Electronic Colloquium on Computational Complexity, Report No. 136 (2024)

One-Way Functions and pKt Complexity

Shuichi Hirahara* Zhenjian Lu† Igor C. Oliveira‡

Abstract

We introduce pKt complexity, a new notion of time-bounded Kolmogorov complexity that can be seen as a probabilistic analogue of Levin's Kt complexity. Using pKt complexity, we upgrade two recent frameworks that characterize one-way functions (OWF) via symmetry of information and meta-complexity, respectively. Among other contributions, we establish the following results:

(i) OWF can be based on the worst-case assumption that BPEXP is not contained infinitely often in P/poly if the failure of symmetry of information for pKt in the *worst-case* implies its failure on *average*.

(ii) (Infinitely-often) OWF exist if and only if the average-case easiness of approximating pKt with *two-sided* error implies its (mild) average-case easiness with *one-sided* error.

Previously, in a celebrated result, Liu and Pass (CRYPTO 2021 and CACM 2023) proved that one can base (infinitely-often) OWF on the assumption that EXP $\not\subseteq$ BPP if and only if there is a reduction from computing Kt on average with *zero* error to computing Kt on average with *two-sided* error. In contrast, our second result shows that closing the gap between two-sided error and one-sided error average-case algorithms for approximating pKt is both necessary and sufficient to *unconditionally* establish the existence of OWF.

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

†University of Warwick, UK. E-mail: zhen.j.lu@warwick.ac.uk

‡University of Warwick, UK. E-mail: igor.oliveira@warwick.ac.uk

ISSN 1433-8092

Contents

1 Introduction

1.1 Context and Motivation

This paper is primarily concerned with research directions **(1)** and **(2)** described next:

(1) Existence of one-way functions.

A one-way function is a function that is easy to compute but hard to invert on average [DH76]. Due to its equivalence to several basic cryptographic primitives, such as private-key encryption [GM84], pseudorandom generators [HILL99], digital signatures [Rom90], and commitment schemes [Nao91], the existence of one-way functions is widely regarded as the most important open problem in Cryptography. In order to be precise in our subsequent discussion, we capture the question of the existence of one-way functions through the following formal statement:

$\exists\mathsf{OWF}$: There is a function $f = \{f_n\} \in \mathsf{FP}$, where each $f_n \colon \{0,1\}^n \to \{0,1\}^{\mathsf{poly}(n)}$, such that for every probabilistic polynomial time (PPT) algorithm A, and for every large enough n,

$$\Pr_{A,\, x\sim\{0,1\}^n}[A(f_n(x)) \in f_n^{-1}(f_n(x))] \leq \frac{1}{n^{\omega(1)}}.$$

(2) Base one-way functions on a natural worst-case computational assumption.

Since a proof of the existence of one-way functions would imply that $\mathsf{P} \neq \mathsf{NP}$, a less ambitious problem is whether we can base their existence on a widely believed worst-case complexity assumption. This question has also received significant attention over the past several decades (see, e.g. [Imp95, AGGM06, BT06, BB15, Nan21, HN23, BLMP23, LP23a] and references therein).

Recently, there have been attempts to investigate directions **(1)** and **(2)** through the lens of the theory of (time-bounded) Kolmogorov complexity, which studies the minimum encoding length of binary strings according to different measures of complexity. Interestingly, the new approaches are complete, in the sense that they provide both necessary and sufficient conditions to achieve **(1)** and **(2)**. We can categorize these approaches into two strands of research:

(A) Structural theory of time-bounded Kolmogorov complexity.

This research direction relates the existence of one-way functions to the failure of key properties of (time-unbounded) Kolmogorov complexity in the time-bounded setting, such as language compression, conditional coding, and symmetry of information.

(B) Complexity of computing time-bounded Kolmogorov complexity.

This research direction, often referred to as meta-complexity, connects one-way functions to the computational hardness of estimating the time-bounded Kolmogorov complexity of a given string.

In both **(A)** and **(B)**, several measures of (time-bounded) Kolmogorov complexity have been considered, such as $\mathsf{K}^{\mathsf{poly}}$, $\mathsf{pK}^{\mathsf{poly}}$ and $\mathsf{rK}^{\mathsf{poly}}$ in **(A)**, as in [LM93, LW95, HIL+23], and K, $\mathsf{K}^{\mathsf{poly}}$, $\mathsf{pK}^{\mathsf{poly}}$, Kt, KT, and $\mathsf{cK}^{\mathsf{poly}}$ in **(B)**, as in [LP20, LP21, RS21, IRS22, LP22, LP23b].

In this work, we advance this area of research by proposing a new measure of time-bounded Kolmogorov complexity that offers significant benefits for the investigation of connections between **(A)** and **(B)** and directions **(1)** and **(2)**. We describe our contributions in detail next.

1.2 Our Contributions

1.2.1 pKt: A Probabilistic Analogue of Levin's Kt Complexity

Recall that the Kolmogorov complexity of a string $x \in \{0,1\}^*$, denoted $\mathsf{K}(x)$, is the description length $|p| \in \mathbb{N}$ of the shortest program p that prints x. Formally, we fix a universal machine U, and minimize over the length of all strings p such that $U(p) = x$.

Despite the numerous applications of Kolmogorov complexity, its inherent uncomputability becomes an important issue in situations where an upper bound on the running time of algorithms is relevant. To mitigate this issue, Levin [Lev84] introduced a time-bounded variant of Kolmogorov complexity called Kt. In Levin's definition, the complexity of a string x considers both the length of a program p generating x and its running time. Formally, $\mathsf{Kt}(x)$ denotes the minimum over $|p| + \lceil \log t \rceil$, where p is a string such that $U(p) = x$ when U computes for at most t steps over the input string p. It is also possible to consider the conditional Kt complexity of a string x given y, denoted $\mathsf{Kt}(x \mid y)$. In this case, in addition to the input string p, we assume that the universal machine U has random access to the string y. While the $\log t$ term might seem arbitrary at first, it leads to a close relationship between Kt complexity and optimal search algorithms. For this and other reasons, Levin's definition has been highly influential in algorithms and complexity theory (see, e.g., [All01, All92, All17]).

In this work, we put forward a *probabilistic* variant of Kt complexity called pKt. Informally, the new definition is simply Kt in the presence of public randomness, i.e., x has "small" pKt complexity if with probability at least $2/3$ over the randomness r, x has "small" Kt complexity given access to r (think of it as Kt in the "Common Random String" (CRS) model). Formally, for a string $x \in \{0,1\}^*$,

$$\mathsf{pKt}(x) := \min\left\{ k \in \mathbb{N} \;\middle|\; \mathop{\mathbf{Pr}}_{r \sim \{0,1\}^{2^k}} \left[\mathsf{Kt}(x \mid r) \leq k \right] \geq \frac{2}{3} \right\}.$$

Thus bounded pKt complexity means that in the presence of a typical random string r, x has bounded Kt complexity. A key advantage of pKt over Kt is that the former considers randomized computations, which provides a much more suitable setting for cryptography.

Our definition of pKt is inspired by rKt, a variant of Kt complexity considered in [Oli19], and $\mathsf{pK}^{\mathsf{poly}}$, a similar probabilistic notion of Kolmogorov complexity for fixed time bounds considered in [GKLO22] (see Section 2 for the corresponding definitions). More information about probabilistic notions of time-bounded Kolmogorov complexity is available in [LO22].

1.2.2 OWF from BPEXP Lower Bounds via Worst-Case to Average-Case Failure of SoI

We explore the possibility of basing one-way functions on a mild worst-case computational assumption. We consider the widely believed hypothesis that $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$, i.e., that there is a language computable in probabilistic time $2^{\mathsf{poly}(n)}$ that requires super-polynomial size Boolean circuits on all large enough input lengths. (Note that this is significantly weaker than the standard hypothesis from derandomization that $\mathsf{E} \not\subseteq \mathsf{i.o.SIZE}[2^{o(n)}]$ [IW97].) In order to state our result, we first need to review a central notion from the theory of Kolmogorov complexity.

Symmetry of Information (SoI) is a fundamental property of Kolmogorov complexity [ZL70] that has found applications in a number of areas (see, e.g., [SUV17, LV19]). It states that for every pair of strings $x, y \in \{0,1\}^n$,

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

up to an additive factor of order $\pm O(\log n)$ in each equation. In other words, SoI says that: (i) to describe both x and y it is sufficient to describe x optimally without considering y, then describe y optimally assuming access to a description of x; and (ii) there is no significantly better way to describe the pair x, y. While (i) is easily seen to hold, (ii) is non-trivial and states that

$$\mathsf{K}(x, y) \geq \mathsf{K}(x) + \mathsf{K}(y \mid x) - O(\log n).$$

Interestingly, while SoI holds for time-unbounded Kolmogorov complexity, it is known that it fails for Levin's Kt complexity. More precisely, [Ron04] established that for every large n, there is a pair of strings $x, y \in \{0,1\}^n$ such that

$$\mathsf{Kt}(x, y) < \mathsf{Kt}(x) + \mathsf{Kt}(y \mid x) - \omega(\log n).$$

In contrast, whether SoI fails for other measures of time-bounded Kolmogorov complexity remains open. It is known that this must be the case under the existence of one-way functions [LM93, LW95]. More recently, [HIL+23] proved that SoI fails for $\mathsf{pK}^{\mathsf{poly}}$ in a certain average-case sense *if and only if* one-way functions exist. In other words, it is not only sufficient but also necessary to understand SoI in time-bounded Kolmogorov complexity in order to determine the existence of one-way functions. Unfortunately, while SoI fails for Kt (unconditionally), we appear to be far from establishing the failure of SoI for polynomial-time measures such as $\mathsf{pK}^{\mathsf{poly}}$.

This motivates the consideration of the failure of SoI for pKt complexity, which is merely a variant of Kt in the presence of a random string. To investigate this question and its connections to cryptography, we introduce the following statements.

Worst-Case-Asymmetry-pKt: For every constant $c > 0$, if n is large enough then there exist $x, y \in \{0,1\}^n$ such that

$$\mathsf{pKt}(x, y) < \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \log n.$$

Average-Case-Asymmetry-pKt: There is a polynomial-time samplable distribution family $\{\mathcal{D}_n\}$, where each $\mathcal{D}_n$ is supported over $\{0,1\}^n \times \{0,1\}^n$, and a polynomial q such that for every constant $c > 0$ and for every large enough n,

$$\mathop{\mathbf{Pr}}_{(x,y)\sim\mathcal{D}_n}[\mathsf{pKt}(x, y) < \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \log n] \geq \frac{1}{q(n)}.$$

We are now ready to state the main result of this section.

Theorem 1 (Conditional Equivalence Between OWF and Worst-to-Average-Case Failure of SoI)**.** *Under the assumption that* $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$*, the following equivalence holds:*

$$(\mathsf{Worst\text{-}Case\text{-}Asymmetry\text{-}pKt} \implies \mathsf{Average\text{-}Case\text{-}Asymmetry\text{-}pKt}) \iff \exists\mathsf{OWF}$$

Additionally, we have

$$\mathsf{Average\text{-}Case\text{-}Asymmetry\text{-}pKt} \iff \exists\mathsf{OWF}$$

$$\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]} \implies \mathsf{Worst\text{-}Case\text{-}Asymmetry\text{-}pKt}$$

As a consequence of Theorem 1, we can base one-way functions on the worst-case hardness assumption $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$ if the failure of SoI for pKt in the worst case implies its failure on average. Note that the assumption that $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$ is significantly weaker than $\mathsf{NP} \not\subseteq \mathsf{i.o.SIZE[poly]}$.

Corollary 2. *Suppose that* Worst-Case-Asymmetry-pKt $\Rightarrow$ Average-Case-Asymmetry-pKt. *Then* BPEXP $\not\subseteq$ i.o.SIZE[poly] $\Rightarrow \exists$OWF.

In fact, we show that Average-Case-Asymmetry-pKt is equivalent to a *worst-case* version of asymmetry of information with some additive error: $\mathsf{pKt}(x,y) < \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - O(\mathsf{cd}^t(x,y) + \log t)$ for some x, $y \in \{0,1\}^n$, and some $t \geq \mathsf{poly}(n)$, where $\mathsf{cd}^t(x,y) := \mathsf{pK}^t(x,y) - \mathsf{K}(x,y)$ is called *computational depth*. Thus, we can base one-way functions on BPEXP $\not\subseteq$ i.o.SIZE[poly] if the failure of SoI for pKt in the worst case is witnessed by a pair (x,y) of strings with small computational depth. See Appendix A for details.

We note that the relation between worst-case and average-case complexity is well understood in certain settings, such as with respect to computational hardness against non-uniform circuits (see, e.g., [STV01, CLLO21] and references therein). Whether similar "amplification" techniques can be developed in the context of (a)symmetry of information is an intriguing research direction.

It would be very interesting to remove the assumption BPEXP $\not\subseteq$ i.o.SIZE[poly] from Theorem 1. Towards showing that the implication Worst-Case-Asymmetry-pKt $\Rightarrow$ Average-Case-Asymmetry-pKt yields one-way functions, we prove, without the lower bound assumption, that if the implication Worst-Case-Asymmetry-pKt $\Rightarrow$ Average-Case-Asymmetry-pKt is true, then *natural properties* [RR97] against sub-exponential size circuits do not exist. (The latter statement about natural properties is closely related to the existence of one-way functions of quasi-polynomial security.) We refer to Section 4.4 for more details about this result.

1.2.3 OWF via One-Sided to Two-Sided Error Reductions for Approximating pKt

In this section, we consider the more ambitious goal of unconditionally proving the existence of one-way functions. As mentioned above, several recent results have shown that to achieve this goal it is sufficient to prove that certain meta-computational problems about time-bounded Kolmogorov complexity are computationally hard [LP20, LP21, RS21, IRS22, LP22, LP23b].

We upgrade this approach by showing that obtaining a reduction from two-sided average-case approximations of pKt to one-sided average-case approximations of pKt suffices to show the existence of one-way functions. In other words, instead of proving an unconditional lower bound, as in previous papers, designing a reduction between two notions of average-case complexity is enough.

To formalize this result, we will need a couple of definitions. For a function $\tau\colon \mathbb{N} \to \mathbb{N}$, let GapMpKtP$[\tau]$ be the following promise problem $(\mathsf{YES}, \mathsf{NO})$:

$$\begin{aligned} \mathsf{YES} &:= \{(x, 1^s) \mid \mathsf{pKt}(x) \leq s\}\,, \\ \mathsf{NO} &:= \{(x, 1^s) \mid \mathsf{pKt}(x) > s + \tau(|x|)\}\,. \end{aligned}$$

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

$$A(x,1^s) = \begin{cases} 1 & \text{if } \mathsf{pKt}(x) \leq s, \\ 0 & \text{if } \mathsf{pKt}(x) > s + \tau(|x|), \\ \textbf{either } 0 \textbf{ or } 1 & \text{otherwise.} \end{cases}$$

We will also need the following statements.

2-Sided-Error-Approx-pKt: For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, and every polynomial q, there exist a PPT algorithm A and a constant $c > 0$

such that for all sufficiently large n and all $1 \leq s \leq n + O(\log n)$,

$$\mathop{\mathbf{Pr}}_{x \sim \mathcal{D}_n, A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x, 1^s)] \geq 1 - \frac{1}{q(n)},$$

where $\tau(n) = c \cdot \log n$.

Mild-1-Sided-Error-Approx-pKt: There is $\varepsilon > 0$ and a PPT algorithm B such that, for every large enough n, the following holds:

(1) If $x \in \{0,1\}^n$ and $\mathsf{pKt}(x) \leq n^\varepsilon$, then $\mathbf{Pr}_B[B(x) = 1] \geq \frac{2}{3}$.

(2) With probability at least $1/n$ over $x \sim \{0,1\}^n$, $\mathbf{Pr}_B[B(x) = 0] \geq \frac{2}{3}$.

(Note that, while 2-Sided-Error-Approx-pKt considers an arbitrary polynomial-time samplable distribution, Mild-1-Sided-Error-Approx-pKt is only concerned with the uniform distribution.)

A remark about terminology is in order. We note that any PPT algorithm B that accepts every string x with pKt complexity at most n^ε and rejects every string x with pKt complexity at least $0.99n$ satisfies conditions (1) and (2) above. This is because one can show that the overwhelming majority of n-bit strings have pKt complexity close to n. Since (2) significantly relaxes the correctness requirement of B on strings of large complexity, we can think of the algorithm B as a mild approximator for pKt. Regarding the error, if we think of strings x with pKt complexity at most n^ε as positive instances, and strings x with pKt complexity at least $0.99n$ as negative instances, then B makes mistakes only on negative instances. On the other hand, the algorithm A in the statement 2-Sided-Error-Approx-pKt can make mistakes on both negative and positive instances. Therefore, A is a two-sided error algorithm, while B is one-sided. (In Section 1.3 below, we discuss a proof that Mild-1-Sided-Error-Approx-pKt $\Rightarrow$ 2-Sided-Error-Approx-pKt.)

Similarly to the statement ∃OWF defined above, we can consider the weaker statement ∃i.o.OWF, which postulates the existence of one-way functions that are hard to invert on infinitely many input lengths. (The precise definition of ∃i.o.OWF appears in Section 2.)

We are ready to state the main result of this section.

Theorem 3 (Equivalence Between OWF and One-Sided to Two-Sided Error Reductions for pKt)**.** *The following equivalence holds:*

$$(\text{2-Sided-Error-Approx-pKt} \Rightarrow \text{Mild-1-Sided-Error-Approx-pKt}) \iff \exists\text{i.o.OWF}$$

Additionally, we have[1]

$$\neg\,\text{2-Sided-Error-Approx-pKt} \iff \exists\text{i.o.OWF}$$

$\neg$Mild-1-Sided-Error-Approx-pKt holds unconditionally

Previously, in a celebrated result, Liu and Pass [LP21] proved that one can base (infinitely-often) OWF on the assumption that EXP $\not\subseteq$ BPP if and only if there is a reduction from computing Kt on average with zero error under the uniform distribution to computing Kt on average with two-sided error. In contrast, as a consequence of Theorem 3, (infinitely-often) one-way functions exist *unconditionally* if there is a probabilistic average-case polynomial-time reduction from mildly

[1] We state the result in this way to allow for a direct comparison with prior work.

approximating pKt with one-sided error to approximating pKt with two-sided error.[2] Moreover, the existence of a quasi-polynomial-time reduction suffices, due to the proof of a slightly stronger result in Section 1.3 below. Therefore, this result shows that the hardness needed for one-way functions comes from the difference between 1-sided and 2-sided error, and not from the assumption that $\mathsf{EXP} \not\subseteq \mathsf{BPP}$.

Theorem 3 suggests a tantalizing possibility: the existence of one-way functions would follow from the design of an efficient reduction involving different notions of average-case complexity for the same computational task.[3] For instance, a reduction of this nature is known for $\mathsf{UP} \cap \mathsf{coUP}$ and can be constructed from an instance checker [HS22].

Given the stakes, it is natural to consider potential barriers that one might need to overcome when attempting to obtain such a reduction. In our next result, we establish that a reduction cannot be obtained through the use of *relativizing* techniques (in the sense of [BGS75]). More precisely, we consider a scenario where an oracle $\mathcal{O}$ is available to all computations, i.e., the universal machine defining pKt, the samplers defining the polynomial-time samplable distributions, and the algorithms that attempt to approximate $\mathsf{pKt}^{\mathcal{O}}$. We prove the following result.

Theorem 4. *There exists an oracle* $\mathcal{O} \in \mathsf{PSPACE}$ *relative to which* 2-Sided-Error-Approx-pKt *is true, but* Mild-1-Sided-Error-Approx-pKt *is false.*

In other words, techniques that hold in the presence of an arbitrary oracle cannot be used to obtain a one-sided to two-sided error reduction for approximating pKt. Indeed, Theorem 4 provides an even stronger PSPACE-relativization barrier in the sense of [HLR23]. We refer to Section 5.4 for further discussions on this barrier.

We remark that many techniques employed in the investigation of time-bounded Kolmogorov complexity and meta-complexity relativize, but not all of them (see [Hir22a] for a striking recent example). For this reason, we view Theorem 4 more as a guiding principle than a strong negative result suggesting that one should not investigate such reductions.

1.3 Techniques

In this section, we discuss the proofs of Theorem 1 and Theorem 3, which rely on some intermediate results that might be of independent interest. Since our arguments make use of a number of techniques from time-bounded Kolmogorov complexity and rely on several ideas from previous papers, we only provide a high-level exposition, referring to the main body of the paper for further technical details. After this discussion, we summarise some advantages of pKt complexity over other time-bounded Kolmogorov complexity measures.

OWF and Worst-Case to Average-Case Failure of SoI for pKt *(Theorem 1)*. First, a careful adaptation of the techniques from [HIL+23] and [LW95] allows us to establish an equivalence between the failure of SoI for pKt on average and the existence of one-way functions.

Theorem 5. *The following equivalence holds:*

$$\text{Average-Case-Asymmetry-pKt} \iff \exists\mathsf{OWF}.$$

[2] It is also possible to define zero-error and one-sided error (as opposed to mildly one-sided error) average-case notions of approximating pKt. However, employing them in Theorem 3 would make the result weaker (i.e., it is easier to obtain a mild one-sided error algorithm from a two-sided error algorithm than to obtain a zero-error algorithm from a two-sided error algorithm). For completeness, we provide a comprehensive discussion of these notions in Section 3.2.

[3] In a sense, one can think of the result as an *algorithmic* approach to establish the existence of one-way functions.

However, the more complex notion of pKt introduces additional technicalities, as we explain next. Using ideas from [LW95], it can be shown that if SoI for $\mathsf{K}^{\mathsf{poly}}$ holds on average, then for an average image y of the candidate one-way function, one can upper bound the $\mathsf{K}^{\mathsf{poly}}$ complexity of most (say, at least $1/2$) of the pre-images x of y as follows.

$$\mathsf{K}^{\mathsf{poly}}(x \mid y) \lesssim \log |f^{-1}(y)| =: k.$$

Then by defining a sampling procedure that randomly picks a program of size k and run it for *polynomially many* steps (conditional on y), x can be obtained with probability roughly 2^{-k}, which is $1/|f^{-1}(y)|$. Since this holds for at least $|f^{-1}(y)|/2$ pre-images, it follows that we obtain *some* pre-image with decent probability. (Contrapositively, if $\exists\mathsf{OWF}$ then asymmetry of information for K^t holds on average.)

Using similar ideas, we can show that if *average-case* SoI for pKt holds, then for an average image y of the candidate one-way function, one can upper bound the pKt complexity of most of its pre-images x as follows.

$$\mathsf{pKt}(x \mid y) \lesssim \log |f^{-1}(y)| =: k.$$

Now since we can only upper bound the pKt complexity of a pre-image (instead of $\mathsf{pK}^{\mathsf{poly}}$), the previously mentioned sampling procedure no longer works. This is because, by the definition of pKt, the above only implies that for a uniformly random string $r \in \{0,1\}^{2^k}$, there are integers s and t such that $s + \log t \leq k$ and there is a program of size s that, given y and r, runs in time t and outputs x. In particular, t may not be bounded by $\mathsf{poly}(n)$ in this case.

To cope with this issue, we further observe that for most of the pre-images x, *with high probability* over $r \sim \{0,1\}^{2^k}$, any program that generates x, given r and y, must be large in the sense that

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

This can by shown by using a counting argument. Now given the above, we can say that for a uniformly random string $r \in \{0,1\}^{2^k}$, there are integers s and t such that $s + \log t \leq k$ and there is a program of size s that, given y and r, runs in time t and outputs x. Moreover, s must be at least $k - O(\log n)$. Note that this implies $t = \mathsf{poly}(n)$.

At this point, it seems we can carry out the previous argument and show that we can obtain a pre-image of y in polynomial time, by randomly picking a string $r \in \{0,1\}^{2^k}$, a program $p \in \{0,1\}^{\leq k}$ and running p for $\mathsf{poly}(n)$ steps, while given oracle access to r and y. However, there is one more issue. To perform this sampling procedure, we need to pick a uniformly random string r of length 2^k, which is not necessarily $\mathsf{poly}(n)$. Fortunately, since we only need to run our programs for $\mathsf{poly}(n)$ steps, we do not need to keep the entire random string. Instead, we can generate random bits on-the-fly and maintain the same behavior of our program as if it were running with a pre-generated random string.

Next, complementing Theorem 5, we explore the failure of SoI for pKt in the worst case. We are able to show that the latter holds under a worst-case circuit lower bound assumption for a language in BPEXP. This is a key result which allows us to link a worst-case lower bound in the complexity-theoretic regime to the cryptographic regime in Theorem 1.

Theorem 6. *If* $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$ *then* $\mathsf{Worst\text{-}Case\text{-}Asymmetry\text{-}pKt}$ *holds.*

To our knowledge, this is the first result showing the failure of symmetry of information for a probabilistic notion of time-bounded Kolmogorov complexity under a lower bound assumption in the complexity-theoretic regime, as opposed to a lower bound assumption in the cryptographic regime (e.g., [LW95, HIL+23]).

In Theorem 6 the goal is to construct a pair (x, y) of n-bit strings witnessing the asymmetry of information of this pair with respect to pKt complexity. Inspired by the unconditional construction of such a pair with respect to Kt complexity [Ron04], we attempt a generalization of the argument to the probabilistic setting of pKt. The construction of [Ron04] relies on a simple (deterministic) exhaustive search that defines an appropriate pair (x, y) and certifies the necessary Kt bounds for the strings. Unfortunately, in a probabilistic setting, it is unclear if a similar (probabilistic) exhaustive search specifies a *canonical* pair (x, y) with the desired properties, which is needed in order to obtain upper bounds on probabilistic Kolmogorov complexity.

In more detail, a key step in the proof from [Ron04] is to compute given a string y the set $S_y^{\mathsf{Kt}} \subseteq \{0,1\}^n$ of strings of conditional Kt complexity at most s, for some threshold s. This is done in time $O(2^s)$ using a deterministic algorithm A that decides whether $\mathsf{Kt}(x \mid y) \leq s$ for a given x. In our case, we are only able to use a corresponding *randomized* algorithm B that checks whether $\mathsf{pKt}(x \mid y) \leq s$ or $\mathsf{pKt}(x \mid y) \geq 2s$, with no guarantee on the remaining instances. Unfortunately, the exhaustive search over all strings performed with the help of B will not produce a fixed set S_y^{pKt}, since its behaviour outside the promise region means that on different executions a different set of strings could be added to S_y^{pKt}, according to the internal randomness of B.

We attempt to fix this issue under the assumption that $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$, which is sufficient for the construction of a non-trivial *pseudodeterministic* pseudorandom generator. The latter allows us to perform an exhaustive search over probabilistic algorithms in a way that produces a canonical pair (x, y) with high probability.

It turns out that this is not quite enough to finish the proof, because under the weak lower bound assumption $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$ we are only able to construct strings of conditional pKt complexity larger than s in time roughly $2^{2^{o(s)}}$. This is an important issue that is not present in [Ron04]. To address this difficulty, we show via a more sophisticated *iterative process* for constructing strings that symmetry of information is indeed violated for *some pair* (x, y) specified during the process. Since this is somewhat delicate and technical to describe, we refer the reader to Section 4.2.

It is easy to see that Theorem 1 follows from Theorem 5 and Theorem 6 (see Section 4.3).[4]

OWF and 1-Sided to 2-Sided Error Reductions for Approximating pKt (Theorem 3). To establish this result, we first obtain an equivalence between the existence of (infinitely-often) one way functions and the average-case hardness of approximating pKt complexity with two-sided error.

Theorem 7. *The following equivalence holds:*

$$\neg\, \mathsf{2\text{-}Sided\text{-}Error\text{-}Approx\text{-}pKt} \iff \exists \mathsf{i.o.OWF}.$$

The proof of Theorem 7 makes use of a connection between one-way functions and the hardness of approximating (time-unbounded) Kolmogorov complexity K [IRS22], which can be adapted to pKt by investigating the relation between K and pKt for strings generated by a polynomial-time samplable distribution. In more detail, the argument in [IRS22] relied on the use of the *coding theorem* for time-unbounded Kolmogorov complexity. Here, we extend their approach and employ a recently discovered efficient coding theorem for $\mathsf{pK}^{\mathsf{poly}}$ [LOZ22], which also applies to pKt.

[4] We note that the proof that there are strings (x, y) for which symmetry of information fails is by constructing such strings for which the running time t of the pKt witness is *exponential*. On the other hand, the proof that average-case symmetry of information implies that we can break any one-way function uses the fact that on any samplable distribution, with high probability on the sample y, the pKt witness has *polynomial* running time. Bridging this gap is a very interesting research direction.

Next, we establish an unconditional lower bound against probabilistic algorithms that mildly approximate pKt on average with one-sided error.

Theorem 8. Mild-1-Sided-Error-Approx-pKt *is false. Moreover, the corresponding lower bound also holds against randomized algorithms running in time* $n^{\mathsf{poly}(\log n)}$.

Theorem 8 highlights an important difference between pKt and Kt that plays a central role in the proof of Theorem 3: we can establish *unconditional* complexity lower bounds for computing pKt, while the same result is unknown for Kt. The proof of Theorem 8 modifies an argument employed to show a complexity lower bound of a similar nature for estimating rKt complexity [Oli19]. It can be described as an indirect diagonalization that heavily relies on techniques from computational pseudorandomness. The proof relies on the following key lemmas:

1. If pKt can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$, then $\mathsf{BPE} \subseteq \mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$.

2. If pKt can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$, then $\mathsf{PSPACE} \subseteq \mathsf{BPTIME}[n^{\mathsf{poly}(\log n)}]$. In particular, under this assumption $\mathsf{DSPACE}[2^{n^{o(1)}}] \subseteq \mathsf{BPE}$. (We observe that this step is problematic in the setting of Kt, as it relies on techniques from pseudorandomness whose underlying algorithms are *randomized*.)

   These two lemmas, which require the analysis of different pseudorandom generators and of the time-bounded Kolmogorov complexity of their output strings, use that pKt is both "probabilistic" and "exponential" (as opposed to Kt, which is "deterministic", and $\mathsf{pK}^{\mathsf{poly}}$, which is "polynomial").

3. There is a language in $\mathsf{DSPACE}[2^{n^{o(1)}}] \setminus \mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$.

   The proof of this third lemma uses a standard diagonalization technique.

Assuming Mild-1-Sided-Error-Approx-pKt holds, we obtain from Items 1 and 2 that $\mathsf{DSPACE}[2^{n^{o(1)}}] \subseteq \mathsf{BPE} \subseteq \mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$, in contradiction with Item 3.

Finally, Theorem 3 easily follows from Theorem 7 and Theorem 8 (see Section 5.3).

We note in passing that Theorem 7 can be used to give a reduction from the task of approximating pKt with two-sided error over *any polynomial-time samplable distribution* to the task of approximating pKt with one-sided error over the *uniform distribution* and, in particular, can be used to prove

$$\text{Mild-1-Sided-Error-Approx-pKt} \Rightarrow \text{2-Sided-Error-Approx-pKt}.$$

Indeed, under Mild-1-Sided-Error-Approx-pKt it is not hard to show that every candidate cryptographic pseudorandom generator can be broken. Since the latter is equivalent to the non-existence of (infinitely-often) one-way functions [HILL99] (i.e., $\neg\exists$i.o.OWF), we immediately derive 2-Sided-Error-Approx-pKt from Theorem 7.

The benefits of pKt complexity. We summarize here some advantages of pKt over other time-bounded Kolmogorov complexity measures:

- An optimal coding theorem is known to hold unconditionally for $\mathsf{pK}^{\mathsf{poly}}$ [LOZ22] and for pKt. This is a key principle in Kolmogorov complexity and a very useful tool in applications. The same result is not known to hold unconditionally for other complexity measures.

- A central aspect in recent investigations of meta-complexity and its applications is the advice complexity and time-bounded Kolmogorov complexity measure associated with the reconstruction procedure of pseudorandom generators. When using $\mathsf{pK}^{\mathsf{poly}}$ and pKt, existing generators offer superior bounds, which allow results to be more easily established in the polynomial-time regime as opposed to the quasi-polynomial time regime and above.
- In contrast to the situation for $\mathsf{pK}^{\mathsf{poly}}$ and other polynomial-time complexity measures, we have unconditional super-polynomial complexity lower bounds for approximating pKt (as in Theorem 8).
- The unconditional failure of symmetry of information is only known to hold for an exponential-time measure (Kt), which suggests that it will be easier to resolve this question for pKt and rKt as opposed to polynomial-time measures such as $\mathsf{pK}^{\mathsf{poly}}$. Indeed, showing the failure of symmetry of information for certain polynomial-time measures would imply that $\mathsf{P} \neq \mathsf{NP}$ [Hir22b], while no consequence of a similar form is known in the case of pKt.

The first two bullets highlight advantages of the probabilistic measures $\mathsf{pK}^{\mathsf{poly}}$ and pKt over deterministic complexity measures such as K^t and Kt, while the last two bullets highlight the advantages of pKt over polynomial-time measures (such as $\mathsf{pK}^{\mathsf{poly}}$). We also note that, while pKt is closely related to Kt, super-polynomial complexity lower bounds are not known for the problem of computing Kt. Overall, the aforementioned features make pKt complexity an attractive complexity measure for the investigation of connections between one-way functions and the theory of time-bounded Kolmogorov complexity.

1.4 Directions and Open Problems

There are a few directions to be explored that would advance this research program. Moreover, to our knowledge the concrete problems listed below might all be within the reach of existing techniques.

Theorem 6 establishes the failure of SoI for pKt under a circuit lower bound assumption. In contrast, as mentioned above, it is known unconditionally that SoI fails for Kt. Can the same be done for pKt? If not, can we connect this question to major open problems about the power of randomness in computation?

Easiness assumptions can often be used to establish symmetry of information for different complexity measures [Hir22b, GK22, GKLO22]. Is it possible to prove that if $\mathsf{BPEXP} \subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$ then SoI holds for pKt on infinitely many input lengths (i.e., $\neg\mathsf{Worst\text{-}Case\text{-}Asymmetry\text{-}pKt}$)? This would allow us to strengthen Theorem 3 and obtain the following equivalence:

$$\begin{array}{c}(\mathsf{Worst\text{-}Case\text{-}Asymmetry\text{-}pKt} \implies \mathsf{Average\text{-}Case\text{-}Asymmetry\text{-}pKt}) \\ \Updownarrow \\ (\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE}[\mathsf{poly}] \implies \exists\mathsf{OWF})\end{array}$$

In other words, we would obtain that connecting the failure of SoI for pKt in the worst case and in the average case is not only sufficient but also necessary to base one-way functions on a worst-case non-uniform lower bound for BPEXP. Given the techniques developed in previous papers, the main difficulty in showing SoI for pKt from the assumption $\mathsf{BPEXP} \subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$ seems to be that it states a non-uniform upper bound instead of a uniform one.

We obtained an unconditional lower bound against probabilistic quasi-polynomial time algorithms for the task of estimating pKt complexity (Theorem 8). If one could show a sub-exponential

time lower bound, this would relax even more the running time of the reduction from two-sided approximation to mild one-sided approximation needed to establish the existence of one-way functions in Theorem 3.

Finally, there would be significant consequences to complexity theory and cryptography if one could show that 2-Sided-Error-Approx-pKt $\Rightarrow$ Mild-1-Sided-Error-Approx-pKt (Theorem 3). Are there additional difficulties that must be overcome beyond the relativization barrier established in Theorem 4?

Acknowledgements. We are thankful to Eric Allender for suggesting the investigation of the failure of symmetry of information for Kt and its connections to cryptography in light of the results in [HIL+23]. We also appreciate the anonymous reviewers for their valuable feedback on the presentation. This work received support from the Royal Society University Research Fellowship URF\R1\191059; the UKRI Frontier Research Guarantee Grant EP/Y007999/1; and the Centre for Discrete Mathematics and its Applications (DIMAP) at the University of Warwick.

2 Preliminaries

For a probability distribution $\mathcal{D}$ and a string $x \in \{0,1\}^*$, we use $\mathcal{D}(x)$ to denote the probability that x is sampled from $\mathcal{D}$.

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

2.1 One-Way Functions

Let FP denote the set of functions that can be computed in deterministic polynomial time.

Definition 9 (One-Way Function)**.** We say that a function $f = \{f_n\} \in \mathsf{FP}$, where $f_n \colon \{0,1\}^n \to \{0,1\}^{\mathsf{poly}(n)}$, is a one-way function if for every probabilistic polynomial time (PPT) algorithm A, and for every large enough n,

$$\Pr_{A,\, x\sim\{0,1\}^n}[A(f(x)) \in f^{-1}(f(x))] \leq \frac{1}{n^{\omega(1)}}.$$

We can also consider an infinitely-often secure variant of one-way functions.

Definition 10 (Infinitely-Often One-Way Function)**.** We say that a function $f = \{f_n\} \in \mathsf{FP}$, where $f_n \colon \{0,1\}^n \to \{0,1\}^{\mathsf{poly}(n)}$, is an infinitely-often one-way function if for every probabilistic polynomial time (PPT) algorithm A, there is an infinite set $S_A \subseteq \mathbb{N}$ such that for every $n \in S_A$,

$$\Pr_{A,\, x\sim\{0,1\}^n}[A(f_n(x)) \in f_n^{-1}(f_n(x))] \leq \frac{1}{n^{\omega(1)}}.$$

Note that the set S_A of inputs can depend on A. It is possible to show that this definition implies that for every k there is an infinite set $S_k \subseteq \mathbb{N}$ such that every PPT algorithm A that runs in time $O(n^k)$ only succeeds to invert f with negligible probability on large input lengths $n \in S_A$. (This is because one can define a "universal" PPT algorithm B that runs every algorithm of time bound n^{k+1} and description length $\log n$ while trying to invert f_n.)

Theorem 11 ([IL90, IL89])**.** *Assume infinitely-often one-way functions do not exist. Let $\{\mathcal{D}_n\}_n$ be a family of polynomial-time samplable distributions, and let q be any polynomial. There exists a probabilistic polynomial-time algorithm B such that for all $n \in \mathbb{N}$,*

$$\Pr_{x\sim\mathcal{D}_n,B}\left[\frac{\mathcal{D}_n(x)}{2} \leq B(1^n, x) \leq \mathcal{D}_n(x)\right] \geq 1 - \frac{1}{q(n)}.$$

2.2 Kolmogorov Complexity

We fix a universal Turing machine U. We write $U(p)$ to indicate the output of U on an input string p, where p is written on the input tape. For a string y, we write U^y to indicate that U has random access to y. In other words, y is written on an oracle tape, and U can query the i-th bit of y by specifying the index i on a query tape.

Definition 12 (Kt [Lev84])**.** For $x, y \in \{0,1\}^*$, the *time-bounded Kolmogorov complexity of x given y* is defined as

$$\mathsf{Kt}(x \mid y) := \min_{p \in \{0,1\}^*,\, t \in \mathbb{N}} \Big\{ |p| + \lceil \log t \rceil \;\mid\; U^y(p) \text{ outputs } x \text{ within } t \text{ steps} \Big\}.$$

Definition 13 (pK^t [GKLO22])**.** Let $x, y \in \{0,1\}^*$ and $t \in \mathbb{N}$. The *probabilistic t-time-bounded Kolmogorov complexity of x given y* is defined as

$$\mathsf{pK}^t(x \mid y) := \min \left\{ k \in \mathbb{N} \;\middle|\; \Pr_{r \sim \{0,1\}^t} \left[\exists\, p \in \{0,1\}^k \text{ s.t. } U^{y,r}(p) \text{ outputs } x \text{ within } t \text{ steps} \right] \geq \frac{2}{3} \right\}.$$

We recall some useful results regarding Kolmogorov complexity.

Lemma 14 (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 for all $n \in \mathbb{N}$,*

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

Theorem 15 (Coding Theorem [Lev74])**.** *Let $\mathcal{E}$ be a distribution whose cumulative distribution function can be computed by some program p. Then for every $x \in \mathsf{Support}(\mathcal{E})$,*

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

Theorem 16 (Efficient Coding Theorem [LOZ22])**.** *For every distribution family $\{\mathcal{D}_n\}_n$ samplable in polynomial time, where each $\mathcal{D}_n$ is supported over $\{0,1\}^n$, there exists a polynomial p such that for every $x \in \mathsf{Support}(\mathcal{D}_n)$,*

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

Theorem 17 ([HIL+23])**.** *The following are equivalent.*

1. *There exist no (resp. infinitely-often) one-way functions.*
2. **(Average-Case Conditional Coding)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_{n \in \mathbb{N}}$ supported over $\{0,1\}^n \times \{0,1\}^n$ and every polynomial q, there exists a polynomial p such that for infinitely many (resp. all) n,*

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

2.3 Pseudorandomness

We will need the following results in pseudorandomness.

Lemma 18 ([IW97, OS17b])**.** *Suppose* $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$ *(resp.* $\mathsf{BPEXP} \not\subseteq \mathsf{SIZE[poly]}$*). Then for every* $\varepsilon > 0$*, there is a pseudorandom generator mapping* $r := s^{\varepsilon}$ *bits to* s *bits, computable in pseudodeterministic time* $2^{O(r)}$*, that fools circuit of size* s *with error* $1/s$*, for all but finitely many (resp. infinitely many)* s*.*

Theorem 19 ([BFNW93, KvM02])**.** *For every constant* $0 < \lambda < 1$*, there is a pseudorandom generator* $\left\{G_n^{(-)} \colon \{0,1\}^{n^\lambda} \to \{0,1\}^n\right\}_n$ *such that the following holds. Let* $f \colon \{0,1\}^* \to \{0,1\}$*.*

1. G_n^f *can be can be computed in deterministic time* $\exp\big(O(n^\lambda)\big)$ *given oracle access to* f *on inputs of length at most* n^λ*.*
2. *For every function* $D \colon \{0,1\}^n \to \{0,1\}$*, if*

$$\left| \mathop{\mathbf{Pr}}_{r\sim\{0,1\}^{n^\lambda}} \Big[D(G_n^f(r)) = 1\Big] - \mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n} [D(x) = 1] \right| \geq \frac{1}{O(n)}$$

for every large enough n*, then there is a sequence* $\{C_n\}_n$ *of polynomial-size* D*-oracle circuits that computes* f *on input length* n*.*

Theorem 20 ([IW97])**.** *For every constant* $0 < \lambda < 1$*, there is a pseudorandom generator* $\left\{\mathsf{IW}_n^{(-)} \colon \{0,1\}^{n^\lambda} \to \{0,1\}^n\right\}_n$ *such that the following holds. Let* $f \colon \{0,1\}^* \to \{0,1\}$ *be a function that is both random self-reducible and downward self-reducible.*

1. IW_n^f *can be can be computed in deterministic time* $\exp\big(O(n^\lambda)\big)$ *given oracle access to* f *on inputs of length at most* n^λ*.*
2. *For every oracle* $\mathcal{O}$*, if there is a probabilistic* $\mathcal{O}$*-oracle algorithm* D *with running time* $t(n)$ *such that*

$$\left| \mathop{\mathbf{Pr}}_{r\sim\{0,1\}^{n^\lambda},D} \Big[D(\mathsf{IW}_n^f(r)) = 1\Big] - \mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n,D} [D(x) = 1] \right| \geq \frac{1}{O(n)}$$

for every large enough n*, then there is a randomized* $\mathcal{O}$*-oracle algorithm with running time* $\mathsf{poly}(n) \cdot t(n)$ *that on every input* x *outputs* $f(x)$ *with high probability.*

Theorem 21 ([TV07])**.** *There is a language* $L_{\mathsf{TV}} \in \mathsf{DSPACE}[O(n)]$ *that is* PSPACE*-hard, random self-reducible, and downward self-reducible.*

2.4 Natural Properties

Definition 22 (Natural Properties)**.** For a size function $s \colon \mathbb{N} \to \mathbb{N}$, a dense prBPP-property useful against $\mathsf{SIZE}[s]$ is a probabilistic polynomial-time algorithm P such that the following holds for every N.

- With probability at least $1/N^{O(1)}$ over $x \sim \{0,1\}^N$, $\mathbf{Pr}_P[P(x) = 1] \geq 2/3$.
- For every $x \in \{0,1\}^N$, viewed as the truth table of a Boolean function, if x has circuit complexity at most $s(\log N)$, then $\mathbf{Pr}_P[P(x) = 0] \geq 2/3$.

Theorem 23 ([RR97])**.** *If there exists a constant* $\varepsilon > 0$ *such that dense* prBPP*-properties useful against* $\mathsf{SIZE}[2^{n^\varepsilon}]$ *exist, then infinitely-often one-way functions do not exist.*

3 pKt: Probabilistic Levin Complexity

In this section, we formally define pKt and state some useful properties which will be used in the proofs of our results.

Also, we discuss in Section 3.2 different notions of average-case complexity for approximating pKt, which will justify the notion of *mild-one-sided error* average-case complexity for approximating pKt.

We start with the definition of pKt.

Definition 24 (pKt)**.** For $x \in \{0,1\}^*$ and $0 < \lambda \leq 1$, the *λ-probabilistic time-bounded Kolmogorov complexity* of x, denoted by $\mathsf{pKt}_\lambda(x)$, is defined to be the minimum $k \in \mathbb{N}$ such that with probability at least λ over $r \sim \{0,1\}^{2^k}$, there exist a program $p \in \{0,1\}^*$ and a time bound $t \in \mathbb{N}$ that satisfy $|p| + t \leq k$ and $U^r(p)$ outputs x within t steps. Equivalently

$$\mathsf{pKt}_\lambda(x) := \min\left\{ k \in \mathbb{N} \;\middle|\; \Pr_{r \sim \{0,1\}^{2^k}}[\mathsf{Kt}(x \mid r) \leq k] \geq \lambda \right\}.$$

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

This definition can be extended to *conditional* Kolmogorov complexity in the natural way. More specifically, in $\mathsf{pKt}(x \mid y)$ the machine U is also given oracle access to the string y.

3.1 Useful Properties of pKt

Proposition 25. *There is a universal constant $b > 0$ such that for every $x, y \in \{0,1\}^*$ and $t \in \mathbb{N}$,*

1. $\mathsf{pKt}(x \mid y) \leq \mathsf{pK}^t(x \mid y) + \log t$*, and*
2. $\mathsf{K}(x \mid y) \leq \mathsf{pKt}(x \mid y) + b \log |x|$.

Proof. Item 1 follows immediately from the definitions of pK^t (Definition 13) and pKt (Definition 24).

Next, we show Item 2. Fix any $x \in \{0,1\}^n$ and $y \in \{0,1\}^*$. Let $k := \mathsf{pKt}(x \mid y)$. Consider the following procedure for sampling x given n and y.

1. Pick a uniformly random $k' \sim [O(n)]$.
2. Pick a uniformly random $r \sim \{0,1\}^{2^{k'}}$.
3. Pick a uniformly random $s' \sim [O(n)]$.
4. Pick a uniformly random $p \sim \{0,1\}^{s'}$.
5. Run $U^{y,r}(p)$ for at most 2^k steps and return its output.

By the definition of pKt, it is easy to see that the above procedure outputs x with probability at least

$$\frac{1}{O(n)} \cdot \frac{2}{3} \cdot \frac{1}{O(n)} \cdot \frac{1}{2^k} = \frac{1}{2^k \cdot \mathsf{poly}(n)}.$$

Then by the coding theorem for K (Theorem 15), we have

$$\mathsf{K}(x \mid y) \leq k + O(\log n),$$

as desired. □

Next, we state a relation between Kt and pKt. The proof of this fact follows by an easy adaptation of results from [GK22, Appendix A.2].

Proposition 26. *If* $\mathsf{E} \not\subseteq \mathsf{i.o.NSIZE}\left[2^{\Omega(n)}\right]$*, then there is a constant* $c > 0$ *such that for every string* $x \in \{0,1\}^*$*,* $\mathsf{pKt}(x) \leq \mathsf{Kt}(x) \leq c \cdot \mathsf{pKt}(x)$*.*

Note that the relation between Kt and pKt from Proposition 26 is not as tight as the relation between $\mathsf{K}^{\mathsf{poly}}$ and $\mathsf{pK}^{\mathsf{poly}}$ described in [GK22, Appendix A.2]. This is due to the polynomial time overhead in the simulation, which can incur a constant factor in the description complexity due to the $\log t$ term. In particular, for this reason, we cannot easily derive the failure of SoI for pKt from the failure of SoI for Kt under a lower bound assumption.

Lemma 27 (Success Amplification; following [GKLO22])**.** *For any string* $x \in \{0,1\}^n$*,* $y \in \{0,1\}^*$*, and* $0 \leq \alpha < \beta \leq 1$*, we have*

$$\mathsf{pKt}_\beta(x \mid y) \leq \mathsf{pKt}_\alpha(x \mid y) + O(\log(q/\alpha) + \log n),$$

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

Proof. Suppose $\mathsf{pKt}_\alpha(x \mid y) = k \leq 2n$. Then by definition, we have with probability at least α over $r \sim \{0,1\}^{2^k}$, there exist a program $p_r \in \{0,1\}^*$ and a time bound $t_r \in \mathbb{N}$ that satisfy $|p_r| + \log t_r \leq k$ and $U^r(p, y)$ outputs x within t steps. Call such an r *good*. After sampling ℓ independent $r^{(1)}, r^{(2)}, \ldots, r^{(\ell)} \in \{0,1\}^{2^k}$, the probability that no $r^{(i)}$ is good is at most $(1-\alpha)^\ell \leq e^{-\alpha\ell}$, which can be made at most $1-\beta$ by choosing $\ell = q/\alpha$. It follows that, with probability at least β over a random $w := \left(r^{(1)}, r^{(2)}, \ldots, r^{(q/\alpha)}\right) \in \{0,1\}^{2^k \cdot q/\alpha}$, there exists an index $1 \leq i \leq q/\alpha$ (which can be described with at most $\lceil \log(q/\alpha) \rceil$ bits) such that for some program $p_{r^{(i)}}$ and $t_{r^{(i)}}$ that satisfy

$$|p_{r^{(i)}}| + \log t_{r^{(i)}} \leq k,$$

and $U^{r^{(i)}}(p_{r^{(i)}}, y)$ outputs x within $t_{r^{(i)}}$ steps. For such w, we can construct a program p_w of size at most $|p_{r^{(i)}}| + O(\log(q/\alpha) + \log k)$ such that $U^w(p_w)$ outputs x within $t_w := t_{r^{(i)}} \cdot O(\log |w|)$ steps. It follows that $\mathsf{pKt}_\beta(x \mid y) \leq k + O(\log(q/\alpha) + \log n)$. □

We define a gap version of the minimum pKt problem, $\mathsf{GapMpKtP}$, which can be viewed as the decision version of the problem of approximating pKt. For $\tau \colon \mathbb{N} \to \mathbb{N}$, let $\mathsf{GapMpKtP}[\tau]$ be the following promise problem $(\mathsf{YES}, \mathsf{NO})$.

$$\begin{aligned} \mathsf{YES} &:= \{(x, 1^s) \mid \mathsf{pKt}(x) \leq s\}, \\ \mathsf{NO} &:= \{(x, 1^s) \mid \mathsf{pKt}(x) > s + \tau(|x|)\}. \end{aligned}$$

Lemma 28. *There is a constant* $c > 0$ *such that for every* $\tau(n) \geq c \log n$*,* $\mathsf{GapMpKtP}[\tau] \in \mathsf{prBPE}$*.*

Proof. Let $c > 0$ a a sufficiently large constant determined later, and let $\tau(n) := c \log n$. First consider the following problem where we are given $x \in \{0,1\}^n$ and $s \in \mathbb{N}$ and asked to decide whether:

1. $\mathsf{pKt}_{2/3}(x) \leq s$, or
2. $\mathsf{pKt}_{1/3}(x) > s$.

We claim that the above problem can be solved in randomized time $2^{O(n)}$. Indeed, consider the following algorithm:

1. Pick $r \sim \{0,1\}^{2^s}$.
2. For all programs $p \in \{0,1\}^*$ and running times $t \in \mathbb{N}$ such that $|p| + \log t \leq s$, check if $U^r(p)$ outputs x within t steps. Accepts if and only if there exist such p and t that pass the check.

It is clear that the running time of the above algorithm is $2^{O(n)}$, and the correctness also follows easily from the definition of pKt. Indeed, if $\mathsf{pKt}_{2/3}(x) \leq s$, then the algorithm accepts with probability at least $2/3$ and if $\mathsf{pKt}_{1/3}(x) > s$, the algorithm accepts with probability at most $1/3$.

Now we claim that the above algorithm also solves the following problem of deciding whether:

1. $\mathsf{pKt}_{2/3}(x) \leq s$, or
2. $\mathsf{pKt}_{2/3}(x) > s + \tau(n)$.

This is because by Lemma 27, we have

$$\mathsf{pKt}_{2/3}(x) \leq \mathsf{pKt}_{1/3}(x) + O(\log m).$$

Hence, if $\mathsf{pKt}_{2/3}(x) > s + \tau(n)$, then we get $\mathsf{pKt}_{1/3}(x) > s + \tau(n) - O(\log n) > s$, provided that c is a sufficiently large constant. □

The following lemma will be convenient for us.

Lemma 29. *There is a probabilistic algorithm B such that given $x \in \{0,1\}^m$ and $y \in \{0,1\}^{\leq 2^m}$, $B(x,y)$ runs in time $2^{O(m)}$, rejects (with high probability) if $\mathsf{pKt}(x \mid y) < m/2$ and accepts (with high probability) if $\mathsf{pKt}(x \mid y) \geq 2m/3$.*

Proof Sketch. The proof is essentially the same as that of Lemma 28. We first consider the following problem where we are given $x \in \{0,1\}^m$ and $y \in \{0,1\}^{\leq 2^m}$ and asked to decide whether:

1. $\mathsf{pKt}_{2/3}(x \mid y) \leq m/2$, or
2. $\mathsf{pKt}_{1/3}(x \mid y) > m/2$.

As in the proof of Lemma 28, we can then show that there is a prBPE-algorithm that solves the above problem. Finally, we observe that the same algorithm solves the problem stated in the lemma, since by using Lemma 27, if $\mathsf{pKt}_{2/3}(x \mid y) > 2m/3$, then $\mathsf{pKt}_{1/3}(x \mid y) > m/2$. □

3.2 Notions of Average-Case Tractability for Approximating pKt

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

$$A(x,1^s) = \begin{cases} 1 & \text{if } \mathsf{pKt}(x) \leq s, \\ 0 & \text{if } \mathsf{pKt}(x) > s + \tau(|x|), \\ \textbf{either } 0 \textbf{ or } 1 & \text{otherwise.} \end{cases}$$

Definition 30 (2-Sided-Error-Approx-pKt)**.** We say that pKt can be approximated on average with *two-sided error* if for every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, and every polynomial q, there exist a PPT algorithm A and a constant $c > 0$ such that for all sufficiently large n and all $s \in [n + O(\log n)]$,

$$\Pr_{x \sim \mathcal{D}_n, A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x, 1^s)] \geq 1 - \frac{1}{q(n)},$$

where $\tau(n) = c \cdot \log n$. We let 2-Sided-Error-Approx-pKt denote the statement that pKt can be approximated on average with two-sided error.

Definition 31 (1-Sided-Error-Approx-pKt)**.** We say that pKt can be approximated on average with *one-sided error* if for every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, and every polynomial q, there exist a PPT algorithm A and a constant $c > 0$ such that for all sufficiently large n and all $s \in [n + O(\log n)]$,

$$\mathop{\mathbf{Pr}}_{x\sim\mathcal{D}_n,A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x,1^s)] \geq 1 - \frac{1}{q(n)},$$

where $\tau(n) = c \cdot \log n$, and for every $x \in \{0,1\}^n$ with $\mathsf{pKt}(x) \leq s$,

$$\mathop{\mathbf{Pr}}_{A}[A(x,1^s) = 1] \geq \frac{2}{3}.$$

We let 1-Sided-Error-Approx-pKt denote the statement that pKt can be approximated on average with one-sided error.

In other words, in addition to computing GapMpKtP on an average $x \sim \mathcal{D}_n$ as in the two-sided error setting, in the one-sided error setting we also require the algorithm to be correct on *all the positive instances*, and hence it can only have one-sided error on the negative instances.

In the following zero error setting, we further require that the algorithm will never output an incorrect answer on both the positive and negative instances (except with small probability over the internal randomness of the algorithm), while it can output $\perp$ to indicate that it does not know the correct answer.

Definition 32 (Zero-Error-Approx-pKt)**.** We say that pKt can be approximated on average with *zero error* if for every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, and every polynomial q, there exist a PPT algorithm A and a constant $c > 0$ such that for all sufficiently large n and all $s \in [n + O(\log n)]$,

$$\mathop{\mathbf{Pr}}_{x\sim\mathcal{D}_n,A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x,1^s)] \geq 1 - \frac{1}{q(n)},$$

where $\tau(n) = c \cdot \log n$, and for every $x \in \{0,1\}^n$ with $\mathsf{pKt}(x) \leq s$,

$$\mathop{\mathbf{Pr}}_{A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x,1^s) \textbf{ OR } A(x,1^s) = \perp] \geq \frac{2}{3}.$$

We let Zero-Error-Approx-pKt denote the statement that pKt can be approximated on average with zero error.

In addition to the above three standard notions of average-case tractability, we also introduce the following notion of average-case tractability for approximating pKt.

Definition 33 (Mild-1-Sided-Error-Approx-pKt)**.** We say that pKt can be approximated on average with *mild-one-sided error* if there exist a constant $0 < \varepsilon < 1$ and a PPT algorithm A such that for all sufficiently large n,

$$\mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n,A}[A(x) = 0] \geq \frac{1}{n},$$

and for every $x \in \{0,1\}^n$ with $\mathsf{pKt}(x) \leq n^\varepsilon$,

$$\mathop{\mathbf{Pr}}_{A}[A(x) = 1] \geq \frac{2}{3}.$$

We let Mild-1-Sided-Error-Approx-pKt denote the statement that pKt can be approximated on average with mild-one-sided error.

As we will show next, the notion of approximating pKt on average with mild-one-sided error is *sandwiched* between that of one-sided error and two-sided error. Also, our result of Theorem 8 states that one-way functions exist if and only if one can close the gap between mild-one-sided error and two-sided error.

Proposition 34. *The following holds.*

$$\begin{array}{ccc} \text{Zero-Error-Approx-pKt} & \Longrightarrow & \text{1-Sided-Error-Approx-pKt} \\ & & \Downarrow \\ & & \text{Mild-1-Sided-Error-Approx-pKt} \Longrightarrow \text{2-Sided-Error-Approx-pKt} \end{array}$$

Proof. (Zero-Error-Approx-pKt $\Longrightarrow$ 1-Sided-Error-Approx-pKt.) If we have an algorithm for computing GapMpKtP on average with zero error, we can replace the output $\perp$ in the algorithm with 1. Then we get an algorithm that satisfies the second condition stated in Definition 31.

(1-Sided-Error-Approx-pKt $\Longrightarrow$ Mild-1-Sided-Error-Approx-pKt.) We will in fact show that to get Mild-1-Sided-Error-Approx-pKt, it suffices to assume that pKt can be approximated within additive error $n - n^{\varepsilon}$ (as opposed to $O(\log n)$ in the definition of 1-Sided-Error-Approx-pKt) on average over the *uniform* distribution.

Suppose 1-Sided-Error-Approx-pKt holds. It follows that there exist a probabilistic polynomial-time algorithm A and a constant $0 < \varepsilon < 1$ such that for all sufficiently large n,

$$\mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n,A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x,1^s)] \geq 1 - \frac{1}{4n^2}, \tag{1}$$

where $\tau(n) := n - 2n^{\varepsilon}$ and $s := n^{\varepsilon}$. Also, for every $x \in \{0,1\}^n$ with $\mathsf{pKt}(x) \leq n^{\varepsilon}$,

$$\mathop{\mathbf{Pr}}_{A}[A(x,1^s) = 1] \geq \frac{2}{3}. \tag{2}$$

Then to get Mild-1-Sided-Error-Approx-pKt, it suffices to show that

$$\mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n}\left[\mathop{\mathbf{Pr}}_{A}[A(x,1^s) = 0] \geq \frac{2}{3}\right] \geq \frac{1}{n}. \tag{3}$$

First of all, by applying an averaging argument to Equation (1), we get that

$$\mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n}\left[\mathop{\mathbf{Pr}}_{A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x,1^s)] \geq 1 - \frac{1}{2n}\right] \geq 1 - \frac{1}{2n}. \tag{4}$$

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

$$\mathsf{pKt}(x) \geq n - O(\log n).$$

For every such x, we get

$$\begin{aligned} \mathsf{pKt}(x) &\geq n - O(\log n) \\ &= n^{\varepsilon} + (n - 2n^{\varepsilon}) + n^{\varepsilon} - O(\log n) \\ &> s + \tau(n). \end{aligned} \tag{5}$$

It follows from Equation (4) and Equation (5) that with probability at least $1 - 1/n$, we get both

$$\mathop{\mathbf{Pr}}_{A}[A \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x,1^s)] \geq 1 - \frac{1}{2n}.$$

and

$$\mathsf{pKt}(x) > s + \tau(n).$$

Note that the above implies that with probability at least $1 - 1/n$,

$$\mathop{\mathbf{Pr}}_{A}[A(x, 1^s) = 0] \geq 1 - \frac{1}{2n},$$

which yields Equation (3), as desired.

(Mild-1-Sided-Error-Approx-pKt $\implies$ 2-Sided-Error-Approx-pKt.) First of all, note that assuming Mild-1-Sided-Error-Approx-pKt one can get an efficient algorithm that breaks any cryptographic pseudorandom generator. This then implies the non-existence of one-way functions [HILL99]. By Theorem 7, this further implies 2-Sided-Error-Approx-pKt. □

The following relates the problem of computing GapMpKtP and that of approximating pKt.

Proposition 35. *The following are equivalent.*

1. 2-Sided-Error-Approx-pKt.
2. *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$ and every polynomial q there exist a probabilistic polynomial-time algorithm A and a polynomial p such that for all $n \in \mathbb{N}$,*

$$\mathop{\mathbf{Pr}}_{x \sim \mathcal{D}_n, A}[\mathsf{pKt}(x) - \log p(n) \leq A(x) \leq \mathsf{pKt}(x)] \geq 1 - \frac{1}{q(n)}.$$

Proof. ($\implies$) Let $\{\mathcal{D}_n\}$ be any distribution family and q be any polynomial. Suppose Item 1 holds. Then we have a probabilistic polynomial-time B and a constant $c > 0$ such that for all sufficiently large n and all $s \in [n + O(\log n)]$,

$$\mathop{\mathbf{Pr}}_{x \sim \mathcal{D}_n}\left[\mathop{\mathbf{Pr}}_{B}[B \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x, 1^s)] \geq 1 - \frac{1}{\sqrt{q(n)}}\right] \geq 1 - \frac{1}{\sqrt{q(n)}}, \tag{6}$$

where $\tau(n) := c \cdot \log n$.

Consider the following algorithm A:

> On input $x \in \{0,1\}^n$, A runs $B(x, 1^s)$ for $s = 1, 2, \ldots, n + O(\log n)$ and outputs the smallest s such that $B(x, 1^s) = 1$.

We say that $x \in \{0,1\}^n$ is *good* if for all $s \in [n + O(\log n)]$,

$$\mathop{\mathbf{Pr}}_{B}[B \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x, 1^s)] \geq 1 - \frac{1}{\sqrt{q(n)}}.$$

Note that by Equation (6) and a union bound over s, a random x picked from $\mathcal{D}_n$ is good with probability at least $1 - 2n/\sqrt{q(n)}$ over $x \sim \mathcal{D}_n$.

Observe that for any good $x \in \{0,1\}^n$, the output of $A(x)$ will lie in the interval $[\mathsf{pKt}(x) - \log \tau(n), \mathsf{pKt}(x)$ if for all $s < \mathsf{pKt}(x) - \log \tau(n)$, $B(x, 1^s) = 0$ and for $s = \mathsf{pKt}(x)$, $B(x, 1^s) = 1$. Since for all s, B decides $\mathsf{GapMpKtP}[\tau]$ on $(x, 1^s)$ with probability at least $1 - 1/\sqrt{q(n)}$, then by a union bound, the probability that the above does not happen is at most $2n/\sqrt{q(n)}$.

It follows that with probability at least $1 - 2n/\sqrt{q(n)}$ over $x \sim \mathcal{D}_n$, $A(x) \in [\mathsf{pKt}(x) - \log \tau(n), \mathsf{pKt}(x)]$ with probability at least $1 - 2n/\sqrt{q(n)}$ (over the internal randomness of A), which implies

$$\Pr_{x\sim\mathcal{D}_n,A}[\mathsf{pKt}(x) - c \cdot \log n \leq A(x) \leq \mathsf{pKt}(x)] \geq 1 - \frac{4n}{\sqrt{q(n)}}.$$

This yields Item 2 by re-scaling the polynomial q.

($\Longleftarrow$) Let $\{\mathcal{D}_n\}$ be any distribution family and q be any polynomial. Suppose Item 2 holds. Then there exist a probabilistic polynomial-time A and a constant $c > 0$ such that for all $n \in \mathbb{N}$,

$$\Pr_{x\sim\mathcal{D}_n,A}[\mathsf{pKt}(x) - c \cdot \log n \leq A(x) \leq \mathsf{pKt}(x)] \geq 1 - \frac{1}{q(n)}.$$

Let B be the algorithm defined as follows. For $x \in \{0,1\}^n$ and $s \in [n + O(\log n)]$,

$$B(x, 1^s) = 1 \iff A(x) \leq s.$$

Fix any $s \in [n + O(\log n)]$, it suffices to show that

$$\Pr_{x\sim\mathcal{D}_n,B}[B \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x, 1^s)] \geq 1 - \frac{1}{q(n)},$$

where $\tau(n) := c \cdot \log n$.

Consider any $x \in \{0,1\}^n$ and any r_A which is the internal randomness used by the algorithm A such that

$$\mathsf{pKt}(x) - c \cdot \log n \leq A(x; r_A) \leq \mathsf{pKt}(x).$$

Note that if $\mathsf{pKt}(x) \leq s$, then we have $A(x; r_A) \leq \mathsf{pKt}(x) \leq s$, which implies $B(x, 1^s) = 1$. Similarly, if $\mathsf{pKt}(x) > s + \tau(n)$, then we have $A(x; r_A) \geq \mathsf{pKt}(x) - c \cdot \log n > s$, which implies $B(x, 1^s) = 0$. Therefore, we get that

$$\begin{aligned}
&\Pr_{x\sim\mathcal{D}_n,B}[B \text{ decides } \mathsf{GapMpKtP}[\tau] \text{ on } (x, 1^s)] \\
&\geq \Pr_{x\sim\mathcal{D}_n,r_A}[\mathsf{pKt}(x) - c \cdot \log n \leq A(x; r_A) \leq \mathsf{pKt}(x)] \\
&\geq 1 - \frac{1}{q(n)},
\end{aligned}$$

as desired. □

4 One-Way Functions and Asymmetry of Information for pKt

4.1 Equivalence of OWF and Average-Case Asymmetry of Information

We show the following which implies Theorem 5.

Theorem 36. *The following are equivalent.*

1. *There exist no (resp. infinitely-often) one-way functions.*
2. **(Infinitely-Often (resp. Almost-Everywhere) Average-Case Symmetry of Information for pKt)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n \times \{0,1\}^n$ and every polynomial q, there exists a constant c such that for infinitely many (resp. all) $n \in \mathbb{N}$,*

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

Lemma 37. (Item 1 ⇒ Item 2 in Theorem 36). *If one-way functions do not exist, then for every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n \times \{0,1\}^n$ and every polynomial q, there exists a constant c such that for infinitely many $n \in \mathbb{N}$,*

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

To show Lemma 37, we need the following technical lemma.

Lemma 38. *If one-way functions do not exist, then for every polynomial-time samplable distribution family $\{\mathcal{E}_n\}_n$ supported over $\{0,1\}^n \times \{0,1\}^n$ and for every polynomial q, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$,*

$$\Pr_{(a,b)\sim\mathcal{E}_n}\left[\mathsf{pKt}(a \mid b) \leq \log \frac{1}{\mathcal{E}_n(a \mid b)} + 2\log p(n)\right] \geq 1 - \frac{1}{q(n)}.$$

Proof. The lemma follows directly from Theorem 17 and Proposition 25. □

We are now ready to show Lemma 37.

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

Let $\{\mathcal{E}_n\}_n$ be the polynomial-time samplable distribution family that is dual to $\{\mathcal{D}_n\}_n$ in the following sense: To sample $\mathcal{E}_n$, we sample (x,y) from $\mathcal{D}_n$ and output (y,x). To show the lemma, it suffices to show that there exists a constant c such that for infinitely many $n \in \mathbb{N}$,

$$\Pr_{(a,b)\sim\mathcal{E}_n}[\mathsf{pKt}(b,a) \geq \mathsf{pKt}(b) + \mathsf{pKt}(a \mid b) - c \cdot \log n] \geq 1 - \frac{1}{q(n)}.$$

Since we assume that one-way function do not exist, then by Lemma 38, there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$,

$$\Pr_{(a,b)\sim\mathcal{E}_n}\left[\mathsf{pKt}(a \mid b) \leq \log \frac{1}{\mathcal{E}_n(a \mid b)} + 2\log p(n)\right] \geq 1 - \frac{1}{2q(n)}.$$

Let $\mathcal{E}'_n$ be the marginal distribution of $\mathcal{E}_n$ on the second half. Note that

$$\begin{aligned}
\mathsf{pKt}(a \mid b) &\leq \log \frac{1}{\mathcal{E}_n(a \mid b)} + 2\log p(n) \\
&= \log \frac{\mathcal{E}_n(a,b)}{\mathcal{E}'_n(b)} + 2\log p(n) \\
&= \log \frac{1}{\mathcal{E}_n(a,b)} - \log \frac{1}{\mathcal{E}'_n(b)} + 2\log p(n). \qquad (7)
\end{aligned}$$

On the one hand, by Lemma 14, we get that for every n, with probability at least $1 - 1/(2q(n))$ over $(a,b) \sim \mathcal{E}_n$,

$$\mathsf{K}(b,a) \geq \log \frac{1}{\mathcal{E}_n(a,b)} - \log 2q(n) - O(\log n).$$

Then by Proposition 25, with the same probability we get

$$\mathsf{pKt}(b,a) \geq \mathsf{K}(b,a) - O(\log n) \geq \log \frac{1}{\mathcal{E}_n(a,b)} - \log 2q(n) - O(\log n). \qquad (8)$$

On the other hand, by Theorem 16, there exists a polynomial p' such that for every n and $b \in \mathsf{Support}(\mathcal{E}'_n)$,

$$\mathsf{pKt}(b) \le \mathsf{pK}^{p'(n)}(b) + \log p'(n) \le \log \frac{1}{\mathcal{E}'_n(y)} + 2\log p'(n). \tag{9}$$

By plugging Equations (8) and (9) into Equation (7), and by a union bound, we get that for infinitely many $n \in \mathbb{N}$, with probability at least $1 - 1/q(n)$ over $(a, b) \sim \mathcal{E}_n$,

$$\mathsf{pKt}(a \mid b) \le \mathsf{pKt}(b, a) - \mathsf{pKt}(b) + 2\log p(n) + \log 2q(n) + 2\log p'(n) + O(\log n),$$

as desired. □

Lemma 39. (Item 2 ⇒ Item 1 in Theorem 36). *If infinitely-often average-case symmetry of information for* pKt *holds, then one-way functions do not exist.*

Proof. Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be any candidate one-way function that is supposed to be infinitely-often secure. Let q be any polynomial. We will construct a polynomial-time algorithm that inverts f with probability at least $1 - 1/q(n)$. We first show a few useful claims.

Claim 40 ([LW95, Lemma 3.5])**.** *For every n and every $x \in \{0,1\}^n$, we have*

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

Proof of Claim 40. Note that for every $x \in \{0,1\}^n$, we have

$$\mathsf{K}(x \mid f(x)) \le \log |f^{-1}(f(x))| + O(\log n). \tag{10}$$

This is because given $f(x)$, we can recover x knowing the index of x in the set $f^{-1}(f(x))$. Also, we have

$$\mathsf{K}(x) \le \mathsf{K}(x \mid f(x)) + \mathsf{K}(f(x)),$$

which combined with Equation (10) yields

$$\begin{aligned}\mathsf{K}(f(x)) &\ge \mathsf{K}(x) - \mathsf{K}(x \mid f(x)) \\ &\ge \mathsf{K}(x) - \log |f^{-1}(f(x))| - O(\log n).\end{aligned}$$

This completes the proof of Claim 40. ◇

Claim 41. *For infinitely many $n \in \mathbb{N}$, with probability at least $1 - 1/q(n)^2$ over $x \sim \{0,1\}^n$, we have*

$$\mathsf{pKt}(x \mid f(x)) \le \log |f^{-1}(f(x))| + O(\log q(n)).$$

Proof of Claim 41. Consider the polynomial-time samplable distribution family $\{\mathcal{D}_n\}$ where each $\mathcal{D}_n$ samples $x \sim \{0,1\}^n$ and outputs $(f(x), x)$.

By the assumption that infinitely-often average-case symmetry of information for pKt holds, there is a constant $c > 0$ such that for infinitely many $n \in \mathbb{N}$, with probability at least $1 - 1/(2q(n)^2)$ over $x \sim \{0,1\}^n$,

$$\begin{aligned}\mathsf{pKt}(x \mid f(x)) &\le \mathsf{pKt}(f(x), x) - \mathsf{pKt}(f(x)) + c\log n \\ &\le \mathsf{pKt}(x) - \mathsf{pKt}(f(x)) + c\log n + O(\log n) \\ &\le \mathsf{pKt}(x) - \mathsf{K}(f(x)) + c\log n + O(\log n) \\ &\le \mathsf{pKt}(x) - \big(\mathsf{K}(x) - \log |f^{-1}(f(x))| - O(\log n)\big) + c\log n + O(\log n) \\ &\le \mathsf{pKt}(x) - \mathsf{K}(x) + \log |f^{-1}(f(x))| + c\log n + O(\log n),\end{aligned} \tag{11}$$

where the second inequality uses the fact that given x we can compute $f(x)$ efficiently and the second last inequality is by Claim 40.

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

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

which implies

$$\mathsf{pKt}(x) - \mathsf{K}(x) \leq O(\log q(n)).$$

Plugging this into Equation (11), we get, by a union bound, that with probability at least $1-1/q(n)^2$ over $x \sim \{0,1\}^n$

$$\mathsf{pKt}(x \mid f(x)) \leq \log |f^{-1}(f(x))| + O(\log q(n)),$$

as desired. ◇

Claim 42. *For every $n \in \mathbb{N}$, every image y of f and $k \leq 2n$, with probability at least $1 - 1/q(n)$ over $x \sim f^{-1}(y)$, we have*

$$\mathop{\mathbf{Pr}}_{r \sim \{0,1\}^{2^k}} \left[\mathsf{K}(x \mid y, r) \geq \log |f^{-1}(y)| - O(\log q(n))\right] \geq 1 - \frac{1}{n}.$$

Proof of Claim 42. By a simple counting argument, for every fixed image y of f and every fixed $r \in \{0,1\}^{2^k}$, we have

$$\mathop{\mathbf{Pr}}_{x \sim f^{-1}(y)} \left[\mathsf{K}(x \mid y, r) \geq \log |f^{-1}(y)| - O(\log q(n))\right] \geq 1 - \frac{1}{n \cdot q(n)}.$$

This implies that every image y, we have

$$\mathop{\mathbf{Pr}}_{\substack{x \sim f^{-1}(y) \\ r \sim \{0,1\}^{2^k}}} \left[\mathsf{K}(x \mid y, r) \geq \log |f^{-1}(y)| - O(\log q(n)\right] \geq 1 - \frac{1}{n \cdot q(n)}.$$

Finally, by an averaging argument, we have that with probability at least $1-1/q(n)$ over $x \sim f^{-1}(y)$, it holds that

$$\mathop{\mathbf{Pr}}_{r \sim \{0,1\}^{2^k}} \left[\mathsf{K}(x \mid y, r) \geq \log |f^{-1}(y)| - O(\log q(n)\right] \geq \frac{1}{n},$$

as desired. ◇

By Claim 41, we get that for infinitely many $n \in \mathbb{N}$, with probability at least $1 - 1/q(n)^2$ over $x \sim \{0,1\}^n$, we have

$$\mathsf{pKt}(x \mid f(x)) \leq \log |f^{-1}(f(x))| + O(\log q(n)). \tag{12}$$

In what follows, we fix n so that Equation (12)) holds.

Now observe the following equivalent way of sampling $(x, f(x))$ while x is uniformly at random: We first sample $y := f(z)$ for a uniformly random z and then sample $x \sim f^{-1}(y)$. By an averaging argument, Equation (12) yields that with probability at least $1 - 1/q(n)$ over y sample this way, for at least $1 - 1/q(n)$ fraction of the $x \in f^{-1}(y)$, we have

$$\mathsf{pKt}(x \mid y) \leq \log |f^{-1}(y)| + O(\log q(n)). \tag{13}$$

Consider any *good* y such that Equation (13) holds. By Claim 42, we get that for at least $1 - 1/q(n)$ fraction of the $x \in f^{-1}(y)$, it holds that

$$\Pr_{r \sim \{0,1\}^{2^k}} \left[\mathsf{K}(x \mid y, r) \geq \log |f^{-1}(y)| - O(\log q(n)) \right] \geq 1 - \frac{1}{n}, \tag{14}$$

where $k := \mathsf{pKt}(x \mid y)$.

let S_y be the set of $x \in f^{-1}(y)$ such that both Equations (13) and (14) hold. Note that by a union bound,

$$|S_y| \geq (1 - 2/q(n)) \cdot |f^{-1}(y)|.$$

Let $d > 0$ be a sufficiently large constant. Consider the following procedure A that takes n and y as input and does the following.

1. Pick a uniformly random $k \sim [O(n)]$,
2. Pick a uniformly random $r \sim \{0,1\}^{2^k}$,
3. Pick uniformly at random $\ell \sim [O(n)]$ and $p \sim \{0,1\}^{\ell}$,
4. Run $U^{y,r}(p)$ for n^d steps and return its output.

Claim 43. *For every $x \in S_y$, $\mathsf{A}(1^n, y)$ outputs x with probability at least*

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

Proof of Claim 43. Fix $x \in S_y$. Note that we have

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

In other words, for $k := \mathsf{pKt}(x \mid y) \leq \log |f^{-1}(y)| + O(\log n)$, with probability at least $2/3$ over $r \sim \{0,1\}^{2^k}$, there exist a program of p and a running time $t \in \mathbb{N}$ such that $|p| + \log t \leq k$ and $U^{y,r}(p)$ outputs x within t steps. Note that t may not be upper bounded by $\mathsf{poly}(n)$. However, if for that r we also have that

$$\mathsf{K}(x \mid y, r) \geq \log |f^{-1}(y)| - O(\log q(n)), \tag{15}$$

then it must be the case that $|p| \geq \log |f^{-1}(y)| - O(\log q(n))$. The condition $|p| + \log t \leq \log |f^{-1}(y)| + O(\log q(n))$ then implies that $t \leq n^d$ for some sufficiently large constant d.

Note that Equation (15) also holds with probability at least $1 - 1/n$ over $r \sim \{0,1\}^{2^k}$. It follows that with probability at least $2/3 - 1/n$ over $r \sim \{0,1\}^{2^k}$, there is a program p of size at most $\log |f^{-1}(y)| + O(\log q(n))$ such that $U^{y,r}(p)$ outputs x within n^d steps.

Therefore, after performing the first 3 steps in the procedure, we get such a program p with probability at least

$$\frac{1}{O(n)} \cdot \left(\frac{2}{3} - \frac{1}{n} \right) \cdot \frac{1}{O(n)} \cdot \frac{1}{2^{\log |f^{-1}(y)| + O(\log q(n))}} \geq \frac{1}{\mathsf{poly}(n) \cdot |f^{-1}(y)|},$$

as desired. ◇

Now consider the following procedure A' that can *simulate* A.

1. Pick a uniformly random $k \sim [O(n)]$,
2. Pick uniformly at random $\ell \sim [O(n)]$ and $p \sim \{0,1\}^{\ell}$,
3. Run $U^{y,(-)}(p)$ for n^d steps while answering its queries to the second oracle string (which is of length 2^k) as follows. For any valid query, if it did not appear before, pick a random bit b, record the query as well as the bit b, and return b; otherwise, return the corresponding bit recorded.

Denote the above procedure by A'. Note that $\mathsf{A}(1^n, y)$ has running time $\mathsf{poly}(n)$. Also, by Claim 43 the probability that $\mathsf{A}'(1^n, y)$ outputs *some* $x \in S_y$ is at least

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

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

Finally, we complete the proof of Theorem 36.

Proof of Theorem 36. Each direction of the theorem follows directly from Lemmas 37 and 39, respectively.

Also, we note that while those lemmas only show the equivalence between the non-existence of one-way functions and infinitely-often average-case symmetry of information for pKt, it is straightforward to adapt the proofs to show the equivalence between the non-existence of infinitely-often one-way functions and almost-everywhere average-case symmetry of information for pKt. □

4.2 Asymmetry of Information from Circuit Lower Bounds

In this subsection, we show the following which implies Theorem 6.

Theorem 44. *Suppose* $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$ *(resp.* $\mathsf{BPEXP} \not\subseteq \mathsf{SIZE[poly]}$*). Then for every constant* $c > 0$*, there exist* $x, y \in \{0,1\}^n$ *such that*

$$\mathsf{pKt}(x, y) < \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \log n,$$

for all but finitely many (resp. infinitely many) n*.*

We first show the following technical lemma.

Lemma 45. *Suppose* $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$ *(resp.* $\mathsf{BPEXP} \not\subseteq \mathsf{SIZE[poly]}$*). Then for every constant* $c > 0$*, the following holds for all but finitely many (resp. infinitely many)* m*. There exist* $v \in \{0,1\}^m$ *and* $u \in \{0,1\}^{m'}$*, where* $m \leq m' \leq 2^{m/(8c)}$*, such that*

$$\mathsf{pKt}(u, v) < \mathsf{pKt}(u) + \mathsf{pKt}(v \mid u) - m/4,$$

Proof. We first show the case for $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$.

For the sake of contradiction, suppose there is a constant $c > 0$ such that the following holds for infinitely many m. For all $v \in \{0,1\}^m$ and all $u \in \{0,1\}^{m'}$, where $m' \leq 2^{m/(8c)}$,

$$\mathsf{pKt}(u, v) \geq \mathsf{pKt}(u) + \mathsf{pKt}(v \mid u) - m/4.$$

Fix any (sufficiently large) m such that the above holds. For $x \in \{0,1\}^m$ and $y \in \{0,1\}^{\leq 2^m}$, let $C_{(x,y)} \colon \{0,1\}^{2^{bm}} \to \{0,1\}$ be a circuit of size 2^{bm} that views its input as internal randomness and computes $B(x,y)$, where B is the algorithm in Lemma 29 and $b \geq 0$ is a constant.

Let $\varepsilon := 1/(32bc)$ and let $G_s : \{0,1\}^r \to \{0,1\}^s$ be the PRG in Lemma 18, where $s := 2^{bm}$ and $r := s^{\varepsilon}$. We assume without loss of generality, using amplification if necessary, that the pseudodeterministic algorithm that computes G_s outputs the correct answer except with exponentially small probability. We abuse notation and use G_s to denote the algorithm that computes the PRG G_s.

Consider the following algorithm.

Algorithm 1 Pseudodeterministic Constructions of Large pKt-Complexity Strings

```
procedure A(1^m, y)
    for x ∈ {0,1}^m do
        μ_x := Pr_{z∼{0,1}^r}[C_(x,y)(G_s(z)) = 1]
        if μ_x > 1/3 + 1/10 then
            output x
    Output ⊥
```

Claim 46. *The above algorithm A, on input 1^m and $y \in \{0,1\}^{\leq 2^m}$, runs in time $2^{2^{m/(16c)}}$ and outputs, with high probability, a fixed m-bit string x such that $\mathsf{pKt}(x \mid y) \geq m/2$.*

Proof of Claim 46. We first argue the running time. It is easy to that the algorithm runs in time

$$2^m \cdot 2^{O(s^{\varepsilon})} \leq 2^{2^{m/(16c)}}.$$

Also, since G_s can be computed pseudodeterministically with error $1/\exp(s)$, by a union bound over $x \in \{0,1\}^m$ and $z \in \{0,1\}^{s^{\varepsilon}}$, the algorithm will output a fixed answer with high probability.

We now argue the correctness. Note that since G_s (1/10)-fools $C_{(x,y)}$ on *every* $x \in \{0,1\}^m$ and $y \in \{0,1\}^{\leq 2^m}$, for every x, (the canonical) μ_x is a good estimate of $\mathbf{Pr}_B[B(x,y)=1]$. Then an output x of the algorithm cannot be that $\mathsf{pKt}(x \mid y) < m/2$. This is because in that case the algorithm $\mathbf{Pr}_B[B(x,y)=1] < 1/3$ and μ_x should be less than $1/3+1/10$. Also, since we enumerate every x in $\{0,1\}^m$, $B(-,y)$ must accept at least one x, and in this case we have $\mu_x \geq 2/3 - 1/10 \geq 1/3 + 1/10$. This completes the proof of Claim 46. ◇

Let $t := 2^{m/(10c)}$. We define $z_1, z_2, \ldots, z_t \in \{0,1\}^m$ as follows.

- z_1 is the canonical output of $A(1^m, \emptyset)$.
- z_i is the canonical output of $A(1^m, z_1, \ldots, z_{i-1})$ for $i = 2, 3. \ldots, t$.

On the one hand, by our assumption and by Claim 46, we have

$$\begin{aligned}
\mathsf{pKt}(z_1, z_2, \ldots, z_t) &\geq \mathsf{pKt}(z_1, z_2, \ldots, z_{t-1}) + \mathsf{pKt}(z_t \mid z_1, z_2, \ldots, z_{t-1}) - m/4 \\
&\geq \mathsf{pKt}(z_1, z_2, \ldots, z_{t-1}) + m/2 - m/4 \\
&\geq \mathsf{pKt}(z_1, z_2, \ldots, z_{t-1}) + m/4.
\end{aligned}$$

We can repeat the above for $\mathsf{pKt}(z_1, z_2, \ldots, z_{t-1})$ and so on. As a result we get

$$\mathsf{pKt}(z_1, z_2, \ldots, z_t) \geq tm/4 \geq 2^{m/(10c)}. \tag{16}$$

On the other hand, since A is pseudodeterministic (and hence outputs the same value with very high probability) and runs in time $2^{2^{m/(16c)}}$, given the numbers m and t, we can obtain $z_1, z_2, \ldots, z_t$ (with high probability) in time

$$O\Big(t \cdot 2^{2^{m/(16c)}}\Big).$$

Therefore, we have

$$\mathsf{pKt}(z_1, z_2, \ldots, z_t) \leq O(\log m) + O(\log t) + 2^{m/(16c)} < 2^{m/(8c)},$$

which contradicts Equation (16).

The case for $\mathsf{BPEXP} \not\subseteq \mathsf{SIZE}[\mathsf{poly}]$ can be shown similarly. We assume, for contradiction, that there is a constant $c > 0$ such that the following holds for all but finitely many m. For all $v \in \{0,1\}^m$ and all $u \in \{0,1\}^{m'}$, where $m \leq m' \leq 2^{m/(8c)}$,

$$\mathsf{pKt}(u, v) \geq \mathsf{pKt}(u) + \mathsf{pKt}(v \mid u) - m/4.$$

We then consider any (sufficiently large) s such that G_s is a good PRG, and let m be the largest integer such that $2^{bm} \leq s$. The remainder of the argument is essentially the same. □

We are now ready to show Theorem 44.

Proof of Theorem 44. We first show the case for $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$.

Let $c > 0$ be a sufficiently large constant, and let n be any large enough integer. We let m be such that

$$2^{m/(8c)} \leq n < 2^{(m+1)/(8c)}. \tag{17}$$

By Lemma 45, there exist $v \in \{0,1\}^m$ and $u \in \{0,1\}^{m'}$, where $m \leq m' \leq 2^{m/(8c)}$, such that

$$\mathsf{pKt}(u, v) < \mathsf{pKt}(u) + \mathsf{pKt}(v \mid u) - m/4. \tag{18}$$

We let

$$x := u0^{n-|u|} \quad \text{and} \quad y := v0^{n-|v|}. \tag{19}$$

Then we have

$$\begin{aligned}
\mathsf{pKt}(x, y) &\leq \mathsf{pKt}(u, v) + O(\log n) && \text{(by Equation (19))} \\
&\leq \mathsf{pKt}(u) + \mathsf{pKt}(v \mid u) - m/4 + O(\log n) && \text{(by Equation (18))} \\
&< \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) + O(\log n) - m/4 + O(\log n) && \text{(by Equation (19))} \\
&\leq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) + O(m/c) - m/4 && \text{(by Equation (17))} \\
&\leq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - m/5 \\
&\leq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \log n, && \text{(by Equation (17))}
\end{aligned}$$

where for the second last inequality we use that c is a sufficiently large constant.

The case for $\mathsf{BPEXP} \not\subseteq \mathsf{SIZE}[\mathsf{poly}]$ can be shown similarly. By Lemma 45, for infinitely many m, there exist $v \in \{0,1\}^m$ and $u \in \{0,1\}^{m'}$, where $m' \leq 2^{m/(8c)}$, such that

$$\mathsf{pKt}(u, v) < \mathsf{pKt}(u) + \mathsf{pKt}(v \mid u) - m/4.$$

We can then let $x := v0^{m'-|v|}$ and $y := u$, and the remainder of the argument is essentially the same as in the case for $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE}[\mathsf{poly}]$ described above. □

4.3 Proof of Theorem 1

In this subsection, we prove Theorem 1, which is restated below.

Theorem 1 (Conditional Equivalence Between OWF and Worst-to-Average-Case Failure of SoI)**.** *Under the assumption that* $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$*, the following equivalence holds:*

$$(\mathsf{Worst\text{-}Case\text{-}Asymmetry\text{-}pKt} \Rightarrow \mathsf{Average\text{-}Case\text{-}Asymmetry\text{-}pKt}) \iff \exists\mathsf{OWF}$$

Proof. Suppose it holds that (Worst-Case-Asymmetry-pKt ⇒ Average-Case-Asymmetry-pKt).

Assuming $\mathsf{BPEXP} \not\subseteq \mathsf{i.o.SIZE[poly]}$, by Theorem 44, Worst-Case-Asymmetry-pKt holds, which then implies that Average-Case-Asymmetry-pKt also holds. By Theorem 5, this implies that one-way functions exist.

Suppose one-way functions exist. Then by Theorem 5, Average-Case-Asymmetry-pKt holds. This implies that (Worst-Case-Asymmetry-pKt ⇒ Average-Case-Asymmetry-pKt). □

4.4 Non-Existence of Natural Properties via Reductions from Worst-Case to Average-Case Asymmetry of Information

Here, we show that if worst-case asymmetry of information pKt implies the average-case, then natural properties do no exist.

Theorem 47. *If infinitely-often worst-case asymmetry of information for* pKt *implies the average-case, then for every* $\varepsilon > 0$*, dense* prBPP*-properties useful against* $\mathsf{SIZE}[2^{n^\varepsilon}]$ *do not exist.*

We need the following result.

Lemma 48. *If dense* prBPP*-properties useful against* P/poly *exist, then* $\mathsf{BPEXP} \not\subseteq \mathsf{P/poly}$.

Proof Sketch. It follows from [CIKK16, OS17a] that the existence of dense prBPP-properties useful against P/poly yields a non-trivial learning algorithm for polynomial-size circuits. It was shown in [OS17a] that such an algorithm gives a language in BPEXP that is not in P/poly. □

We are now ready to show Theorem 47.

Proof of Theorem 47. Suppose, for the sake of contradiction, dense prBPP-properties useful against $\mathsf{SIZE}[2^\varepsilon]$ exist, for some $\varepsilon > 0$.

On the one hand, by Theorem 23, the existence of such properties implies that infinitely-often one-way functions do not exist.

On the other hand, by Lemma 48, such a property implies $\mathsf{BPEXP} \not\subseteq \mathsf{P/poly}$. Then by Theorem 44, infinitely-often worst-case asymmetry of information for pKt holds. By our assumption, this implies infinitely-often average-case asymmetry of information for pKt, which, by Theorem 36, implies the existence of infinitely-often one-way functions. A contradiction. □

5 One-Way Functions and Hardness of Approximating pKt

5.1 Equivalence of OWF and Average-Case Hardness of Approximating pKt

We show the following which, by Proposition 35, implies Theorem 7.

Theorem 49. *The following are equivalent.*

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

2. **(Average-Case Easiness of Approximating pKt)** *For every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, there exist a probabilistic polynomial-time algorithm A and a polynomial τ such that for all $n \in \mathbb{N}$,*

$$\Pr_{x\sim\mathcal{D}_n,A}[\mathsf{pKt}(x) - \log\tau(n) \leq A(x) \leq \mathsf{pKt}(x)] \geq 1 - \frac{1}{q(n)}.$$

Lemma 50. (Item 1 ⇒ Item 2 in Theorem 49). *If infinitely-often one-way functions do not exist, then for every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, there exist a probabilistic polynomial-time algorithm A and a polynomial τ such that for all $n \in \mathbb{N}$,*

$$\Pr_{x\sim\mathcal{D}_n,A}[\mathsf{pKt}(x) - \log\tau(n) \leq A(x) \leq \mathsf{pKt}(x)] \geq 1 - \frac{1}{q(n)}.$$

Proof. Let $\{\mathcal{D}_n\}_n$ be a polynomial-time samplable distribution family, where each $\mathcal{D}_n$ is over $\{0,1\}^n$. Let q be any polynomial. Let τ be a polynomial specified later.

First of all, by Theorem 16 and Proposition 25, there is a polynomial p such that for all $x \in \mathsf{Support}(\mathcal{D}_n)$,

$$\mathsf{pKt}(x) \leq \mathsf{pK}^{p(n)}(x) + \log p(n) \leq \log\frac{1}{\mathcal{D}_n(x)} + 2\log p(n). \tag{20}$$

On the other hand, by Lemma 14, we get

$$\Pr_{x\sim\mathcal{D}_x}\left[\mathsf{K}(x) > \log\frac{1}{\mathcal{D}_n(x)} - b\cdot\log n - 2\log q(n)\right] \geq 1 - \frac{1}{q(n)^2},$$

for some large constant b. Note that by Proposition 25,

$$\mathsf{K}(x) \leq \mathsf{pKt}(x) + b\cdot\log n.$$

Then the above implies that with probability at least $1 - 1/q(n)^2$ over $x \sim \mathcal{D}_n$, it holds that

$$\mathsf{pKt}(x) > \log\frac{1}{\mathcal{D}_n(x)} - 2b\cdot\log n - 2\log q(n). \tag{21}$$

Let B be the algorithm in Theorem 11, instantiated with the polynomial $q'(n) := q(n)^2$. Then by Theorem 11, we get that at least $1 - 1/q(n)^2$ over $x \sim \mathcal{D}_n$ and the internal randomness of B,

$$\frac{\mathcal{D}_n(x)}{2} \leq B(1^n, x) \leq \mathcal{D}_n(x). \tag{22}$$

Our algorithm A works as follows: On $(x, 1^s)$, output

$$\beta := \log\frac{1}{B(1^n,x)} - 2b\cdot\log n - 2\log q(n) - 1.$$

It is easy to see that A runs in polynomial time. Next, we show its correctness.

Note that if both Equation (21) and Equation (22) hold, which happens with probability at least $1 - 1/q(n)$ over $x \sim \mathcal{D}_n$ and the internal randomness of A, we have both

$$\begin{aligned}
\beta &:= \log\frac{1}{B(1^n,x)} - 2b\cdot\log n - 2\log q(n) - 1 \\
&\leq \log\frac{1}{\mathcal{D}_n(x)} + 1 - 2b\cdot\log n - 2\log q(n) - 1 && \text{(by Equation (22))} \\
&\leq \mathsf{pKt}(x). && \text{(by Equation (21))}
\end{aligned}$$

and

$$\begin{aligned}
\beta :&= \log \frac{1}{B(1^n, x)} - 2b \cdot \log n - 2 \log q(n) - 1 \\
&\geq \log \frac{1}{\mathcal{D}_n(x)} - 1 - 2b \cdot \log n - 2 \log q(n) - 1 && \text{(by Equation (22))} \\
&\geq \mathsf{pKt}(x) - 2 - 2b \cdot \log n - 2 \log q(n) - 2 \log p(n) && \text{(by Equation (20))} \\
&\geq \mathsf{pKt}(x) - \log \tau(n),
\end{aligned}$$

where the last inequality holds if we let τ be a sufficiently large polynomial. This completes the proof of the lemma. □

Lemma 51. (Item 2 ⇒ Item 1 in Theorem 49). *Suppose for every polynomial-time samplable distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$, there exist a probabilistic polynomial-time algorithm A and a polynomial τ such that for all $n \in \mathbb{N}$,*

$$\Pr_{x \sim \mathcal{D}_n, A}[\mathsf{pKt}(x) - \log \tau(n) \leq A(x) \leq \mathsf{pKt}(x)] \geq 1 - \frac{1}{q(n)}.$$

Then infinitely-often one-way functions do not exist.

Proof Sketch. The proof follows that of [IRS21, Theorem 36]. At a higher level, the idea is that if we have an efficient algorithm for approximating pKt on average over polynomial-time samplable distributions, then we can construct a function that distinguishes the output distribution of a cryptographic pseudorandom generator from the uniform distribution. This is because the outputs of such a generator have low pKt complexity while a uniformly random string has high pKt complexity. By [HILL99], this implies that infinitely-often one-way functions do not exist. □

We now complete the proof of Theorem 49.

Proof of Theorem 49. The theorem follows directly from Lemmas 50 and 51. □

5.2 Hardness of Approximating pKt with Mild-One-Sided Error

In this subsection, we prove Theorem 8, which is restated below.

Theorem 8. Mild-1-Sided-Error-Approx-pKt *is false. Moreover, the corresponding lower bound also holds against randomized algorithms running in time $n^{\mathsf{poly}(\log n)}$.*

To show Theorem 8, we will need the following lemmas.

Lemma 52. *If pKt can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$, then $\mathsf{BPE} \subseteq \mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$.*

Proof. Let $L \in \mathsf{BPE}$. we identify L with a function from $\{0,1\}^*$ to $\{0,1\}$.

Since pKt can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$, by standard amplification techniques, there exist a constant $0 < \varepsilon < 1$ and a probabilistic polynomial-time A such that for all sufficiently large n,

$$\Pr_{x \sim \{0,1\}^n}\left[\Pr_A[A(x) = 0] \geq 1 - \frac{1}{n^2}\right] \geq \frac{1}{n},$$

Note that this implies

$$\mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n,A}[A(x)=0]\geq\frac{1}{2n}.$$

Also, for every $x\in\{0,1\}^n$ with $\mathsf{pKt}(x)\leq n^{\varepsilon}$,

$$\mathop{\mathbf{Pr}}_{A}[A(x)=1]\geq 1-\frac{1}{n^2}.$$

Now let $\left\{G_n^{(-)}\colon\{0,1\}^{n^\lambda}\to\{0,1\}\right\}_n$ be the pseudorandom generator from Theorem 19 instantiated with parameter $\lambda:=\varepsilon/2$.

Firstly, note that for every $r\in\{0,1\}^{n^\lambda}$, $G_n^L(r)$ can be computed in deterministic time $\exp\big(O(n^\lambda)\big)$ given oracle access to L on inputs of length at most n^λ. Since L on inputs of length at most n^λ can be computed in *randomized* time $\exp(O(n^\lambda))$. It follows that $G_n^L(r)$ can be obtained (with high probability) in randomized time $\exp(O(n^\lambda))$, which implies

$$\mathsf{pKt}(G_n^L(r))\leq O(n^\lambda)\leq n^{\varepsilon}. \tag{23}$$

We remark that since G_n^L can only be computed in a randomized manner, in the above we can only upper-bound the pKt complexity of the output strings of G_n^L, rather than the Kt complexity.[5]

It follows by the properties of the algorithm A that

$$\mathop{\mathbf{Pr}}_{r\sim\{0,1\}^{n^\lambda},A}\big[A(G_n^L(r))=1\big]\geq 1-\frac{1}{n^2}\quad\text{and}\quad\mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n,A}[A(x)=1]\leq 1-\frac{1}{2n}.$$

Now by averaging, there exists some fixing of the internal randomness of A, which gives a quasi-polynomial-size $\mathcal{O}$-oracle circuit A', so that A' can distinguish the output of G_n^L from the uniform distribution with advantage at least $1/(3n)$. By Theorem 19, we get that L can be computed by polynomial-size A'-oracle circuits, and hence by a quasi-polynomial-size $\mathcal{O}$-oracle circuit. □

Lemma 53. *If* pKt *can be approximated on average with mild-one-sided error in time* $n^{\mathsf{poly}(\log n)}$*, then* $\mathsf{PSPACE}\subseteq\mathsf{BPTIME}[n^{\mathsf{poly}(\log n)}]$.

Proof. The proof is similar to that of Lemma 52. We present the full proof for completeness.

Let L_{TV} be the PSPACE-hard language in Theorem 21. It suffices to show that $L_{\mathsf{TV}}\in\mathsf{BPTIME}[n^{\mathsf{poly}(\log n)}]$.

Since Since pKt can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$, by standard amplification techniques, there exist a constant $0<\varepsilon<1$ and a probabilistic polynomial-time A such that for all sufficiently large n,

$$\mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n}\left[\mathop{\mathbf{Pr}}_{A}[A(x)=0]\geq 1-\frac{1}{n^2}\right]\geq\frac{1}{n},$$

Note that this implies $\mathbf{Pr}_{x\sim\{0,1\}^n,A}[A(x)=0]\geq 1/(2n)$. Also, for every $x\in\{0,1\}^n$ with $\mathsf{pKt}(x)\leq n^{\varepsilon}$, $\mathbf{Pr}_A[A(x)=1]\geq 1-1/n^2$.

Let $\left\{\mathsf{IW}_n^{(-)}\colon\{0,1\}^{n^\lambda}\to\{0,1\}\right\}_n$ be the pseudorandom generator from Theorem 20 instantiated with parameter $\lambda:=\varepsilon/2$.

[5] We would be able to upper bound Kt complexity if we started with a language in E instead of BPE, but the resulting inclusion $\mathsf{E}\subseteq\mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$ would be insufficient for the proof of a deterministic variant of Theorem 8, since a deterministic analogue of Lemma 53 is unknown.

For every $r \in \{0,1\}^{n^\lambda}$, $\mathsf{IW}_n^{L_{\mathsf{TV}}}(r)$ can be computed in deterministic time $\exp\big(O(n^\lambda)\big)$ given oracle access to L_{TV} on inputs of length at most n^λ. Since $L_{\mathsf{TV}} \in \mathsf{DPACE}[O(n)]$, L_{TV} on inputs of length at most n^λ can be computed in deterministic time $\exp(O(n^\lambda))$. It follows that $\mathsf{IW}_n^{L_{\mathsf{TV}}}(r)$ can be computed in time $\exp(O(n^\lambda))$, which implies

$$\mathsf{pKt}\big(\mathsf{IW}_n^{L_{\mathsf{TV}}}(r)\big) \leq O(n^\lambda) \leq n^\varepsilon. \tag{24}$$

It follows by the properties of the algorithm A that

$$\mathop{\mathbf{Pr}}_{r\sim\{0,1\}^{n^\lambda},A}\big[A(\mathsf{IW}_n^{L_{\mathsf{TV}}}(r)) = 1\big] \geq 1 - \frac{1}{n^2} \quad \text{and} \quad \mathop{\mathbf{Pr}}_{x\sim\{0,1\}^n,A}[A(x) = 1] \leq 1 - \frac{1}{2n}.$$

By Theorem 20, this yields a quasi-polynomial-time randomized algorithm for computing L_{TV}. □

Finally, we need the following lemma.

Lemma 54. *There is a language in* $\mathsf{DSPACE}[2^{n^{o(1)}}]\backslash\mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$.

Proof. This can be shown using a diagonalization argument, which is similar to the proof of the folklore result that PSPACE does not have fixed-polynomial-size circuits. Here, for $n \in \mathbb{N}$, we use one input of length n to diagonalize against at least half of the circuits in $\mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$. Details follow.

Let s be any time-constructable function such that $s(n) \in 2^{n^{o(1)}}$ and $s(n) \in 2^{\log^{\omega(1)}(n)}$. For $n \in \mathbb{N}$, we let S_n denote the set of circuits with size at most $s(n)$. Note that $|S_n| \leq \exp(s(n) \cdot \log s(n))$.

We describe a language L that is computable in space $\mathsf{poly}(s(n))$ but not in $\mathsf{SIZE}[s(n)]$.

Fix $n \in \mathbb{N}$. We let $t := t(n)$ be the smallest integer such that $|S_n| \cdot 2^{-t} < 1$. Note that $t \leq O(s(n) \cdot \log s(n))$. Also, we identify t with the n-bit string which is the binary representation of t.

For $x \in \{0,1\}^n$ such that $x \preceq t$, where "$\preceq$" is defined with respect to lexicographic order, we define $b_x \in \{0,1\}$ recursively as follows.

$$b_x := \mathsf{Majority}\{C(x) \mid C \in S_n \text{ and } C(y) = b_y \text{ for all } y \prec x\}.$$

We then let $L(x) := 1 - b_x$. Also, for $x \succ t$, we let $L(x) := 0$.

First of all, note that given $x \preceq t$ and $\{b_y\}_{y\prec x}$, we can compute b_x in space $\mathsf{poly}(s(n))$, by enumerating each $C \in S_n$ and simulating C on inputs $\preceq x$. Since we can also recursively compute (and store) b_y for all $y \prec x$, we conclude that $L(x)$ can be computed in space $\mathsf{poly}(s(n))$.

To see that L is not in $\mathsf{SIZE}[s(n)]$, note that using 0^n, we can diagonalize against at least half of the circuits in S_n. Then using $0^{n-1}1$, we can further diagonalize against at least half of the remaining circuits. After t steps, we will be able to diagonalize against all the circuits in S_n. □

We are now ready to show Theorem 8.

Proof of Theorem 8. Suppose, for the sake of contradiction, pKt can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$. First of all, by Lemma 52, we get that $\mathsf{BPE} \subseteq \mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$.

By Lemma 54, there is a language $L \in \mathsf{DSPACE}[2^{n^{o(1)}}]\backslash\mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$. Now by Lemma 53, we get that $\mathsf{PSPACE} \subseteq \mathsf{BPTIME}[n^{\mathsf{poly}(\log n)}]$. Then by a padding argument, we get that $L \in \mathsf{BPE}$. However, this means that $\mathsf{BPE} \notin \mathsf{SIZE}[n^{\mathsf{poly}(\log n)}]$. A contradiction. □

5.3 Proof of Theorem 3

In this subsection, we show Theorem 3, which is restated below.

Theorem 3 (Equivalence Between OWF and One-Sided to Two-Sided Error Reductions for pKt)**.** *The following equivalence holds:*

$$(\text{2-Sided-Error-Approx-pKt} \Rightarrow \text{Mild-1-Sided-Error-Approx-pKt}) \iff \exists\text{i.o.OWF}$$

Proof. Suppose it holds that (2-Sided-Error-Approx-pKt $\Rightarrow$ Mild-1-Sided-Error-Approx-pKt).

By Theorem 8, we have that $\neg$ Mild-1-Sided-Error-Approx-pKt holds, which by the above implication yields that $\neg$ 2-Sided-Error-Approx-pKt holds. Then by Theorem 7, we get that infinitely-often one-way functions exist.

On the other hand, suppose infinitely-often one-way functions exist. It follows from Theorem 7 that 2-Sided-Error-Approx-pKt does not hold. This trivially implies that (2-Sided-Error-Approx-pKt $\Rightarrow$ Mild-1-Sided-Error-Approx-pKt). □

5.4 PSPACE-Relativization Barrier for One-Sided to Two-Sided Error Reductions

Here, we prove Theorem 4, which is restated below.

Theorem 4. *There exists an oracle $\mathcal{O} \in$ PSPACE relative to which* 2-Sided-Error-Approx-pKt *is true, but* Mild-1-Sided-Error-Approx-pKt *is false.*

To be precise, by saying that 2-Sided-Error-Approx-pKt is true relative to an oracle $\mathcal{O}$ (or pKt can be approximated on average with two-sided error in polynomial time relative to $\mathcal{O}$), we mean the following.

> For every distribution family $\{\mathcal{D}_n\}_n$ supported over $\{0,1\}^n$ that is samplable in polynomial time given oracle access to $\mathcal{O}$, and every polynomial q, there exist a PPT oracle algorithm A and a constant $c > 0$ such that for all sufficiently large n and all $1 \le s \le n + O(\log n)$,
>
> $$\Pr_{x\sim\mathcal{D}_n, A}\left[A^{\mathcal{O}} \text{ decides } \mathsf{GapMpKtP}^{\mathcal{O}}[\tau] \text{ on } (x, 1^s)\right] \ge 1 - \frac{1}{q(n)},$$
>
> where $\tau(n) = c \cdot \log n$, and $\mathsf{GapMpKtP}^{\mathcal{O}}[\tau]$ is the following promise problem $(\mathsf{YES}, \mathsf{NO})$:
>
> $$\mathsf{YES} := \left\{(x, 1^s) \mid \mathsf{pKt}^{\mathcal{O}}(x) \le s\right\},$$
> $$\mathsf{NO} := \left\{(x, 1^s) \mid \mathsf{pKt}^{\mathcal{O}}(x) > s + \tau(|x|)\right\}.$$

Here $\mathsf{pKt}^{\mathcal{O}}(x)$ denotes the variant of pKt complexity where the universal machine is an oracle machine with access to $\mathcal{O}$.

Similarly, we say that Mild-1-Sided-Error-Approx-pKt is true relative to an oracle $\mathcal{O}$ (or pKt can be approximated on average with mild-one-sided error in polynomial time relative to $\mathcal{O}$) if the following holds.

> There is $\varepsilon > 0$ and a PPT oracle algorithm B such that, for every large enough n, the following hold.
>
> - If $x \in \{0,1\}^n$ and $\mathsf{pKt}^{\mathcal{O}}(x) \le n^{\varepsilon}$, then $\mathbf{Pr}_B[B^{\mathcal{O}}(x) = 1] \ge \frac{2}{3}$.
> - With probability at least $1/n$ over $x \sim \{0,1\}^n$, $\mathbf{Pr}_B[B^{\mathcal{O}}(x) = 0] \ge \frac{2}{3}$.

Proof of Theorem 4. The proof follows directly from Lemma 55 and Lemma 60, which are stated and proved below. □

2-Sided-Error-Approx-pKt is true relative to some oracle $\mathcal{O} \in \mathsf{PSPACE}$.

Lemma 55. *There exists an oracle $\mathcal{O} \in \mathsf{PSPACE}$ relative to which* 2-Sided-Error-Approx-pKt *is true.*

To show Lemma 55, we need a "relativizing" version of Lemma 50.

Lemma 56. *Let $\mathcal{O}$ be any oracle. If infinitely-often one-way functions do not exist relative to $\mathcal{O}$,*[6] *then* 2-Sided-Error-Approx-pKt *is true relative to $\mathcal{O}$.*

Proof Sketch. The proof is essentially the same as that of Lemma 50, except that we will use the relativizing versions of Theorem 11, Lemma 14, Theorem 16, which can be obtained by adapting their original proofs in a straightforward manner. We stated them formally below.

Lemma 57 (Following [IL90, IL89])**.** *Let $\mathcal{O}$ be any oracle. If infinitely-often one-way functions do not exist relative to $\mathcal{O}$. then for every distribution family $\{\mathcal{D}_n\}_n$ samplable in polynomial time given oracle access to $\mathcal{O}$, and every polynomial* 1*, there exists a probabilistic polynomial-time oracle algorithm B such that for all $n \in \mathbb{N}$,*

$$\Pr_{x\sim\mathcal{D}_n,B}\left[\frac{\mathcal{D}_n(x)}{2} \le B^{\mathcal{O}}(1^n, x) \le \mathcal{D}_n(x)\right] \ge 1 - \frac{1}{q(n)}.$$

Lemma 58 (See, e.g., the proof of [HIL+23, Lemma 9])**.** *Let $\mathcal{O}$ be any oracle. 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 for all $n \in \mathbb{N}$,*

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

Lemma 59 (Following [LOZ22])**.** *Let $\mathcal{O}$ be any oracle. For every distribution family $\{\mathcal{D}_n\}_n$ samplable in polynomial time given oracle access to $\mathcal{O}$, where each $\mathcal{D}_n$ is supported over $\{0,1\}^n$, there exists a polynomial p such that for every $x \in \mathsf{Support}(\mathcal{D}_n)$,*

$$\mathsf{pK}^{\mathcal{O},p(n)}(x) \le \log\frac{1}{\mathcal{D}_n(x)} + \log p(n).$$

Given the above lemmas, one can easily adapt the proof of Lemma 50 to get Lemma 56. We omit the details here. □

We are now ready to prove Lemma 55.

Proof of Lemma 55. A classical work by Baker, Gill and Solovay [BGS75] showed that there exists an oracle $\mathcal{O} \in \mathsf{PSPACE}$ relative to which $\mathsf{NP} = \mathsf{P}$. It follows that relative to the same oracle, infinitely-often one-way functions do not exits. Then by Lemma 56, 2-Sided-Error-Approx-pKt is true relative to the same oracle $\mathcal{O}$. □

[6] This means that for every $f = \{f_n\} \in \mathsf{FP}^{\mathcal{O}}$, where $f_n\colon \{0,1\}^n \to \{0,1\}^{\mathsf{poly}(n)}$, there exists a PPT algorithm A and a constant $c > 0$ such that for infinitely-many $n \in \mathbb{N}$, $\mathbf{Pr}_{A,\, x\sim\{0,1\}^n}[A^{\mathcal{O}}(f(x)) \in f^{-1}(f(x))] \ge 1/n^c$.

Mild-1-Sided-Error-Approx-pKt **is false relative to every oracle** $\mathcal{O} \in \mathsf{PSPACE}$**.**

Lemma 60. *Let $\mathcal{O}$ be any oracle in* PSPACE*. Then* Mild-1-Sided-Error-Approx-pKt *is false relative to $\mathcal{O}$ (i.e., $\neg$* Mild-1-Sided-Error-Approx-pKt *holds relative to $\mathcal{O}$). Moreover, the corresponding lower bound also holds against randomized ($\mathcal{O}$-oracle) algorithms running in time $n^{\mathsf{poly}(\log n)}$.*

To show Lemma 60, we first need the following "relativizing" versions of Lemma 52 and Lemma 52.

Lemma 61. *Let $\mathcal{O}$ be any oracle. If* pKt *can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$ relative to $\mathcal{O}$, then $\mathsf{BPE}^{\mathcal{O}} \subseteq \mathsf{SIZE}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$.*

Lemma 62. *Let $\mathcal{O}$ be any oracle. If* pKt *can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$ relative to $\mathcal{O}$, then $\mathsf{PSPACE} \subseteq \mathsf{BPTIME}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$.*

Again, the proofs of Lemma 61 and Lemma 62 can be easily adapted from those of Lemma 52 and Lemma 52, and we omit the details here.

Next, we show a lemma analogous to Lemma 63. Note that here we only consider oracles in PSPACE. This allows us to establish the existence of a language in $\mathsf{DSPACE}[2^{n^{o(1)}}]\backslash\mathsf{SIZE}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$ instead of $\mathsf{DSPACE}^{\mathcal{O}}[2^{n^{o(1)}}]\backslash\mathsf{SIZE}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$, which will be crucial for proving Lemma 60 later.

Lemma 63. *Let $\mathcal{O} \in$* PSPACE*. There is a language in $\mathsf{DSPACE}[2^{n^{o(1)}}]\backslash\mathsf{SIZE}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$.*

Proof Sketch. This can be shown using a diagonalization argument, which is similar to the proof of Lemma 54. More specifically, for $n \in \mathbb{N}$, we use one input of length n to diagonalize against at least half of the $\mathcal{O}$-oracle circuits whose size are at most $n^{\mathsf{poly}(\log n)}$. (To be more precise, we will consider the set of size-$s(n)$ $\mathcal{O}$-oracle circuits where s is some super-quasi-polynomial size function.)

The reason this diagonalization argument works is because $\mathcal{O} \in \mathsf{PSPACE}$, and any $\mathcal{O}$-oracle circuit can only query $\mathcal{O}$ on inputs of length at most its size. Therefore, we can simulate any such circuit using $2^{n^{o(1)}}$ space . □

Next, we prove Lemma 60.

Proof of Lemma 60. The proof follows closely to that of Theorem 8.

Suppose, for the sake of contradiction, there exists an oracle $\mathcal{O} \in \mathsf{PSPACE}$ such that pKt can be approximated on average with mild-one-sided error in time $n^{\mathsf{poly}(\log n)}$ relative to $\mathcal{O}$. First of all, by Lemma 61, we get that $\mathsf{BPE}^{\mathcal{O}} \subseteq \mathsf{SIZE}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$.

By Lemma 63, there is a language $L \in \mathsf{DSPACE}[2^{n^{o(1)}}]\backslash\mathsf{SIZE}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$. Now by Lemma 62, we get that $\mathsf{PSPACE} \subseteq \mathsf{BPTIME}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$. Then by a padding argument, we get that $L \in \mathsf{BPE}^{\mathcal{O}}$. However, this means that $\mathsf{BPE}^{\mathcal{O}} \not\subseteq \mathsf{SIZE}^{\mathcal{O}}[n^{\mathsf{poly}(\log n)}]$. A contradiction. □

References

[AGGM06] Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on NP-hardness. In *Symposium on Theory of Computing* (STOC), pages 701–710, 2006.

[All92] Eric Allender. Applications of time-bounded Kolmogorov complexity in complexity theory. In *Kolmogorov complexity and computational complexity*, pages 4–22. Springer, 1992.

[All01] Eric Allender. When worlds collide: Derandomization, lower bounds, and kolmogorov complexity. In *Conference on Foundations of Software Technology and Theoretical Computer Science* (FSTTCS), pages 1–15, 2001.

[All17] Eric Allender. The complexity of complexity. In *Computability and Complexity*, pages 79–94. Springer, 2017.

[BB15] Andrej Bogdanov and Christina Brzuska. On basing size-verifiable one-way functions on NP-hardness. In *Theory of Cryptography Conference* (TCC), pages 1–6, 2015.

[BFNW93] László Babai, Lance Fortnow, Noam Nisan, and Avi Wigderson. BPP has subexponential time simulations unless EXPTIME has publishable proofs. *Comput. Complex.*, 3:307–318, 1993.

[BGS75] Theodore P. Baker, John Gill, and Robert Solovay. Relativizatons of the $\mathsf{P} =? \mathsf{NP}$ Question. *SIAM J. Comput.*, 4(4):431–442, 1975.

[BLMP23] Marshall Ball, Yanyi Liu, Noam Mazor, and Rafael Pass. Kolmogorov comes to cryptomania: On interactive Kolmogorov complexity and key-agreement. In *Symposium on Foundations of Computer Science* (FOCS), 2023.

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

[CIKK16] Marco L. Carmosino, Russell Impagliazzo, Valentine Kabanets, and Antonina Kolokolova. Learning algorithms from natural proofs. In *Conference on Computational Complexity* (CCC), pages 10:1–10:24, 2016.

[CLLO21] Lijie Chen, Zhenjian Lu, Xin Lyu, and Igor C. Oliveira. Majority vs. approximate linear sum and average-case complexity below NC^1. In *International Colloquium on Automata, Languages, and Programming* (ICALP), pages 51:1–51:20, 2021.

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Trans. Inf. Theory*, 22(6):644–654, 1976.

[GK22] Halley Goldberg and Valentine Kabanets. A simpler proof of the worst-case to average-case reduction for polynomial hierarchy via symmetry of information. *Electron. Colloquium Comput. Complex.*, TR22-007, 2022.

[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.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.

[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.

[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.

[Hir22a] Shuichi Hirahara. Np-hardness of learning programs and partial MCSP. In *Symposium on Foundations of Computer Science* (FOCS), pages 968–979, 2022.

[Hir22b] Shuichi Hirahara. Symmetry of information from meta-complexity. In *Computational Complexity Conference* (CCC), pages 26:1–26:41, 2022.

[HLR23] Shuichi Hirahara, Zhenjian Lu, and Hanlin Ren. Bounded relativization. In *Conference on Computational Complexity* (CCC), pages 6:1–6:45, 2023.

[HN23] Shuichi Hirahara and Mikito Nanashima. Learning in pessiland via inductive inference. In *Symposium on Foundations of Computer Science* (FOCS), 2023.

[HS22] Shuichi Hirahara and Rahul Santhanam. Errorless versus error-prone average-case complexity. In *Innovations in Theoretical Computer Science Conference* (ITCS), pages 84:1–84:23, 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 Structure in Complexity Theory* (CCC), 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.*, TR21-082, 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.

[IW97] Russell Impagliazzo and Avi Wigderson. $P = BPP$ if E requires exponential circuits: Derandomizing the XOR lemma. In Frank Thomson Leighton and Peter W. Shor, editors, *Symposium on Theory of Computing* (STOC), pages 220–229, 1997.

[KvM02] Adam R. Klivans and Dieter van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM J. Comput.*, 31(5):1501–1526, 2002.

[Lev74] Leonid A. Levin. Laws of information conservation (nongrowth) and aspects of the foundation of probability theory. *Problemy Peredachi Informatsii*, 10(3):30–35, 1974.

[Lev84] Leonid A. Levin. Randomness conservation inequalities; information and independence in mathematical theories. *Information and Control*, 61(1):15–37, 1984.

[LM93] Luc Longpré and Sarah Mocas. Symmetry of information and one-way functions. *Inf. Process. Lett.*, 46(2):95–100, 1993.

[LO22] Zhenjian Lu and Igor C. Oliveira. Theory and applications of probabilistic Kolmogorov complexity. *Bull. EATCS*, 137, 2022.

[LOZ22] Zhenjian Lu, Igor Carboni 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≠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.

[LP23a] Yanyi Liu and Rafael Pass. On one-way functions and the worst-case hardness of time-bounded Kolmogorov complexity. *Electron. Colloquium Comput. Complex.*, TR23-103, 2023.

[LP23b] Yanyi Liu and Rafael Pass. One-way functions and the hardness of (probabilistic) time-bounded Kolmogorov complexity w.r.t. samplable distributions. In *International Cryptology Conference* (CRYPTO), pages 645–673, 2023.

[LV19] Ming Li and Paul M. B. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications, 4th Edition*. Texts in Computer Science. Springer, 2019.

[LW95] Luc Longpré and Osamu Watanabe. On symmetry of information and polynomial time invertibility. *Inf. Comput.*, 121(1):14–22, 1995.

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

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *J. Cryptol.*, 4(2):151–158, 1991.

[Oli19] Igor C. Oliveira. Randomness and intractability in Kolmogorov complexity. In *International Colloquium on Automata, Languages, and Programming* (ICALP), pages 32:1–32:14, 2019.

[OS17a] Igor C. Oliveira and Rahul Santhanam. Conspiracies between learning algorithms, circuit lower bounds, and pseudorandomness. In *Computational Complexity Conference* (CCC), pages 18:1–18:49, 2017.

[OS17b] Igor C. Oliveira and Rahul Santhanam. Pseudodeterministic constructions in subexponential time. In *Symposium on Theory of Computing* (STOC), pages 665–677, 2017.

[Rom90] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *Symposium on Theory of Computing* (STOC), pages 387–394, 1990.

[Ron04] Detlef Ronneburger. *Kolmogorov Complexity and Derandomization*. PhD thesis, Rutgers University, 2004.

[RR97] Alexander A. Razborov and Steven Rudich. Natural proofs. *J. Comput. Syst. Sci.*, 55(1):24–35, 1997.

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

[STV01] Madhu Sudan, Luca Trevisan, and Salil P. Vadhan. Pseudorandom generators without the XOR lemma. *J. Comput. Syst. Sci.*, 62(2):236–266, 2001.

[SUV17] Alexander Shen, Vladimir A. Uspensky, and Nikolay Vereshchagin. *Kolmogorov complexity and algorithmic randomness.* American Mathematical Society, 2017.

[TV07] Luca Trevisan and Salil P. Vadhan. Pseudorandomness and average-case complexity via uniform reductions. *Comput. Complex.*, 16(4):331–364, 2007.

[ZL70] Alexander K. Zvonkin and Leonid A. Levin. The complexity of finite objects and the algorithmic concepts of randomness and information. *UMN (Russian Math. Surveys)*, 25(6):83–124, 1970.

A One-Way Functions and Worst-Case SoI for pKt with Computational Depth

For a time bound $t \in \mathbb{N}$, the computational depth of x, denoted by $\mathsf{cd}^t(x)$, is defined as

$$\mathsf{cd}^t(x) := \mathsf{pK}^t(x) - \mathsf{K}(x).$$

We say that a string is computationally shallow if its computational depth is small. The main result of this section is to characterize the existence of a one-way function by symmetry of information for computationally shallow strings.

Theorem 64. *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*
2. **(Worst-Case Symmetry of Information using Computational Depth)** *There exist a constant c and a polynomial ρ such that for all $t, n \in \mathbb{N}$ and $x, y \in \{0,1\}^n$ such that $t \geq \rho(n)$,*

$$\mathsf{pKt}(x,y) \geq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \big(\mathsf{cd}^t(x,y) + \log t\big).$$

3. **(Average-Case Symmetry of Information)** *For every 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 every polynomial q, there exists a constant c such that for all n,*

$$\mathop{\mathbf{Pr}}_{(x,y)\sim\mathcal{D}_n}[\mathsf{pKt}(x,y) \geq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \log n] \geq 1 - \frac{1}{q(n)}.$$

Proof. The theorem follows directly from Theorem 65 (stated and proved in Sections A.1), Proposition 72 and Lemma 39, respectively. □

A.1 Worst-Case SoI with Computational Depth from Inverting One-Way Functions

Theorem 65. (Item 1 ⇒ Item 2 in Theorem 64). *If infinitely-often one-way functions do not exist, then there exist a constant c and a polynomial ρ such that for all $t, n \in \mathbb{N}$ and $x, y \in \{0,1\}^n$ such that $t \geq \rho(n)$,*

$$\mathsf{pKt}(x,y) \geq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \big(\mathsf{cd}^t(x,y) + \log t\big).$$

We will need a couple of the technical lemmas. The following theorem shows that there exists an efficient algorithm that approximates $\mathsf{K}(x)$ for every computationally shallow string x under the non-existence of one-way functions.

Theorem 66 (See [HN23, Theorem 7.2])**.** *The following are equivalent.*

1. *Infinitely-often one-way functions do not exist.*
2. *There exists a randomized polynomial-time algorithm M such that for all $n, k, t, \alpha \in \mathbb{N}$ and all $x \in \{0,1\}^n$ with $\mathsf{cd}^t(x) \leq \alpha$,*

$$\Pr_M\Big[\mathsf{K}(x) \leq M(x, 1^k, 1^t, 1^{2^\alpha}) \leq \mathsf{K}(x) + \alpha + O(\log tk)\Big] \geq 1 - \frac{1}{k}.$$

An important property of computational depth is that any efficient randomized algorithm cannot significantly increase the computational depth.

Lemma 67 (Slow growth law [HN23, Lemma 6.15])**.** *Let M be any randomized polynomial-time algorithm of description length $d \in \mathbb{N}$. Then, there exist polynomials τ, ρ such that for all $n, t, k \in \mathbb{N}$ and all $x \in \{0,1\}^n$ such that $t \geq \rho(d, n)$,*

$$\Pr_M\Big[\mathsf{cd}^{\tau(t)}(M(x)) \leq \mathsf{cd}^t(x) + k\Big] \geq 1 - 2^{-k + \log t + O(d)},$$

where the probability is taken over the internal randomness of M.

Using the symmetry of information for K, we upgrade the algorithm of Theorem 66 to one that approximates the conditional Kolmogorov complexity $\mathsf{K}(y \mid x)$ for every computationally shallow pair (y, x).

Lemma 68. *If infinitely-often one-way functions do not exist, then there exists a randomized polynomial-time algorithm M such that for all $n, m, k, t, \alpha \in \mathbb{N}$ and all $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$ with $\mathsf{cd}^t(x,y) \leq \alpha$,*

$$\Pr_M\Big[\mathsf{K}(y \mid x) - 2\alpha - O(\log tknm) \leq M(y, x, 1^k, 1^t, 1^{2^\alpha}) \leq \mathsf{K}(y \mid x)\Big] \geq 1 - \frac{1}{k}.$$

Proof Sketch. By the slow growth law (Lemma 67), we have

$$\mathsf{cd}^{\rho(t)}(x) \leq \mathsf{cd}^t(x,y) + O(\log t)$$

for every $x \in \{0,1\}^n, y \in \{0,1\}^m$ such that $n + m \geq \rho(t)$. Using the algorithm from Theorem 66, we can first compute u such that

$$\mathsf{K}(x,y) \leq u \leq \mathsf{K}(x,y) + \alpha + O(\log t + \log k),$$

and then compute v such that

$$\mathsf{K}(x) \le v \le \mathsf{K}(x) + \alpha + O(\log t + \log k).$$

Then using symmetry of information for K [ZL70], we have

$$u - v \ge \mathsf{K}(x, y) - \mathsf{K}(x) - \alpha - O(\log t + \log k) \ge \mathsf{K}(y \mid x) - \alpha - O(\log tknm),$$

and

$$u - v \le \mathsf{K}(x, y) + \alpha + O(\log t + \log k) - \mathsf{K}(x) \le \mathsf{K}(y \mid x) + \alpha + O(\log t + \log k).$$

We can then output $u - v - \alpha - O(\log tknm)$, which will be a proper approximation of $\mathsf{K}(y \mid x)$. □

The proof strategy of Lemma 37 is to break the security of a pseudorandom generator construction, which is called the direct product generator.

Definition 69 (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

$$\mathsf{DP}_k(x; z_1, \dots, z_k) := (z_1, \dots, z_k, x \cdot z_1, \dots, x \cdot z_k),$$

where $x \cdot z_i \in \{0,1\}$ denotes the inner product between x and z_i modulo 2.

It is known that $\mathsf{DP}_k(x; \mathcal{U})$ is pseudorandom if x has high Kolmogorov complexity.

Lemma 70 (Reconstruction Lemma for pK^t; see [GKLO22])**.** *There exists a polynomial p_{DP} such that, for every $\varepsilon > 0$, $x \in \{0,1\}^n$, $s \in \mathbb{N}$, and $k \in \mathbb{N}$ satisfying $k \le 2n$, for every randomized algorithm D that takes an advice string β and runs in time t_D such that*

$$\Pr_{z,D}[D(\mathsf{DP}_k(x; z)) = 1] - \Pr_{w,D}[D(w) = 1] \ge \varepsilon,$$

where the probabilities are taken over $z \sim \{0,1\}^{nk}$, $w \sim \{0,1\}^{nk+k}$, and the internal randomness of D, it holds that

$$\mathsf{pK}^{p_{\mathsf{DP}}(nt_D/\varepsilon)}(x \mid \beta) \le k + \log p_{\mathsf{DP}}(nt_D/\varepsilon).$$

We now combine Lemmas 68 and 70 to show an upper bound of pKt.

Lemma 71. *If infinitely-often one-way functions do not exist, then there exist a constant $c > 0$ and a polynomial ρ such that for all $n, t \in \mathbb{N}$ and all $x, y \in \{0,1\}^n$ such that $t \ge \rho(n)$,*

$$\mathsf{pKt}(y \mid x) \le \mathsf{K}(y \mid x) + c \cdot \left(\mathsf{cd}^t(x, y) + \log t\right).$$

Proof. Let $x, y \in \{0,1\}^n$. Let $k \le 2n$ be a parameter chosen later. Let $\epsilon := 1/10$.

Recall the definition of direct product generator in Definition 69. Since DP_k is computable, we have for all $z \in \{0,1\}^{nk}$,

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

where we define s as in the last equality.

Applying Lemma 67 to the randomized algorithm that outputs $(\mathsf{DP}_k(x;z),y)$ on input (x,y) over a coin flip sequence $z \sim \{0,1\}^{nk}$, we obtain polynomials ρ, τ such that

$$\mathsf{cd}^{\tau(t)}(\mathsf{DP}_k(x;z),y) \leq \mathsf{cd}^t(x,y) + O(\log t) =: \alpha \tag{25}$$

with probability at least $1-\epsilon$ over $z \sim \{0,1\}^{nk}$ for all $t \geq \rho(n)$. We define $\alpha := \mathsf{cd}^t(x,y) + O(\log t)$ so that $\mathsf{cd}^{\tau(t)}(\mathsf{DP}_k(x;z),y) \leq \alpha$ with probability $1-\epsilon$. Similarly, we also obtain

$$\mathsf{cd}^{\tau(t)}(w,y) \leq \mathsf{cd}^t(x,y) + O(\log t) \leq \alpha \tag{26}$$

with probability at least $1-\epsilon$ over $w \sim \{0,1\}^{nk+k}$.

Let M be the algorithm in Lemma 68 and let M' be the algorithm such that $M'(w,x) := M(w,x,1^{\epsilon^{-1}},1^{\tau(t)},1^{2^\alpha})$. By Lemma 68 and Equation (25), with probability at least $1-2\epsilon$ over the internal randomness of M' and $z \sim \{0,1\}^{nk}$, we have

$$M'(\mathsf{DP}_k(y;z),x) \leq \mathsf{K}(\mathsf{DP}_k(y;z) \mid x) \leq s.$$

Let D be a randomized algorithm that takes non-uniform advice $s,t,\alpha \in \mathbb{N}$ and $x \in \{0,1\}^n$, and outputs 1 on input $w \in \{0,1\}^{nk+k}$ if and only if $M'(w,x) \leq s$. Then we have

$$\Pr_{z,D}[D(\mathsf{DP}_k(y;z)) = 1] \geq 1 - 2\epsilon, \tag{27}$$

where the probability is over the internal randomness of D and $z \sim \{0,1\}^{nk}$.

On the other hand, by Lemma 68 and Equation (26), with probability at least $1-2\epsilon$ over the internal randomness of M' and $w \sim \{0,1\}^{nk+k}$, we also have

$$M'(w,x) \geq \mathsf{K}(w \mid x) - 2\alpha - O(\log t)$$

Moreover, by a simple counting argument, we have

$$\mathsf{K}(w \mid y) \geq |w| - O(\log(1/\epsilon)) \geq nk + k - O(\log(1/\epsilon))$$

with probability at least $1-\epsilon$ over $w \sim \{0,1\}^{nk+k}$. By a union bound, with probability at least $1-3\epsilon$, we obtain

$$M'(w,x) \geq nk + k - 2\alpha - O(\log t) > s,$$

where the last inequality holds by choosing a sufficiently large $k := \mathsf{K}(y \mid x) + 2\alpha + O(\log t)$. Then we have

$$\Pr_{w,D}[D(w) = 1] \leq 3\epsilon, \tag{28}$$

where the probability is over the internal randomness of D and $w \sim \{0,1\}^{nk+k}$.

By Equations (27) and (28), the non-uniform randomized algorithm D distinguishes the output distribution of $\mathsf{DP}_k(x;\cdot)$ from the uniform distribution. By Lemma 70, we obtain

$$\mathsf{pK}^{t_D}(x) \leq k + \log t_D \leq \mathsf{K}(y \mid x) + 2\alpha + O(\log t_D),$$

where $t_D \leq \mathsf{poly}(t, 2^\alpha)$ is a sufficiently large upper bound of the running time of D. Finally, we have

$$\mathsf{pKt}(x) \leq \log t_D + \mathsf{pK}^{t_D}(x) \leq \mathsf{K}(y \mid x) + O(\alpha + \log t).$$

□

Proof of Theorem 65. We apply Lemma 71 twice for (x, y) and (the empty string, x). By Lemma 71, there exist a constant $c > 0$ and a polynomial ρ such that for all $n, t \in \mathbb{N}$ and all $x, y \in \{0,1\}^n$ such that $t \geq \rho(n)$,

$$\mathsf{pKt}(y \mid x) \leq \mathsf{K}(y \mid x) + c \cdot \left(\mathsf{cd}^t(x, y) + \log t\right).$$

Similarly, we also have

$$\mathsf{pKt}(x) \leq \mathsf{K}(x) + c \cdot \left(\mathsf{cd}^{p(t)}(x) + \log p(t)\right),$$

where p is an arbitrary polynomial. By Lemma 67, we have $\mathsf{cd}^{p(t)}(x) \leq \mathsf{cd}^t(x, y) + O(\log t)$. It follows from the three inequalities above that

$$\begin{aligned}\mathsf{pKt}(y \mid x) + \mathsf{pKt}(x) &\leq \mathsf{K}(y \mid x) + \mathsf{K}(x) + O\big(\mathsf{cd}^t(x, y) + \log t\big) \\ &\leq \mathsf{K}(x, y) + O(\mathsf{cd}^t(x, y) + \log t),\end{aligned}$$

where the last inequality holds because of the symmetry of information for K. The result follows by observing that $\mathsf{K}(x, y) \leq \mathsf{pKt}(x, y) + O(1)$. □

A.2 Average-Case SoI from Worst-Case SoI with Computational Depth

Proposition 72. (Item 2 ⇒ Item 3 in Theorem 64). *If for every polynomial ρ, there exists a constant c such that for all $n \in \mathbb{N}$ and $x, y \in \{0,1\}^n$,*

$$\mathsf{pKt}(x, y) \geq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \big(\mathsf{cd}^t(x, y) + \log n\big),$$

where $t := \rho(n)$, then average-case symmetry of information for pKt (Item 3 in Theorem 64) holds.

It is easy to observe that Item 2 of Theorem 64 implies the assumption of Proposition 72. Thus, (Item 2 ⇒ Item 3) in Theorem 64 follows from Proposition 72.

The proof of Proposition 72 is based on the fact that a string drawn from an arbitrary polynomial-time samplable distribution is computationally shallow.

Lemma 73 (see also [HN23, Lemma 6.14])**.** *For every polynomial-time samplable distribution $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, there exists a polynomial ρ such that for every $n \in \mathbb{N}$, every $t \geq \rho(n)$, and every $k \in \mathbb{N}$,*

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

Proof. Let M be the randomized algorithm that, on input 1^n, samples a string distributed according to $\mathcal{D}_n$. The result follows by applying Lemma 67 to M. □

Proof of Proposition 72. Let $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ be any polynomial-time samplable distribution. By Lemma 73, with probability at least $1 - 1/q(n)$ over $(x, y) \sim \mathcal{D}_n$, it holds that

$$\mathsf{cd}^t(x, y) \leq O(\log n).$$

Thus, it follows that

$$\begin{aligned}\mathsf{pKt}(x, y) &\geq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - c \cdot \big(\mathsf{cd}^t(x, y) + \log n\big) \\ &\geq \mathsf{pKt}(x) + \mathsf{pKt}(y \mid x) - O(\log n),\end{aligned}$$

as desired. □

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il