{q(n)}.$$

3. *For every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}[\mathsf{K}(x,y) \le A(x,y) \le \mathsf{K}(x,y) + \log p(n)] \ge 1 - \frac{1}{q(n)}.$$

Proof. The equivalence of Item 1 and Item 2 follows from Lemma 57 and Lemma 58, stated and proved in Section 5.1 and Section 5.2 respectively.

The equivalence of Item 2 and Item 3 follows from Lemma 61, which is proved in Section 5.3. □

5.1 Approximating Kolmogorov Complexity from Average-Case Easiness of NP

Lemma 57. *If $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$, then Item 2 of Theorem 9 is true, i.e., for every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,*

$$\Pr_{(x,y)\sim\mathcal{D}_n}[\mathsf{K}(x \mid y) \le A(x,y) \le \mathsf{K}(x \mid y) + \log p(n)] \ge 1 - \frac{1}{q(n)}.$$

Proof. Let $\{\mathcal{D}_n\}_n$ be any independent polynomial-time samplable distribution family and q be any polynomial. By Lemma 26, there is a polynomial-time samplable distribution family $\{\mathcal{D}'_n\}_{n\in\mathbb{N}}$ such that for every $n \in \mathbb{N}$,

$$\mathsf{L}_1(\mathcal{D}_n, \mathcal{D}'_n) \le \frac{1}{2q(n)}. \tag{40}$$

Also, $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$ implies that infinitely-often one-way functions do not exist. Then by Theorem 6, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,

$$\begin{aligned} 1 - \frac{1}{2q(n)} &\leq \Pr_{(x,y)\sim\mathcal{D}'_n}[\mathsf{K}(x \mid y) \leq A(x,y) \leq \mathsf{K}(x \mid y) + \log p(n)] \\ &\leq \Pr_{(x,y)\sim\mathcal{D}_n}[\mathsf{K}(x \mid y) \leq A(x,y) \leq \mathsf{K}(x \mid y) + \log p(n)] + \frac{1}{2q(n)}, \end{aligned}$$

where the second inequality follows from Equation (40). The lemma follows by rearranging the above inequality. □

5.2 Average-Case Easiness of NP from Approximating Kolmogorov Complexity

Lemma 58. *Suppose Item 2 of Theorem 9 is true, i.e., for every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,*

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

Then $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$.

We first show the following technical lemma.

Lemma 59. *If Item 2 of Theorem 9 is true, then for every independent polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and for every polynomial q, there exists a polynomial p such that for all $n \in \mathbb{N}$,*

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

Proof. Let $\mathcal{D} := \{\mathcal{D}_n\}_n$ be any independent polynomial-time samplable distribution family, and let $\mathcal{A}$ and $\mathcal{B}$ be the pair of distribution families that witness the independent polynomial-time samplability of $\mathcal{D}$. Let q be any polynomial.

Let $\mathcal{E}'$ be the following distribution:

Sample $(x,y) \sim \mathcal{D}_n$, $k \sim [2n]$, $z \sim \{0,1\}^{nk}$, and output $(\mathsf{DP}_k(x;z), y)$.

Similarly, let $\mathcal{E}''$ be the following distribution:

Sample $(x,y) \sim \mathcal{D}_n$, $k \sim [2n]$, $w \sim \{0,1\}^{nk+k}$, and output (w,y).

We claim the following.

Claim 60. *There is an independent polynomial-time samplable distribution[10] $\mathcal{E}$ such that for every (α, y),*

1. $\mathcal{E}(\alpha,y) \geq \mathcal{E}'(\alpha,y)/2$, *and*
2. $\mathcal{E}(\alpha,y) \geq \mathcal{E}''(\alpha,y)/2$.

[10]By padding, we can ensure that each of $\mathcal{E}'$, $\mathcal{E}''$ and $\mathcal{E}$ always outputs strings of the same length. This will not affect the correctness of the argument. We omit this technicality for simplicity of presentation.

Proof of Claim 60. First note that the independent polynomial-time samplability of $\mathcal{D}$ is witnessed by the pair of distributions $\mathcal{A}$ and $\mathcal{B}$. That is, sampling $(x, y) \sim \mathcal{D}_n$ is equivalent to first sampling $y \sim \mathcal{A}_n$ and then $x \sim \mathcal{B}_n(\cdot \mid y)$.

We now define the distribution family $\mathcal{E}$, where each $\mathcal{E}_n$ is given by the following sampling procedure.

> We first sample $y \sim \mathcal{A}_n$ and then sample $v \sim \mathcal{F}_n(\cdot \mid y)$, where $\mathcal{F}_n$ is the following polynomial-time samplable distribution:
>
> 1. sample $(x, y) \sim \mathcal{B}_n$, $k \sim [2n]$, $z \sim \{0,1\}^{nk}$ and $w \sim \{0,1\}^{nk+k}$;
> 2. with probability $1/2$, output $(\mathsf{DP}_k(x; z), y)$ and with probability $1/2$, output (w, y).
>
> Finally, we output (v, y).

It is easy to see that $\mathcal{E}$ is independent polynomial-time samplable (witnessed by $\mathcal{A}$ and $\mathcal{F}$). To see the two conditions stated in the claim, note that $\mathcal{E}_n$ can be equivalently sampled as follows:

> We first sample $y \sim \mathcal{A}_n$ and then sample $x \sim \mathcal{B}_n(\cdot \mid y)$ (which is the same as sampling $(x, y) \sim \mathcal{D}_n$ in the first step of sampling $\mathcal{E}'$ and $\mathcal{E}'$). We then sample $k \sim [2n]$. Finally, with probability $1/2$, we sample $z \sim \{0,1\}^{nk}$ and output $(\mathsf{DP}_k(x; z), y)$ and with probability $1/2$, we sample $w \sim \{0,1\}^{nk+k}$ and output (w, y).

This completes the proof of Claim 60. ◇

Now by the assumption that Item 2 of Theorem 9 is true, we have that for the independent polynomial-time samplable distribution $\mathcal{E}$ and for q' such that $q'(n) := 50n \cdot q(n)$, there exist a polynomial p' and a probabilistic polynomial-time algorithm $A_{\mathcal{E}}$ such that for all $n \in \mathbb{N}$,

$$\Pr_{(v,y)\sim\mathcal{E}_n}\left[\mathsf{K}(v \mid y) \le A_{\mathcal{E}}(v, y) \le \mathsf{K}(v \mid y) + \log p'(n)\right] \ge 1 - \frac{1}{q'(n)}. \tag{41}$$

Since the direct product generator DP_k is computable, we have for every $z \in \{0,1\}^{nk}$,

$$\mathsf{K}(\mathsf{DP}_k(x; z) \mid y) \le \mathsf{K}(x \mid y) + |z| + O(\log n) \le \mathsf{K}(x \mid y) + nk + \log p'(n),$$

where the last inequality holds by choosing a large enough polynomial p'. Then using Equation (41), we have

$$\begin{aligned} 1 - \frac{2}{q'(n)} &\le \Pr_{\substack{(x,y)\sim\mathcal{D}_n \\ k\sim[2n] \\ z\sim\{0,1\}^{nk}}}\left[A_{\mathcal{E}}(\mathsf{DP}_k(x; z), y) \le \mathsf{K}(\mathsf{DP}_k(x; z) \mid y)\right] \\ &\le \Pr_{\substack{(x,y)\sim\mathcal{D}_n \\ k\sim[2n] \\ z\sim\{0,1\}^{nk}}}\left[A_{\mathcal{E}}(\mathsf{DP}_k(x; z), y) \le \mathsf{K}(x \mid y) + nk + 2\log p'(n)\right], \end{aligned} \tag{42}$$

where the factor 2 in $\frac{2}{q'(n)}$ comes from the first item of Claim 60 as we switch from the distribution $\mathcal{E}$ in Equation (41) to $\mathcal{E}'$.

By averaging, Equation (42) yields that with probability at least $1 - 20n/q'(n)$ over $(x, y) \sim \mathcal{D}_n$,

$$\Pr_{\substack{k\sim[2n] \\ z\sim\{0,1\}^{nk}}}\left[A_{\mathcal{E}}(\mathsf{DP}_k(x; z), y) \le \mathsf{K}(x \mid y) + nk + 2\log p'(n)\right] \ge 1 - \frac{1}{10n}.$$

Note that the above implies that for *all* $k \in [2n]$,

$$\mathop{\mathbf{Pr}}_{z\sim\{0,1\}^{nk}}\left[A_{\mathcal{E}}(\mathsf{DP}_k(x;z),y) \le \mathsf{K}(x \mid y) + nk + 2\log p'(n)\right] \ge 1 - \frac{1}{5}. \tag{43}$$

By a simple counting argument, we have

$$\mathsf{K}(w \mid y) \ge |w| - \log q'(n) > |w| - \log p'(n)$$

with probability at least $1 - \frac{1}{q'(n)}$ over $w \sim \{0,1\}^{nk+k}$. Following a similar argument as in previous paragraphs, we can show that with probability at least $1 - 30n/q'(n)$ over $(x,y) \sim \mathcal{D}_n$,

$$\mathop{\mathbf{Pr}}_{w\sim\{0,1\}^{nk+k}}\left[A_{\mathcal{E}}(w,y) > k + nk - \log p'(n)\right] \ge \frac{4}{5}, \tag{44}$$

for all $k \in [2n]$.

By a union bound, with probability at least at least $1-50n/q'(n) = 1-1/q(n)$ over $(x,y) \sim \mathcal{D}_n$, both Equations (43) and (44) hold. Fix any such (x,y), and let

$$k^* = k^*(x,y) := \mathsf{K}(x \mid y) + 3\log p'(n).$$

Note that in this case, Equation (43) yields

$$\mathop{\mathbf{Pr}}_{z\sim\{0,1\}^{nk^*}}\left[A_{\mathcal{E}}(\mathsf{DP}_{k^*}(x;z),y) \le k^* + nk^* - \log p'(n)\right] \ge \frac{4}{5}. \tag{45}$$

Now let $D_{k^*,y}\colon \{0,1\}^{nk^*+k^*} \to \{0,1\}$ be such that

$$D_{k^*,y}(w) = 1 \iff A_{\mathcal{E}}(w,y) \le nk^* + k^* - \log p(n).$$

Note that $D_{k^*,y}$ has $\mathsf{poly}(n)$ (randomized) running time given k^* and y. It follows from Equations (44) and (45) that,

$$\mathop{\mathbf{Pr}}_{z\sim\{0,1\}^{nk^*}}[D_{k^*,y}(\mathsf{DP}_{k^*}(x;z)) = 1] - \mathop{\mathbf{Pr}}_{w\sim\{0,1\}^{nk^*+k^*}}[D_{k^*,y}(w) = 1] \ge \frac{3}{5}.$$

Applying Lemma 33 to $D_{k^*,y}$, we obtain

$$\mathsf{pK}^{p'(n)}(x \mid y) \le k^* + \log p'(n), \tag{46}$$

provided that p' is a large enough polynomial. Since $k^* = \mathsf{K}(x,y) + 3\log p'(n)$, it follows that

$$\mathop{\mathbf{Pr}}_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{pK}^{p'(n)}(x \mid y) \le \mathsf{K}(x \mid y) + 4\log p'(n)\right] \ge 1 - \frac{1}{q(n)},$$

which completes the proof by letting $p > p'$ be a sufficiently large polynomial. □

We are now ready to show Lemma 58.

Proof of Lemma 58. We will show that the assumption of the lemma implies Item 2 of Theorem 30, which then implies $\mathsf{DistNP} \subseteq \mathsf{HeurBPP}$. That is, we want to show that for every independent

polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ and for every polynomial q, there exists a polynomial p such that for all $n \in \mathbb{N}$,

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{pK}^{p(n)}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + \log p(n)\right] \geq 1 - \frac{1}{q(n)}. \tag{47}$$

On the one hand, by the coding theorem for time-unbounded Kolmogorov complexity (Theorem 18), we have for every $(x, y) \in \mathsf{Support}(\mathcal{D}_n)$,

$$\mathsf{K}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + O(\log n). \tag{48}$$

On the other hand, by the assumption of the lemma and Lemma 59, we have for some p_0,

$$\Pr_{(x,y)\sim\mathcal{D}_n}\left[\mathsf{pK}^{p_0(n)}(x \mid y) \leq \mathsf{K}(x \mid y) + \log p_0(n)\right] \geq 1 - \frac{1}{q(n)}. \tag{49}$$

By combining Equations (48) and (49), and letting p to be a large enough polynomial, we get Equation (47). □

5.3 Kolmogorov Complexity versus Conditional Kolmogorov Complexity

Lemma 61. *Item 2 and Item 3 of Theorem 56 are equivalent.*

Proof Sketch. The idea is to use symmetry of information, which roughly says that for all strings x and y,

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

Suppose Item 2 of Theorem 56 is true. Then for any independent polynomial-time samplable distribution $\mathcal{D}$, we have an algorithm that approximates $\mathsf{K}(x \mid y)$ over $(x, y) \sim \mathcal{D}_n$. Also, it is not hard to show that one can also use Item 2 to obtain an algorithm that approximates $\mathsf{K}(y)$ over $y \sim \mathcal{D}_n^{(2)}$, where $\mathcal{D}_n^{(2)}$ is the marginal distribution of $\mathcal{D}_n$ on the second half. (Hint: Consider the polynomial-time samplable distribution that samples $y \sim \mathcal{D}_n^{(2)}$ and outputs $(y, 0^n)$.) This enables us to approximate $\mathsf{K}(x, y)$ by using symmetry of information stated above, which yields Item 3 of Theorem 56.

The other direction can be shown by a similar argument. □

6 Characterizing NP ⊆ BPP by Approximating Kolmogorov Complexity

The goal of this section is to establish Theorem 7, which follows from Lemma 62 and Lemma 63, stated and proved in Section 6.1 and Section 6.2 respectively.

6.1 Approximating Kolmogorov Complexity from Worst-Case Easiness of NP

Lemma 62. *If* $\mathsf{NP} \subseteq \mathsf{BPP}$, *then Item 2 of Theorem 7 is true, i.e., for every polynomial-time samplable l family* $\{\mathcal{D}_n\}_n$, *where each* $\mathcal{D}_n$ *is over* $\{0,1\}^n \times \{0,1\}^n$, *and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all* $n \in \mathbb{N}$ *and* $y \in \{0,1\}^n$,

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

Proof. Let $\{\mathcal{D}_n\}_n$ be any polynomial-time samplable distribution family, where each $\mathcal{D}_n$ is over $\{0,1\}^n \times \{0,1\}^n$ Let q be any polynomial. Fix any large enough $n \in \mathbb{N}$.

From the assumption $\mathsf{NP} \subseteq \mathsf{BPP}$, we get $\mathsf{PH} \subseteq \mathsf{BPP}$. Then by Lemma 27, there exists a probabilistic polynomial-time algorithm B such that for all $(x,y) \in \mathsf{Support}(\mathcal{D}_n)$,

$$\Pr_B[\mathcal{D}_n(x \mid y)/4 \leq B(x,y) \leq \mathcal{D}_n(x \mid y)] \geq 1 - \frac{1}{2q(n)}. \tag{50}$$

On the one hand, by the coding theorem for time-unbounded Kolmogorov complexity (Theorem 18) and the fact that given any y, $\mathcal{D}_n(\cdot \mid y)$ is computable, we have for all $(x,y) \in \mathsf{Support}(\mathcal{D}_n)$,

$$\mathsf{K}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + b \cdot \log n, \tag{51}$$

for some constant $b > 0$. On the other hand, by using Lemma 19, we can show that for every $y \in \mathsf{Support}\left(\mathcal{D}_n^{(2)}\right)$,

$$\Pr_{x \sim \mathcal{D}_n(\cdot|y)}\left[\mathsf{K}(x \mid y) \geq \log \frac{1}{\mathcal{D}_n(x \mid y)} - \log p'(n)\right] \geq 1 - \frac{1}{2q(n)}, \tag{52}$$

for some polynomial p'.

Our algorithm A, which aims to approximate $\mathsf{K}(x \mid y)$ for every y and most $x \sim \mathcal{D}_n(\cdot \mid y)$, does the following.

> On $x, y \in \{0,1\}^n$, let $\beta := B(x,y)$ and output $\log(1/\beta) + b \cdot \log n$, where b is the constant in Equation (51).

It is easy to see that A runs in polynomial time. To see its correctness, note that if all of Equation (50), Equation (51), and Equation (52) hold, which happens with probability at least $1 - 1/q(n)$ over $x \sim \mathcal{D}_n(\cdot \mid y)$ (for every y) and the internal randomness of A, we have both

$$\log \frac{1}{B(x,y)} + b \cdot \log n \geq \log \frac{1}{\mathcal{D}_n(x \mid y)} + b \cdot \log n \geq \mathsf{K}(x \mid y),$$

and

$$\log \frac{1}{B(x,y)} + b \cdot \log n \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + b \cdot \log n + 2 \leq \mathsf{K}(x \mid y) + b \cdot \log n + 2 + \log p'(n).$$

The lemma follows by letting $p > p'$ be a large enough polynomial. □

6.2 Worst-Case Easiness of NP from Approximating Kolmogorov Complexity

Lemma 63. *Suppose Item 2 of Theorem 7 is true, i.e., for every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$, where each $\mathcal{D}_n$ is over $\{0,1\}^n \times \{0,1\}^n$, and every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$ and $y \in \{0,1\}^n$,*

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

Then $\mathsf{NP} \subseteq \mathsf{BPP}$.

To show Lemma 63, we first observe that for the characterization of "$\mathsf{NP} \subseteq \mathsf{BPP}$" in Theorem 31, it suffices to use a notion of *semi-worst-case* conditional coding instead of *worst-case* conditional coding. We present this characterization in the following subsection.

6.2.1 Worst-Case Easiness of NP and Semi-Worst-Case Conditional Coding

Lemma 64. *The following are equivalent.*

1. $\mathsf{NP} \subseteq \mathsf{BPP}$.
2. **(Semi-Worst-Case Conditional Coding)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ and every polynomial q, there exists a polynomial p such that for all $n \in \mathbb{N}$ and all $y \in \mathsf{Support}\left(\mathcal{D}_n^{(2)}\right)$,*

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

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

We need the following notion of a universal sampler.

Definition 65 (Universal Time-Bounded Sampler)**.** Let $n, t \in \mathbb{N}$ and $y \in \{0,1\}^*$. The universal sampler $\mathsf{USamp}(1^n, 1^t, y)$ does the following.

1. Pick a uniformly random $k \sim [2n]$,
2. Pick a uniformly random $r \sim \{0,1\}^t$,
3. Pick a uniformly random $d \sim \{0,1\}^k$,
4. Outputs x which is the output of a universal oracle Turing machine (fixed in advance) U, on input d with an oracle to the bits of y and r (i.e., $U^{y,r}(d)$), running for t steps.

Note that USamp runs in polynomial time. The following proposition follows easily from the definitions of pK^t and USamp.

Proposition 66. *For every $n, t, \ell \in \mathbb{N}$, $x \in \{0,1\}^n$ and $y \in \{0,1\}^*$, if $\mathsf{pK}^t(x \mid y) \leq k$, then $\mathsf{USamp}(1^n, 1^t, y)$ outputs x with probability $\Omega\big(1/(n \cdot 2^k)\big)$, where USamp is the universal sampler defined in Definition 65.*

We now show Lemma 64.

Proof of Lemma 64. ($\implies$) This direction follows from Theorem 31, as worst-case conditional coding implies semi-worst conditional coding.

($\impliedby$) Without loss of generality (by padding), we show how to solve every language $L \in \mathsf{NP}$ where a yes-instance $x \in L \cap \{0,1\}^m$ admits a witness of length $m+1$ (with respect to a fixed verifier V_L).

Consider the distribution family $\{\mathcal{D}_n\}_n$, where each $\mathcal{D}_n$ is given by the following sampling algorithm:

1. Sample a uniformly random $x \sim \{0,1\}^{n-1}$,
2. Sample a uniformly random $w \sim \{0,1\}^n$,
3. Output $\begin{cases} (w, x1), & \text{if } w \text{ is a } V_L\text{-witness for } x \\ (0^n, x0), & \text{otherwise} \end{cases}$

Consider any $x \in L \cap \{0,1\}^{n-1}$, and let $W \subseteq \{0,1\}^n$ be the set of V_L-witnesses for x. Note that by construction, $\mathcal{D}_n(\cdot \mid x1)$ is uniformly distributed over W. By the assumed semi-worst-case conditional coding condition (Item 2 of Lemma 64), there exist a polynomial p and a constant $c > 0$ such that for at least $1/n^c$ of the $w \in W$,

$$\begin{aligned} \mathsf{pK}^{p(n)}(w \mid x1) &\leq \log \frac{1}{\mathcal{D}_n(w \mid x1)} + \log p(n) \\ &= \log |W| + \log p(n). \end{aligned} \tag{53}$$

Let $W' \subseteq W$ be the set of w that satisfies Equation (53). Note that $|W'| \geq |W|/n^c$. By Proposition 66, for *each* $w \in W'$, $\mathsf{USamp}(1^n, 1^{p(n)}, x1)$ outputs w with probability at least

$$\frac{1}{O(n \cdot p(n) \cdot |W|)}.$$

Hence the probability that $\mathsf{USamp}(1^n, 1^{p(n)}, x1)$ outputs *some* $w \in W'$ is at least

$$|W'| \cdot \frac{1}{O(n \cdot p(n) \cdot |W|)} \geq \frac{1}{O(n \cdot p(n) \cdot n^c)}.$$

In other words, $\mathsf{USamp}(1^n, 1^{p(n)}, x1)$ outputs a witness for x with probability at least $1/\mathsf{poly}(n)$. By standard amplification, this yields an efficient randomized algorithm for solving L with high probability. □

6.2.2 Proof of Lemma 63

Given the characterization of $\mathsf{NP} \subseteq \mathsf{BPP}$ by semi-worst-case conditional coding (Lemma 64), we now sketch the proof of Lemma 63.

Proof Sketch of Lemma 63. The proof follows a similar approach to that of Lemma 58.

First of all, by carefully adapting the proof of Lemma 59, we show that using the assumption of the lemma, for every polynomial-time distribution family $\mathcal{D} := \{\mathcal{D}_n\}_n$ over $\{0,1\}^n \times \{0,1\}^n$, we can upper bound $\mathsf{pK}^{\mathsf{poly}(n)}(x \mid y)$ by $\mathsf{K}(x \mid y)$ for every $y \in \{0,1\}^n$ and almost all $x \sim \mathcal{D}_n(\cdot \mid y)$. More specifically, let

- $\mathcal{E}'$ be the distribution that samples $(x, y) \sim \mathcal{D}$, $k \sim [2n]$, $z \sim \{0,1\}^n$ and outputs $(\mathsf{DP}_k(x; z), y)$, and
- $\mathcal{E}''$ be the distribution that samples $(x, y) \sim \mathcal{D}$, $k \sim [2n]$, $w \sim \{0,1\}^{nk+k}$ and outputs (w, y).

Now let $\mathcal{E}$ be the following distribution.

> Sample $(x, y) \sim \mathcal{D}$, $k \sim [2n]$, $z \sim \{0,1\}^n$, $w \sim \{0,1\}^{nk+k}$. With probability $1/2$ output $(\mathsf{DP}_k(x; z), y)$, and with probability $1/2$ output (w, y).

Note that $\mathcal{E}$ is polynomial-time samplable, and for every (α, y), we have $\mathcal{E}(\alpha, y) \geq \mathcal{E}'(\alpha, y)/2$ (*resp.* $\mathcal{E}(\alpha, y) \geq \mathcal{E}''(\alpha, y)/2$). Also note that $\mathcal{E}^{(2)}$ and $(\mathcal{E}')^{(2)}$ (*resp.* $(\mathcal{E}'')^{(2)}$) are identical. Then we have for every (α, y),

$$\begin{aligned} \mathcal{E}(\alpha \mid y) &= \frac{\mathcal{E}(\alpha, y)}{\mathcal{E}^{(2)}(y)} \\ &\geq \frac{\mathcal{E}'(\alpha, y)/2}{(\mathcal{E}')^{(2)}(y)} \\ &= \frac{\mathcal{E}'(\alpha \mid y)}{2}. \end{aligned}$$

Similarly, we have $\mathcal{E}(\alpha \mid y) \geq \mathcal{E}''(\alpha \mid y)/2$.

Now by applying the algorithm in the assumption of the lemma to $\mathcal{E}$ and following a similar argument as in the proof of Lemma 59, we can obtain that for *every* y and with high probability over $x \sim \mathcal{D}_n(\cdot \mid y)$, there is a polynomial-time algorithm that given k^* and y, distinguishes $\mathsf{DP}_{k^*}(x; \mathcal{U}_{nk^*})$ from $\mathcal{U}_{nk^*+k^*}$, for $k^* := \mathsf{K}(x \mid y) + O(\log n)$. Then by applying Lemma 33, we get that for every polynomial q, there exists a polynomial p such that for every y,

$$\Pr_{x \sim \mathcal{D}_n(\cdot|y)}\left[\mathsf{pK}^{p(n)}(x \mid y) \leq \mathsf{K}(x \mid y) + \log p(n)\right] \geq 1 - \frac{1}{q(n)}. \tag{54}$$

On the other hand, by the coding theorem (Theorem 18), we have

$$\mathsf{K}(x \mid y) \leq \log \frac{1}{\mathcal{D}_n(x \mid y)} + O(\log n). \tag{55}$$

Equations (54) and (55) together yield the semi-worst-case conditional coding condition as stated in Item 2 of Lemma 64, which implies $\mathsf{NP} \subseteq \mathsf{BPP}$. □

7 Characterizing Auxiliary-Input One-Way Functions by Approximating Kolmogorov Complexity

We prove Theorem 10 in this section. The proof follows a similar approach to that of Theorem 6 in [IRS22].

Lemma 67. *If auxiliary-input one-way functions do not exist, then for every sequence of strings $\{y_n\}_n$ where each $y_n \in \{0,1\}^n$, every distribution family $\{\mathcal{D}_n\}_n$ samplable in polynomial time using $\{y_n\}_n$ as advice, where each $\mathcal{D}_n$ is over $\{0,1\}^n$, and for every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,*

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

Proof Sketch. The proof is similar to that of Lemma 62.

Let $y_n \in \{0,1\}^n$ be a sequence of string and $\{\mathcal{D}_n\}_n$ be a distribution family samplable in polynomial-time using $\{y_n\}_n$ as advice. Note that $\{\mathcal{D}_n\}_n$ can be sampled using a family of s-size circuits $\{C_n \colon \{0,1\}^{s(n)} \to \{0,1\}^n\}_n$ where s is some polynomial. Also, C_n can be constructed in $\mathsf{poly}(n)$-time given y_n.

On the one hand, we have that with high probability over $x \sim \mathcal{D}_n$,

$$\mathsf{K}(x \mid y_n) \approx \log \frac{1}{\mathcal{D}_n(x)}. \tag{56}$$

This again can be shown using the coding theorem for time-unbounded Kolmogorov complexity (Theorem 18) and Lemma 19.

On the other hand, by Theorem 28, we have a probabilistic polynomial-time algorithm that given y_n, estimates $\mathcal{D}_n(x)$ within a multiplicative constant factor with high probability over $x \sim \mathcal{D}_n$. This enables us to estimate $\mathsf{K}(x \mid y_n)$ using Equation (56). □

Lemma 68. *Suppose for every sequence of strings $\{y_n\}_n$ where each $y_n \in \{0,1\}^n$, every distribution family $\{\mathcal{D}_n\}_n$ samplable in polynomial time using $\{y_n\}_n$ as advice, where each $\mathcal{D}_n$ is over $\{0,1\}^n$,*

and for every polynomial q, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,

$$\Pr_{x \sim \mathcal{D}_n}[\mathsf{K}(x \mid y_n) \le A(x, y_n) \le \mathsf{K}(x \mid y_n) + \log p(n)] \ge 1 - \frac{1}{q(n)}.$$

Then auxiliary-input one-way functions do not exist.

Proof. We show that if the assumption of the lemma is true, then auxiliary-input pseudorandom generators do not exist.

Let $\{G_z \colon \{0,1\}^{s(n)} \to \{0,1\}^{\ell(n)}\}_{z \in \{0,1\}^*}$ be any candidate auxiliary-input pseudorandom generator, where s, ℓ are polynomials such that $s(n) < \ell(n)$. Without loss of generality, we assume $\ell(n) = n$. Also, by standard techniques in cryptography for increasing the stretch of a PRG, we assume that $s(n) \le \ell(n)/2$.

Fix any large enough $n \in \mathbb{N}$ and any $z \in \{0,1\}^n$. Let $\mathcal{D}_n$ be the uniform mixture of $G_z(\mathcal{U}_{s(n)})$ and $\mathcal{U}_n$, i.e., we sample $\mathcal{D}_n$ as follows.

> Sample $r \sim \{0,1\}^{s(n)}$ and $u \sim \{0,1\}^n$. With probability $1/2$, output $G_z(r)$, and with probability $1/2$, output u.

Note that $\mathcal{D}_n$ is polynomial-time sample using z as advice. Let q be a large enough polynomial specified later. Then by the assumption of the lemma, there exist a probabilistic polynomial-time algorithm A and a polynomial p such that

$$\Pr_{x \sim \mathcal{D}_n}[\mathsf{K}(x \mid z) \le A(x, z) \le \mathsf{K}(x \mid y_n) + \log p(n)] \ge 1 - \frac{1}{q(n)}. \tag{57}$$

Now note that for every $r \in \{0,1\}^{s(n)}$, we have

$$\mathsf{K}(G_z(r) \mid z) \le s(n) + \log p(n),$$

given that p is a sufficiently large polynomial. Then since $\mathcal{D}_n$ samples from $G_z(\mathcal{U}_{s(n)})$ with probability at least $1/2$, by Equation (57), we have

$$\Pr_{r \sim \{0,1\}^{s(n)}}[A(G_z(r), z) \le s(n) + 2\log p(n)] \ge 1 - \frac{2}{q(n)}. \tag{58}$$

Also, by a simple counting argument, with probability at least $1 - 1/n$ over $u \sim \{0,1\}^n$, we have

$$\mathsf{K}(u \mid z) \ge n - O(\log n) > s(n) + 2\log p(n).$$

Again, since $\mathcal{D}_n$ samples from $\mathcal{U}_n$ with probability at least $1/2$, using Equation (57), we can show that

$$\Pr_{u \sim \{0,1\}^n}[A(u, z) > s(n) + 2\log p(n)] \ge 1 - \frac{2}{q(n)} - \frac{1}{n}. \tag{59}$$

Now let $D \colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be the randomized algorithm such that

$$D(w, z) = 1 \iff A(w, z) \le s(n) + 2\log p(n).$$

It follows from Equations (58) and (59) that D distinguishes $G_z(\mathcal{U}_{s(n)})$ from $\mathcal{U}_n$, so $\{G_z\}_z$ cannot be an auxiliary-input pseudorandom generator. □

Acknowledgment

We thank Shuichi Hirahara, Yanyi Liu, Igor C. Oliveira, and Hanlin Ren for useful discussions.

References

[ACM+21] Eric Allender, Mahdi Cheraghchi, Dimitrios Myrisiotis, Harsha Tirumala, and Ilya Volkovich. One-way functions and a conditional variant of MKTP. In *Conference on Foundations of Software Technology and Theoretical Computer Science* (FSTTCS), pages 7:1–7:19, 2021.

[BL88] Josh Cohen Benaloh and Jerry Leichter. Generalized secret sharing and monotone functions. In *Annual Cryptology Conference* (CRYPTO), pages 27–35, 1988.

[BLvM05] Harry Buhrman, Troy Lee, and Dieter van Melkebeek. Language compression and pseudorandom generators. *Comput. Complex.*, 14(3):228–255, 2005.

[BT06] Andrej Bogdanov and Luca Trevisan. Average-case complexity. *Found. Trends Theor. Comput. Sci.*, 2(1), 2006.

[DHK15] Irit Dinur, Prahladh Harsha, and Guy Kindler. Polynomially low error PCPs with polyloglog n queries via modular composition. In *Symposium on Theory of Computing* (STOC), pages 267–276, 2015.

[DS04] Irit Dinur and Shmuel Safra. On the hardness of approximating label-cover. *Inf. Process. Lett.*, 89(5):247–254, 2004.

[GKLO22] Halley Goldberg, Valentine Kabanets, Zhenjian Lu, and Igor C. Oliveira. Probabilistic Kolmogorov complexity with applications to average-case complexity. In *Computational Complexity Conference* (CCC), pages 16:1–16:60, 2022.

[HIL+23] 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 *Symposium on Theory of Computing* (STOC), pages 1039–1050, 2023.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[Hir20] Shuichi Hirahara. Characterizing average-case complexity of PH by worst-case meta-complexity. In *Symposium on Foundations of Computer Science* (FOCS), pages 50–60, 2020.

[Hir21] Shuichi Hirahara. Average-case hardness of NP from exponential worst-case hardness assumptions. In *Symposium on Theory of Computing* (STOC), pages 292–302, 2021.

[Hir22] Shuichi Hirahara. Symmetry of information from meta-complexity. In *Computational Complexity Conference* (CCC), pages 26:1–26:41, 2022.

[Hir23] Shuichi Hirahara. Capturing one-way functions via NP-hardness of meta-complexity. In *Symposium on Theory of Computing* (STOC), pages 1027–1038, 2023.

[HKLO24] Shuichi Hirahara, Valentine Kabanets, Zhenjian Lu, and Igor C. Oliveira. Exact search-to-decision reductions for time-bounded Kolmogorov complexity. *Electronic Colloquium on Computational Complexity* (ECCC), TR24-059, 2024.

[HS17] Shuichi Hirahara and Rahul Santhanam. On the average-case complexity of MCSP and its variants. In *Computational Complexity Conference* (CCC), pages 7:1–7:20, 2017.

[HS22] Shuichi Hirahara and Rahul Santhanam. Excluding PH pessiland. In *Innovations in Theoretical Computer Science Conference* (ITCS), pages 85:1–85:25, 2022.

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *Symposium on Theory of Computing* (STOC), pages 230–235, 1989.

[IL90] Russell Impagliazzo and Leonid A. Levin. No better ways to generate hard NP instances than picking uniformly at random. In *Symposium on Theory of Computing* (STOC), pages 812–821, 1990.

[Imp95] Russell Impagliazzo. A personal view of average-case complexity. In *Proceedings of the Tenth Annual Structure in Complexity Theory Conference*, pages 134–147, 1995.

[IRS21] Rahul Ilango, Hanlin Ren, and Rahul Santhanam. Hardness on any samplable distribution suffices: New characterizations of one-way functions by meta-complexity. *Electron. Colloquium Comput. Complex.*, page 82, 2021.

[IRS22] Rahul Ilango, Hanlin Ren, and Rahul Santhanam. Robustness of average-case meta-complexity via pseudorandomness. In *Symposium on Theory of Computing* (STOC), pages 1575–1583, 2022.

[LOS21] Zhenjian Lu, Igor C. Oliveira, and Rahul Santhanam. Pseudodeterministic algorithms and the structure of probabilistic time. In *Symposium on Theory of Computing* (STOC), pages 303–316, 2021.

[LOZ22] Zhenjian Lu, Igor C. Oliveira, and Marius Zimand. Optimal coding theorems in time-bounded Kolmogorov complexity. In *International Colloquium on Automata, Languages, and Programming* (ICALP), pages 92:1–92:14, 2022.

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

[LP21] Yanyi Liu and Rafael Pass. On the possibility of basing cryptography on EXP$\neq$BPP. In *International Cryptology Conference* (CRYPTO), pages 11–40, 2021.

[LP22] Yanyi Liu and Rafael Pass. On one-way functions from NP-complete problems. In *Conference on Computational Complexity* (CCC), pages 36:1–36:24, 2022.

[LP23] Yanyi Liu and Rafael Pass. One-way functions and the hardness of (probabilistic) time-bounded Kolmogorov complexity w.r.t. samplable distributions. In *Annual Cryptology Conference* (CRYPTO), pages 645–673, 2023.

[Nan21] Mikito Nanashima. On basing auxiliary-input cryptography on NP-hardness via non-adaptive black-box reductions. In *Innovations in Theoretical Computer Science Conference* (ITCS), pages 29:1–29:15, 2021.

[RS21] Hanlin Ren and Rahul Santhanam. Hardness of KT characterizes parallel cryptography. In *Computational Complexity Conference* (CCC), pages 35:1–35:58, 2021.

[San20] Rahul Santhanam. Pseudorandomness and the minimum circuit size problem. In *Innovations in Theoretical Computer Science Conference* (ITCS), pages 68:1–68:26, 2020.

[SS22] Michael E. Saks and Rahul Santhanam. On randomized reductions to the random strings. In *Computational Complexity Conference* (CCC), pages 29:1–29:30, 2022.

[Sto85] Larry J. Stockmeyer. On approximation algorithms for #P. *SIAM J. Comput.*, 14(4):849–861, 1985.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il