Consequences of Randomized Reductions from SAT to Time-Bounded Kolmogorov Complexity

Halley Goldberg
Simon Fraser University, Burnaby, Canada

Valentine Kabanets
Simon Fraser University, Burnaby, Canada

Abstract

A central open question within meta-complexity is that of NP-hardness of problems such as MCSP and $\mathsf{MK^tP}$. Despite a large body of work giving consequences of and barriers for NP-hardness of these problems under (restricted) deterministic reductions, very little is known in the setting of randomized reductions. In this work, we give consequences of randomized NP-hardness reductions for both approximating and exactly computing time-bounded and time-unbounded Kolmogorov complexity.

In the setting of *approximate* $\mathsf{K^{poly}}$ complexity, our results are as follows.

1. Under a derandomization assumption, for any constant $\delta > 0$, if approximating K^t complexity within n^δ additive error is hard for SAT under an honest randomized non-adaptive Turing reduction running in time polynomially less than t, then NP = coNP.
2. Under the same assumptions, the worst-case hardness of NP is equivalent to the existence of one-way functions.

Item 1 above may be compared with a recent work of Saks and Santhanam [39], which makes the same assumptions except with $\omega(\log n)$ additive error, obtaining the conclusion NE = coNE.

In the setting of *exact* $\mathsf{K^{poly}}$ complexity, where the barriers of Item 1 and [39] do not apply, we show:

3. If computing K^t complexity is hard for SAT under reductions as in Item 1, then the average-case hardness of NP is equivalent to the existence of one-way functions. That is, "Pessiland" is excluded.

Finally, we give consequences of NP-hardness of *exact time-unbounded* Kolmogorov complexity under randomized reductions.

4. If computing Kolmogorov complexity is hard for SAT under a randomized many-one reduction running in time t_R and with failure probability at most $1/(t_R)^{16}$, then coNP is contained in non-interactive statistical zero-knowledge; thus NP $\subseteq$ coAM. Also, the worst-case hardness of NP is equivalent to the existence of one-way functions.

We further exploit the connection to NISZK along with a previous work of Allender et al. [7] to show that hardness of K complexity under randomized many-one reductions is highly robust with respect to failure probability, approximation error, output length, and threshold parameter.

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

Keywords and phrases Meta-complexity, Randomized reductions, NP-hardness, Worst-case complexity, Time-bounded Kolmogorov complexity

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2024.51

Category RANDOM

Related Version *Full Version*: https://eccc.weizmann.ac.il/report/2024/120/ [15]

Funding *Halley Goldberg*: Supported by NSERC CGS D.
Valentine Kabanets: Supported by NSERC Discovery research grant.

1 Introduction

Meta-complexity aims to determine the computational complexity of the tasks to compute various intrinsic complexity measures of given binary strings. Two prominent examples of such complexity measures are the minimum circuit size of a given truth table of a Boolean function, and the minimum time-bounded Kolmogorov complexity (denoted K^t) of a given binary string. The corresponding meta-complexity problems are the *Minimum Circuit Size Problem* (MCSP):

> given a binary string $x \in \{0,1\}^{2^n}$ and a parameter $s \leq 2^n$, decide if there is an n-input boolean circuit of size at most s whose truth table equals x,

and the *Minimum* K^t *Problem* ($\mathsf{MK}^t\mathsf{P}$):

> given a binary string $x \in \{0,1\}^n$, and a parameter $s \leq n$, decide if there is a binary input w of length at most s such that some fixed universal Turing machine U on input w prints x within t time steps.

The history of these two problems goes back to at least the 1950s and '60s. In the Soviet Union, during that period, those involved in "theoretical cybernetics" were keenly interested in problems related to switching circuits and Kolmogorov's new theory of the complexity of strings. It was widely suspected that one could not avoid *perebor* (exhaustive search) in the solution of the corresponding minimization problems. Levin's interest in perebor, culminating in his discovery of NP-completeness in the early 1970s, was motivated in particular by questions about the complexity of time-bounded Kolmogorov complexity [40]. Since then, both MCSP and $\mathsf{MK}^t\mathsf{P}$ have resisted categorization as efficiently decidable or as NP-complete, a somewhat uncommon state of affairs for natural problems in NP.

In 2000, Kabanets and Cai took up the study of circuit minimization again, with a result suggesting that NP-hardness of MCSP may be very difficult to resolve: if MCSP is NP-hard under a deterministic many-one reduction such that output length depends only on input length, then one gets the lower bound $\mathsf{E} \not\subseteq \mathsf{P/poly}$ [32]. At least, if MCSP is NP-hard, then showing its hardness would seem to require different techniques than those applied in the past, barring any further major breakthroughs. A line of work has continued to push further in this negative direction, progressively obtaining (1) "stronger" consequences, and (2) consequences of NP-hardness under more powerful forms of reducibility. An example of the former is a result of Murray and Williams, which obtains $\mathsf{NP} \not\subseteq \mathsf{P/poly}$ from NP-hardness of MCSP under log-time uniform AC^0 reductions [35]. An example of the latter is a result of Hitchcock and Pavan, which obtains $\mathsf{EXP} \neq \mathsf{ZPP}$ from NP-hardness of MCSP under deterministic non-adaptive Turing reductions [27]. There are many more examples of this kind of work relying essentially on the determinism of the reductions in question; see, e.g., [6, 38, 8, 26].[1]

In contrast to the negative line of work for deterministic reductions, there is a positive line of work obtaining NP-hardness of variants of MCSP and $\mathsf{MK}^t\mathsf{P}$ that seem to come progressively closer to the standard definitions of these problems. Examples include [29, 23, 30, 28]. A common feature of these results is their employment of randomness in the NP-hardness reductions. An impressive example of such a result is Hirahara's recent proof of NP-hardness of partial-function versions of MCSP and $\mathsf{MK}^t\mathsf{P}$ [23]. Additionally, from [5], MCSP is hard

[1] One result of [26] deals with one-query randomized reductions to $\mathsf{MCSP}^{\mathcal{O}}$ working for *every* oracle $\mathcal{O}$, which may be seen as an exception. Other results of that work give consequences of deterministic reductions to, for example, approximating circuit size and Levin's Kt complexity.

for SZK (statistical zero-knowledge) under randomized reductions, which is the strongest unconditional hardness known for MCSP. All of this begs the question whether randomness is the key ingredient for the hardness of problems in meta-complexity: most barriers apply to deterministic reductions, whereas most progress has been made via randomized reductions.

As for the negative direction for randomized reductions, there has been far less headway. In fact, prior to this work, only two such results were known for MCSP and $\mathsf{MK}^t\mathsf{P}$. Murray and Williams ruled out NP-hardness of MCSP in the very restrictive setting of poly-logarithmic-time randomized projections [35]. More recently, Saks and Santhanam showed that NE = coNE if approximating K^t-complexity is NP-hard under randomized non-adaptive polynomial-time reductions (with some caveats, including a derandomization assumption and that the time-bound t in the superscript of K^t must be greater than the running time of the reduction) [39].

Of course, any NP-hardness of $\mathsf{MK}^t\mathsf{P}$ or MCSP would be a major breakthrough for complexity theory, including hardness under a non-black-box reduction. In that sense, the *kind* of reduction in question is hardly important in itself. That being said, obtaining consequences of restricted forms of reduction can certainly help guide the "search for NP-hardness". For example, a recent work of Ilango proved that approximating K^t within $\Omega(n)$ additive error is NP-hard in the random oracle model [29]. As mentioned in that paper, the reduction circumvents the barrier of [39] by requiring more time than the superscript t. As with much of complexity theory, one can always take negative results as putting into focus the space for positive progress.

In this paper, we advance in the negative direction for randomized reductions, obtaining results with stronger consequences and from reductions to harder problems compared to prior work.

2 Main Results

We show a number of consequences of the assumptions that there exist restricted randomized NP-hardness reductions for the exact and approximate variants of the problem to determine the (time-bounded) Kolmogorov complexity of a given binary string.

In addition to the problem $\mathsf{MK}^t\mathsf{P}$ introduced above, we shall also consider its time-unbounded version, MKP, where given a binary string $x \in \{0,1\}^n$ and a threshold parameter $s \leq n$, one needs to decide if there is a string $w \in \{0,1\}^{\leq s}$ such that a fixed universal TM $U(w)$ outputs x. We also consider the probabilistic variant of K^t, denoted by pK^t, where $\mathsf{pK}^t(x)$ is defined as the minimum length s such that, for each of at least $2/3$ of random strings r, there exists some input $w_r \in \{0,1\}^{\leq s}$ such that $U(w_r, r)$ outputs x within t time steps. The corresponding Minimum pK^t Problem is denoted by $\mathsf{MpK}^t\mathsf{P}$. For $g : \mathbb{N} \to \mathbb{N}$ and $\mu \in \{\mathsf{K}, \mathsf{pK}\}$, $\mathsf{Approx}_g\text{-}\mu^t$ refers to the problem of approximating μ^t complexity of a given $x \in \{0,1\}^n$ to within a $g(n)$ additive error. $\mathsf{Approx}_g\text{-}\mathsf{K}[s]$ refers to the problem of approximating K complexity except with threshold parameter fixed to s.

2.1 Consequences of showing the NP-hardness of an approximation to pK^t or K^t

Informally, our first results show that NP-hardness of $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$ under honest non-adaptive randomized reductions with runtime sufficiently smaller than t implies that

- NP $\subseteq$ coAM (and hence, the polynomial-time hierarchy collapses [12]), and
- if, in addition, no one-way functions exist, then NP $\subseteq$ BPP;

here "honest" reductions are those that make queries of length at least some polynomial of the input to the reduction. We also get a similar result for $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$, under a derandomization assumption that E requires exponential-size nondeterministic circuits.

More precisely, we show that under the same NP-hardness assumptions, there is a black-box non-adaptive reduction from SAT to inverting an auxiliary input one-way function.[2] Moreover, this reduction is of a restricted form in which the oracle only needs to invert the function on auxiliary input φ, where φ is the input to SAT; this is called a "fixed-auxiliary-input reduction" [9]. The "γ-honesty" condition below means that all queries $q \in \{0,1\}^*$ made by the reduction are such that $|q| \geq n^\gamma$, where n is the length of the input to the reduction.

▶ **Theorem 1** (Collapsing the Polynomial Hierarchy). *For any constants $\delta, \gamma > 0$, there is a polynomial p such that, for any $t, t_R : \mathbb{N} \to \mathbb{N}$ satisfying $p(t_R(n)) \leq t(n)$ for all $n \in \mathbb{N}$, we have the following.*

1. *If $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$ is hard for SAT under a γ-honest non-adaptive randomized reduction running in time t_R, then there is a black-box non-adaptive fixed-auxiliary-input reduction from SAT to inverting an auxiliary-input OWF. The latter implies that*

   $$\mathsf{NP} \subseteq \mathsf{coAM}.$$

2. *Assume $\mathsf{E} \nsubseteq \mathsf{io}\text{-}\mathsf{NSIZE}[2^{o(n)}]$. If $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$ is hard for SAT under an honest non-adaptive randomized reduction running in time t_R, then*

   $$\mathsf{NP} = \mathsf{coNP}.$$

As a consequence of the above non-adaptive black-box reduction from SAT to inverting an auxiliary-input one-way function, we further obtain from the hypothesis of Theorem 1 that the existence of a standard one-way function can be based on the worst-case hardness of NP. That is, proving NP-hardness of $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$ (under restricted randomized reductions) is as hard as achieving the "holy grail of cryptography".

We obtain both adaptive black-box and non-adaptive BPP-black-box[3] reductions from SAT to the problem of inverting a standard OWF. The former follows immediately from our Theorem 1 and a recent work of Nanashima [36], and the latter is implicit in [24], though we provide a short, self-contained proof building on Theorem 1.

▶ **Theorem 2** (Excluding Pessiland and Heuristica). *For any constants $\delta, \gamma > 0$, there is a polynomial p such that, for any $t_R, t : \mathbb{N} \to \mathbb{N}$ satisfying $p(t_R(n)) \leq t(n)$ for all $n \in \mathbb{N}$, we have the following.*

1. *If $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$ is hard for SAT under a γ-honest non-adaptive randomized reduction running in time t_R, then there exist both (I) a black-box adaptive randomized polynomial-time reduction, and (II) a BPP-black-box non-adaptive randomized polynomial-time reduction, from SAT to inverting a OWF. As a consequence, we get*

   $$\mathsf{NP} \nsubseteq \mathsf{BPP} \iff \exists\, \mathsf{OWF}.$$

2. *Assume $\mathsf{E} \nsubseteq \mathsf{io}\text{-}\mathsf{NSIZE}[2^{o(n)}]$. If $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$ is hard for SAT under an honest non-adaptive randomized reduction running in time t_R, then*

   $$\mathsf{NP} \neq \mathsf{P} \iff \exists\, \mathsf{OWF}.$$

---

[2] We consider auxiliary input functions $f = \{f_\varphi\}_{\varphi \in \{0,1\}^*}$ as defined in [37].

[3] As defined by [18], a BPP-black-box reduction R from a problem L to a problem L' is an efficient oracle Turing machine that correctly decides L, given any oracle $A \in \mathsf{BPP}$ such that A decides L'.

With a similar argument, we also get the following statement for Levin's Kt complexity.

▶ **Corollary 3.** *For any constant $\delta > 0$, we have the following. Assume* $\mathsf{E} \not\subseteq \text{io-NSIZE}[2^{o(n)}]$. *If* $\text{Approx}_{n^\delta}\text{-Kt}$ *is hard for* SAT *under an honest non-adaptive randomized reduction, then* NP = coNP. *Moreover, if no one-way functions exist, then* NP = P.

2.2 Consequences of showing the NP-hardness of K^t

Though the conclusions of Theorems 1 and 2 are incomparable, one may find NP ⊆ coAM unbelievable, in which case Theorem 2 would not appear to yield a promising route for actually excluding Pessiland and Heuristica. Indeed, the earlier barrier result of [39] was part of Hirahara's motivation to introduce a harder "distributional" variant of K^t complexity in a recent work [24], delineating an intact positive approach for excluding Impagliazzo's worlds via NP-hardness of meta-complexity.

As a counterpoint, building on a work of Liu and Pass [33], we show that NP-hardness of *exact* K^t complexity would still suffice to exclude Pessiland while circumventing the barrier of Theorem 1 (and [39]). As noted in [33], problems of exact and approximate K^t complexity are qualitatively different: approximating K^t within $\omega(\log n)$ additive error is unconditionally easy on average (in the "error-prone" sense) over the uniform distribution, but the argument fails in the setting of exact K^t. Thus, there is still room for optimism with regard to excluding Pessiland via NP-hardness of standard K^t complexity.

▶ **Theorem 4** (Excluding Pessiland). *There is a polynomial p such that, for any $t, t_R : \mathbb{N} \to \mathbb{N}$ satisfying $t(n) \geq p(t_R(n))$ for all $n \in \mathbb{N}$, we have the following.*

1. *If* $\mathsf{MpK}^t\mathsf{P}$ *is hard for* SAT *under an honest non-adaptive randomized reduction running in time t_R, then there is a black-box average-case reduction from* SAT *to inverting OWFs. As a consequence, we get that*

   $$\mathsf{DistNP} \not\subseteq \mathsf{HeurBPP} \iff \exists\ \mathsf{OWF}.$$

2. *Assume* $\mathsf{E} \not\subseteq \text{io-NSIZE}[2^{o(n)}]$. *If* $\mathsf{MK}^t\mathsf{P}$ *is hard for* SAT *under an honest non-adaptive randomized reduction running in time t_R, then*

   $$\mathsf{DistNP} \not\subseteq \mathsf{HeurP} \iff \exists\ \mathsf{OWF}.$$

2.3 Consequences of showing the NP-hardness of K

Finally, we show that NP-hardness of Kolmogorov complexity under randomized many-one reductions would imply NP ⊆ coAM and a collapse of the polynomial hierarchy. To the best of our knowledge, this is the first evidence against NP-hardness of exact Kolmogorov complexity under randomized many-one reductions. We also get under the same assumption that if NP ⊈ BPP then one-way functions exist.

▶ **Theorem 5** (Collapsing the Polynomial Hierarchy). *There is a polynomial p such that, for any $t_R : \mathbb{N} \to \mathbb{N}$, we have the following. If* MKP *is hard for* SAT *under a randomized polynomial time many-one reduction running in time $t_R(n)$ and with failure probability at most $1/p(t_R(n))$, then*

$$\mathsf{NP} \subseteq \mathsf{coAM}.$$

If, in addition, no one-way functions exist, then NP ⊆ BPP.

2.4 Robustness of reductions to K

In fact, we can get a stronger result than that stated above: namely, we show that if a decidable language L reduces to MKP as in Theorem 5, then $\overline{L} \subseteq$ NISZK, where NISZK is the class of promise problems admitting *non-interactive* statistical zero-knowledge proofs. In particular, we prove the following.

▶ **Theorem 6.** *For any polynomial t_R and decidable language L, if* MKP *is hard for L under a randomized many-one reduction running in time $t_R(n)$ and with failure probability at most $1/t_R(n)^{16}$, then $\overline{L} \subseteq$* NISZK*.*

Since it is known that NISZK $\subseteq$ SZK $\subseteq$ AM $\cap$ coAM [17, 14, 1], where SZK is the class of problems admitting statistical zero-knowledge proofs, Theorem 6 captures Theorem 5. It also improves on the following statement implicit in a previous work of Allender et al. [7].

▶ **Theorem 7** ([7])**.** *For any decidable language L, if* $\mathsf{Approx}_{\omega(\log n)}$-K$[n/2]$ *is hard for L under an honest randomized many-one reduction with failure probability at most $1/n^{\omega(1)}$, then $\overline{L} \subseteq$* NISZK*.*

Note that Theorem 6 improves on Theorem 7 in three respects: we do not require the reduction to be honest, we do not require an $\omega(\log n)$ approximation term, and we do not require the threshold parameter to be fixed.

Combining the above with a converse provided in [7], we show that hardness of MKP under randomized many-one reductions (with sufficiently small failure probability) is remarkably robust with respect to approximation error, failure probability, honesty, and threshold parameter (fixed or unfixed). For instance, if MKP is NP-hard under a $t_R(n)$-time many-one reduction with failure probability $1/\mathsf{poly}(t_R(n))$, then it is also NP-hard under a polynomial-time many-one reduction with exponentially small failure probability. More specifically,

▶ **Theorem 8.** *There is a polynomial p such that for any decidable language L and polynomial t_R, the following are equivalent.*

1. $\overline{L} \subseteq$ NISZK*;*
2. MKP *is hard for L under a randomized many-one reduction running in time $t_R(n)$ and with two-sided failure probability at most $1/p(t_R(n))$;*
3. $\mathsf{Approx}_{n^{o(1)}}$-K$[n/2]$ *is hard for L under an honest randomized many-one reduction with one-sided failure probability at most $2^{-\mathsf{poly}(n)}$.*

3 Related Work

Saks and Santhanam obtain a barrier result similar to our Theorem 1, Item 2, for the regime of super-logarithmic additive error. Specifically, they prove the following.

▶ **Theorem 9** ([39])**.** *Assume* E $\not\subseteq$ io-NSIZE$[2^{o(n)}]$*. There is a polynomial p satisfying the following. For any $t, t_R : \mathbb{N} \to \mathbb{N}$ such that $p(t_R(n)) \leq t(n)$, if* $\mathsf{Approx}_{\omega(\log n)}$-K^t *is hard for* SAT *under an honest, fixed query length, non-adaptive randomized reduction running in time t_R, then* NE $=$ coNE*.*

Here, "fixed query length" means that the lengths of all queries made in the reduction are identical and depend only on the length of the input to the reduction, independent of randomness. In comparison, at the cost of increasing the approximation error term from $\omega(\log n)$ to n^δ for any constant $\delta > 0$, we obtain the stronger (and presumably less believable) consequence NP $=$ coNP. Moreover, we do not require that the reduction have fixed query

length: in our case, the length of queries need not be the same, and they can depend on the input and the randomness of the reduction. The honesty condition is identical in this work and [39]. We also note that our proof techniques can be made to capture the regime of $\omega(\log n)$ additive error, in which case we recover the statement of [39] improved to reductions without fixed query length.

Our Theorem 2 is related to a recent work of Hirahara [24], which introduces a "distributional" variant of K^t complexity, denoted dK^t, defined as follows: for a string $x \in \{0,1\}^*$, a time bound $t \in \mathbb{N}$, and a distribution $\mathcal{D}$,

$$\mathsf{dK}^t(x \mid \mathcal{D}) = \min_{s \in \mathbb{N}} \left\{ \exists d \in \{0,1\}^s \;\middle|\; \Pr_{r \sim \mathcal{D}}[U(d,r) \text{ halts and outputs } x \text{ within } t \text{ steps}] \geq 2/3 \right\}.$$

Using the techniques of that work, it is possible to recover a part of our Theorem 2 exactly: namely, the existence of a BPP-black-box non-adaptive reduction from SAT to inverting a OWF. This is essentially due to the fact that if, for example, approximating K^t is NP-hard, then approximating dK^t is also NP-hard, since dK^t captures K^t when the provided distribution $\mathcal{D}$ always outputs the empty string. A probabilistic variant of dK^t is also introduced in [24], which similarly generalizes pK^t.

However, our proof of Theorem 2 takes a partly different approach to that implicit in [24]. In particular, though both our proof and that work employ a non-black-box worst-case to average-case reduction as in [19, 20, 16], the latter approach would use this kind of reduction in two places: once to reduce NP to inverting an auxiliary-input one-way function, and once to obtain $\mathsf{NP} \not\subseteq \mathsf{BPP} \implies \mathsf{DistNP} \not\subseteq \mathsf{AvgBPP}$. To accommodate the reduction to inverting an auxiliary-input OWF, Hirahara introduces a new kind of mildly black-box reduction, which is more restrictive than the standard notion of a class-specific black-box reduction [18]. In contrast, as an intermediate step, we obtain a completely black-box non-adaptive reduction from NP to inverting an auxiliary-input OWF. We employ a class-specific worst-to-average reduction only to obtain $\mathsf{NP} \not\subseteq \mathsf{BPP} \implies \mathsf{DistNP} \not\subseteq \mathsf{AvgBPP}$.

As noted above, we could alternatively simply combine our Theorem 1 with [36] to obtain the statement

$$\mathsf{NP} \not\subseteq \mathsf{BPP} \implies \exists\mathsf{OWF}.$$

However, we provide in [15] a self-contained proof of a BPP-black-box non-adaptive reduction. This is for completeness and to clarify the connection to Theorem 1.

Finally, we mention a few previous works related to our Theorem 5. Interestingly, by Allender et al., computing Kolmogorov complexity is known to be hard for PSPACE under deterministic adaptive Turing reductions [4]. This was improved by Hirahara to show that Kolmogorov complexity is hard for $\mathsf{EXP}^{\mathsf{NP}}$ under deterministic adaptive Turing reductions and hard for NEXP under randomized non-adaptive reductions [21]. Thus, Theorem 5 indicates a sharp contrast between the power of randomized many-one reductions and more powerful reductions with respect to the hardness of Kolmogorov complexity. Saks and Santhanam also prove that NP-hardness of *approximating* Kolmogorov complexity within $\omega(\log n)$ additive error under honest randomized non-adaptive reductions would imply $\mathsf{NP} \subseteq \mathsf{coAM}$ [39]. Note that Theorem 5 does not assume honesty.

4 Techniques

In this section, we give an overview of the techniques used to prove our main results. Formal details can be found in the full version of the paper [15].

4.1 Proof sketch of Theorem 1

As a warm-up, first consider the case of a deterministic length-increasing many-one reduction. In particular, let R be such a reduction from SAT to $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$ mapping inputs $\varphi \in \{0,1\}^n$ to outputs $(x, 1^s)$ with $|x| \geq n^{2/\delta}$ and with the superscript t greater than the running time of R. It is easy to see that, for any output $(x, 1^s)$ of $R(\varphi)$,

$$\begin{aligned}\mathsf{K}^t(x) &\leq |\varphi| + O(\log n)\\ &\leq |x|^\delta\\ &\leq s + |x|^\delta.\end{aligned}$$

This follows from the procedure that, given φ hard-coded, simulates $R(\varphi)$ and returns its output. Accordingly, a reduction of this kind cannot exist: since all of its outputs are Yes-instances, it would imply $\varphi \in \mathsf{SAT}$ for every formula φ.

When moving to the more general case of a randomized many-one reduction, one can think of $R(\varphi)$ as a distribution over instances of $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$, and a given output x is made with probability according to $R(\varphi)$. Observe that in the deterministic case, it held trivially that with high probability over $x \sim R(\varphi)$,

$$\mathsf{K}^t(x) \lesssim s \iff \Pr[R(\varphi) = x] > \beta,$$

for any choice of $\beta \in (0,1)$. We would like to show that something similar is true in the randomized setting. That is, there is still a correspondence between the K^t complexity of outputs and their probability under $R(\varphi)$. This means that $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$ (and thereby SAT) will reduce to a problem of probability estimation.

There exists unconditionally a coAM protocol A that, given (φ, x, β) as input, accepts iff $\Pr[R(\varphi) = x]$ is roughly greater than β, with high probability over $x \sim R(\varphi)$ [14, 11]; see also [25, Appendix A]. Under our derandomization assumption, A can be implemented in coNP. For simplicity, assume that every output $(x, 1^s)$ of R has the same threshold parameter $s \in \mathbb{N}$, so we may omit this part of the outputs. Define a parameter

$$\beta = \frac{1}{2^s \cdot \mathsf{poly}(n)}.$$

We claim that for every $\varphi \in \{0,1\}^n$, $A(\varphi, x, \beta)$ will work well at deciding $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$ on outputs x of $R(\varphi)$.

On one hand, we will show that with high probability over $x \sim R(\varphi)$, if $\mathsf{K}^t(x) \leq s$, then $\Pr[R(\varphi) = x] > \beta$. The idea is to use a counting argument, giving an upper bound on x such that $\mathsf{K}^t(x) \leq s$, to show that $R(\varphi)$ must be "concentrated" on these inputs. In particular, the probability over $x \sim R(\varphi)$ that $\mathsf{K}^t(x) \leq s$ and $\Pr[R(\varphi) = x] \leq \beta$ is roughly at most

$$2^s \cdot \beta = \frac{1}{\mathsf{poly}(n)}.$$

So, with high probability over $x \sim R(\varphi)$, if x is a Yes-instance of $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$, then $\Pr[R(\varphi) = x] > \beta$, in which case $A(\varphi, x, \beta)$ correctly outputs 1.

On the other hand, we will show that if an output x has probability greater than β under $R(\varphi)$, then x must have K^t complexity roughly upper-bounded by s. In the realm of time-unbounded Kolmogorov complexity, we could rely on the well-known Coding Theorem to prove a statement of this kind. Namely, for any samplable distribution D, it holds that

$$\mathsf{K}(x) \leq \log(1/D(x)) + O(\log n).$$

Similarly, if D is samplable given some non-uniform input φ, then

$$\mathsf{K}(x) \leq \log(1/D(x)) + |\varphi| + O(\log n).$$

Observe that our distribution $R(\varphi)$ is samplable in polynomial time given φ as input. Thus, if x is samplable with probability greater than β under $R(\varphi)$, then it holds that

$$\begin{aligned}\mathsf{K}(x) &< \log(1/\beta) + |\varphi| + O(\log n)\\ &\leq s + |\varphi| + O(\log n)\\ &\leq s + |x|^{\delta}.\end{aligned}$$

Of course, bounding K-complexity does not suffice for our purposes. Instead, we apply a recent work of Lu, Oliveira, and Zimand [34], which gives unconditionally a coding theorem for *probabilistic* K^t complexity, denoted pK^t. Specifically, we use a version of the coding theorem for distributions samplable in polynomial time given an auxiliary non-uniform input. For some polynomial p_{sc} and time-bound $t_0 = \mathsf{poly}(n)$ at least the running time of R, this yields

$$\mathsf{pK}^{p_{sc}(t_0)}(x) \leq s + |\varphi| + O(\log n).$$

Roughly speaking, pK^t-complexity refers to the time-bounded Kolmogorov complexity of a string in the presence of some uniform randomness. This notion is in some sense intermediate between K^t complexity and K complexity. Moreover, under the derandomization assumption $\mathsf{E} \not\subseteq \mathsf{io\text{-}NSIZE}[2^{o(n)}]$, pK^t and K^t turn out to be nearly equal: for some polynomial p_0, $\mathsf{K}^{p_0(t)}(x) \leq \mathsf{pK}^t(x) + \log p_0(t)$ [16]. So, for $t \geq p_0(p_{sc}(t_0))$, the above implies

$$\begin{aligned}\mathsf{K}^t(x) &\leq s + |\varphi| + O(\log n)\\ &\leq s + |x|^{\delta}.\end{aligned}$$

To summarize, with a sufficiently large $t = \mathsf{poly}(n)$ and a derandomization assumption, we obtain an auxiliary-input coding theorem for K^t complexity. This yields the required converse, namely, that high probability under $R(\varphi)$ implies bounded K^t.[4]

We conclude that the coNP procedure A can be used to decide SAT. Therefore, $\mathsf{NP} \subseteq \mathsf{coAM} = \mathsf{coNP}$.

To obtain Theorem 1 for *honest* reductions rather than polynomially length-increasing reductions, we can simply rely on the "paddability" of SAT. That is, given a SAT-instance $\varphi \in \{0,1\}^n$, it is trivial to append some terms to φ in a way that does not affect its satisfiability but increases its length as desired. Since our assumed reduction R is honest, for some constant $\gamma > 0$, for any query x of $R(\varphi)$, it holds that $|x| \geq |\varphi|^{\gamma}$. If we let R' be the reduction that, on input $\varphi \in \{0,1\}^n$, pads to obtain $\varphi' \in \{0,1\}^{n^{c/\gamma}}$ and then runs $R(\varphi')$ to obtain x, we will now have $|x| \geq |\varphi'|^{\gamma} = n^c$. To summarize, if there is an honest reduction from SAT to some language L, then there is also a polynomially length-increasing reduction from SAT to L.[5]

For the full statement of Theorem 1, we need techniques that can handle randomized non-adaptive Turing reductions. We exploit the fact from [31] that the non-existence of a one-way function would provide an algorithm A for probability estimation as described above. In particular, for any distribution $D \in \mathsf{PSAMP}$, for some poly-time computable

[4] We note that the use of the coding theorem for pK^t is the main reason why we need to require that the runtime of our randomized NP-hardness reductions for $\mathsf{Approx}_{n^s}\text{-}\mathsf{K}^t$ must be polynomially smaller than the parameter t.

[5] A similar application of padding is in [24].

function f, there is an oracle algorithm A such that $A^I(x)$ outputs an estimate of $\Pr[D = x]$ with high probability over $x \sim D$, where I is any inverter for f. Thus, in the presence of a non-adaptive reduction from SAT to $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{K}^t$, we also get a non-adaptive reduction from SAT to the inversion of a one-way function. It was shown in [2, 3], with the construction of a sophisticated protocol building on techniques from [13, 11], that such a reduction would imply $\mathsf{SAT} \in \mathsf{coAM}$. However, as mentioned above, our distributions of interest $R(\varphi)$ are not in PSAMP, but require φ as a non-uniform input. Luckily, a result of [9] transposes [2] to this non-uniform setting. Specifically, we have a reduction from SAT to the inversion of an *auxiliary-input* function $f = \{f_\varphi\}_{\varphi \in \{0,1\}^*}$, where on input φ to SAT, the reduction only needs to invert f on auxiliary input φ; given this, [9] yields $\mathsf{SAT} \in \mathsf{coAM}$. This completes our overview of the proof of Theorem 1.

4.2 Proof Sketch of Theorem 2

Our proof of Theorem 2 builds on that of Theorem 1, making use of a few more ideas to obtain a reduction from NP to inversion of a standard OWF. The first idea is the fact that any inverter for an appropriate function can be used as an *errorless average-case* inverter for a desired auxiliary-input function. In particular, let $f = \{f_\varphi\}_{\varphi \in \mathbb{N}}$ be an auxiliary-input function, and define g to be the function that randomly samples φ from a distribution D' and then applies f_φ to a uniformly random input z. It is not hard to show by an averaging argument that any inverter for g works as an inverter for f_φ with high probability over $\varphi \sim D'$. Moreover, crucially, if the inverter fails to invert some f_φ, then it can be made to output a special failure symbol $\perp$ when given the auxiliary input φ, with high probability. This is due to the fact that successful inversion can be verified in poly-time: given a candidate pre-image y of some string z under f_φ, simply run $f_\varphi(y)$ to verify; see [24, Theorem 10.3]. This, along with a reduction from SAT to inverting an auxiliary-input OWF, yields an errorless randomized heuristic for SAT over any distribution $D' \in \mathsf{PSAMP}$.

The final piece of Theorem 2 is a worst-case to average-case reduction. The goal is to obtain

$$(\mathsf{SAT}, D') \in \mathsf{AvgBPP} \implies \mathsf{SAT} \in \mathsf{BPP},$$

which will complete the proof given the discussion above. To that end, we employ tools from [19] and follow-up works. A difficulty is that, from $(\mathsf{SAT}, D') \in \mathsf{AvgBPP}$, the available worst-case to average-case reductions only yield

$$\mathsf{Gap}_{\tau, n^\delta}\mathsf{pK}^{\mathsf{t}} \in \mathsf{BPP}.$$

The promise-problem $\mathsf{Gap}_{\tau, n^\delta}\mathsf{pK}^{\mathsf{t}}$ is potentially easier than $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$, since it involves a polynomial gap τ between time-bounds in Yes-instances and No-instances. As a result, the gap version may not be NP-hard, so its easiness would not yield $\mathsf{SAT} \in \mathsf{BPP}$. Fortunately, by a different application of the coding theorem for pK^t, we are able to show that NP-hardness of $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$ implies NP-hardness of $\mathsf{Gap}_{\tau, n^\delta}\mathsf{pK}^{\mathsf{t}}$. Roughly, with high probability over the randomness of the reduction from SAT to $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$, the pK^t complexity of queried strings will be somewhat close to their time-unbounded K complexity. Thus, granted the leeway of the n^δ approximation term, the difference in time-bounds between t and $\tau(t)$ does not affect the correctness of the (slightly modified) reduction when we use $\mathsf{Gap}_{\tau, n^\delta}\mathsf{pK}^{\mathsf{t}}$ as an oracle in lieu of $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$.

To summarize, an outline of the proof is as follows.

1. Arguing as in Theorem 1, we get a black-box non-adaptive fixed-auxiliary input reduction from SAT to inverting an auxiliary-input function, $f = \{f_\varphi\}_{\varphi \in \{0,1\}^*}$.

2. Under our assumption of the non-existence of OWFs, we get, for any polynomial-time samplable distribution D, a PPT machine that inverts f_φ with high probability over $\varphi \sim D$. Combined with step (1), this yields that $(\mathsf{SAT}, D) \in \mathsf{AvgBPP}$.
3. From the worst-case to average-case reduction of [19] (and subsequent works [22] and [16]), for some distribution $D' \in \mathsf{PSAMP}$, there is a BPP-black-box non-adaptive randomized polynomial-time reduction from $\mathsf{Gap}_{\tau,O(\log n)}\mathsf{pK^t}$ to the average-case problem of solving SAT over D'. That is,

$$(\mathsf{SAT}, D') \in \mathsf{AvgBPP} \implies \mathsf{Gap}_{\tau,O(\log n)}\mathsf{pK^t} \in \mathsf{BPP}$$

   for a sufficiently large polynomial τ depending on the running time of the heuristic for SAT. Combined with step (2), we get that $\mathsf{Gap}_{\tau,O(\log n)}\mathsf{pK^t} \in \mathsf{BPP}$.
4. For a sufficiently large t, if $\mathsf{Approx}_{n^\delta}\text{-}\mathsf{pK}^t$ is NP-hard, then $\mathsf{Gap}_{\tau,O(\log n)}\mathsf{pK^t}$ is also NP-hard. Combined with step (3), this yields $\mathsf{NP} \subseteq \mathsf{BPP}$.

4.3 Proof Sketch of Theorem 4

For the proof of Theorem 4 in the setting of exact pK^t and K^t, the approach discussed above does not work; recall that the approximation term n^δ was critical at a number of points. Thus, our starting point is the following statement from a recent work of Liu and Pass [33].

> Assuming $\mathsf{E} \not\subseteq \mathsf{io\text{-}NSIZE}[2^{o(n)}]$, if $\{\mathsf{MK}^t\mathsf{P}\} \times \mathsf{SAMP}[t_D(n)] \not\subseteq \mathsf{HeurP}$ for some time bound t_D polynomially less than t, then one-way functions exist.

That is, the average-case hardness of $\mathsf{MK}^t\mathsf{P}$ with respect to *any* distribution samplable within some polynomial running time smaller than t would suffice to imply one-way functions.

Our goal now is to show that if $\mathsf{MK}^t\mathsf{P}$ is NP-hard, then $\{\mathsf{MK}^t\mathsf{P}\} \times \mathsf{SAMP}[t_D(n)]$ is "hard for distributional NP": namely, if $\mathsf{MK}^t\mathsf{P}$ is easy on average over every distribution D samplable in time t_D, then every distributional problem $(L, D') \in \mathsf{NP} \times \mathsf{PSAMP}$ is likewise easy on average. Combining this with the statement from [33], we would get

$$\begin{aligned}\mathsf{DistNP} \not\subseteq \mathsf{HeurP} &\implies \{\mathsf{MK}^t\mathsf{P}\} \times \mathsf{SAMP}[t_D(n)] \not\subseteq \mathsf{HeurP} \\ &\implies \exists\mathsf{OWF}.\end{aligned}$$

To show the distributional NP-hardness of $\mathsf{MK}^t\mathsf{P}$, we reduce from an arbitrary distributional problem $(L, D') \in \mathsf{DistNP}$. Under the assumed NP-hardness of $\mathsf{MK}^t\mathsf{P}$, there is a randomized non-adaptive reduction R from L to $\mathsf{MK}^t\mathsf{P}$. With a large enough choice of the polynomial t, we can ensure that the reduction from L to $\mathsf{MK}^t\mathsf{P}$ runs in time polynomially less than t. In particular, we get that the following distribution Q is samplable in time at most t_D:

> Sample $x \sim D'$, and then output a sample from the query distribution of $R(x)$.

From there, it is not too hard to show that, if H is a heuristic for $\mathsf{MK}^t\mathsf{P}$ working over Q, then the algorithm R^H (that simulates R and answers any oracle queries with H) is a heuristic for L over D'. This yields the desired result.

4.4 Proof Sketch of Theorem 5

Finally, the proof of Theorem 5 proceeds along the lines of that of Theorem 1, but with several important changes.[6] The main challenge is that the Coding Theorem for K only gives us an *approximate* equality between $\mathsf{K}(x)$ and $\log(1/D(x))$ for x's sampled from a distribution

[6] As mentioned above, we actually give two different proofs of Theorem 5. We describe the first one here.

D. This was not a problem for Theorem 1 as it dealt with an *approximate* version of K^t, and we could absorb some slack of the Coding Theorem into an approximation error of K^t. But Theorem 5 is for the *exact* version of K, and we cannot apply the same strategy here. Instead, we show that this slack can be absorbed by a different argument, crucially relying on the fact that the randomized reductions R in the assumption of Theorem 5 are *many-one* and have the error probability *inverse-polynomially small* in their runtime t_R.

Namely, for $\varphi \in \{0,1\}^n$, consider the distribution of queries $(x, 1^s)$ made by the reduction $R(\varphi)$. We call such a query "heavy" if its probability (according to $R(\varphi)$) is at least $1/(\mathsf{poly}(t_R(n)) \cdot 2^s)$.

Our SAT algorithm (using a probability estimation protocol as in Theorem 1) essentially behaves as follows:

> On input φ, sample a query $(x, 1^s)$ according to $R(\varphi)$, and accept if $(x, 1^s)$ is heavy.

For $\varphi \notin \mathsf{SAT}$ (which is the difficult case to analyze), heavy queries will cause our SAT algorithm to make a mistake by incorrectly accepting φ. We bound the error probability of our SAT algorithm by upperbounding the total probability mass of such heavy queries.

Roughly speaking, we upperbound the total probability mass of "heavy" queries $(x, 1^s)$ by

$$\mathsf{poly}(t_R(n)) \cdot \Pr[\mathsf{K}(x) \leq s].$$

Note that, since $\varphi \notin \mathsf{SAT}$, we have by the condition of correctness of the many-one reduction R that $R(\varphi)$ must place a very small γ probability on its queries that are Yes-instances of MKP, i.e., $\Pr[\mathsf{K}(x) \leq s] \leq \gamma$. Hence, the error probability of our SAT algorithm is at most $\mathsf{poly}(t_R(n)) \cdot \gamma$, which can be made sufficiently small if the error probability γ of the reduction R is inverse-polynomially small in the runtime $t_R(n)$.

5 NP-hardness of $(\mathsf{K}^t$ vs. $\mathsf{K})$ and $(\mathsf{K}^t$ vs. $\mathsf{K})^*$

In this section, we examine promise problems of the form $(\mathsf{K}^t$ vs. $\mathsf{K}^{t'})$, for time bounds $t, t' \in \mathbb{N}$, in comparison with the "partial function" versions $(\mathsf{K}^t$ vs. $\mathsf{K}^{t'})^*$ recently shown NP-complete by Hirahara [23]. While NP-hardness of $(\mathsf{K}^t$ vs. $\mathsf{K})$ would imply $\mathsf{NP} \subseteq \mathsf{coAM}$ via our proof techniques above, the consequence does not seem to follow in the partial setting, as we discuss further below. We then show that NP-hardness via *deterministic Turing* reductions of either $(\mathsf{K}^t$ vs. $\mathsf{K}^{t'})$ or $(\mathsf{K}^t$ vs. $\mathsf{K}^{t'})^*$ (with appropriate settings of t and t') would imply $\mathsf{NP} = \mathsf{P}$. It follows that these problems are NP-intermediate with respect to deterministic Turing reductions, provided the existence of one-way functions.

5.1 Randomized Reductions

We start with formal definitions of the partial version of K^t complexity and the promise problems mentioned above.

▶ **Definition 10** (Partial (Time-bounded) Kolmogorov Complexity). *For a time bound $t \in \mathbb{N}$, a string $x \in \{0,1,*\}^*$, and a complexity measure $\mu \in \{\mathsf{pK}^t, \mathsf{K}^t, \mathsf{K}\}$, the* partial ($t$-time-bounded, probabilistic) Kolmogorov complexity of x, *denoted $(\mu)^*(x)$, is equal to*

$$\min\{\mu(x') \mid x' \text{ consistent with } x\},$$

where a string $x' \in \{0,1\}^*$ is said to be consistent with $x \in \{0,1,*\}^*$ *if $|x'| = |x|$ and, for every index $i \in [|x|]$ such that $x[i] \neq *$, it holds that $x[i] = x'[i]$.*

▶ **Definition 11** ((K^t vs. $\mathsf{K}^{t'}$)). *Let $t, t' : \mathbb{N} \to \mathbb{N}$. For $\mu_1 \in \{\mathsf{K}^t, \mathsf{pK}^t\}$ and $\mu_2 \in \{\mathsf{K}^{t'}, \mathsf{pK}^{t'}, \mathsf{K}\}$, ($\mu_1$ vs. μ_2) is the following promise problem.*

- $\Pi_Y = \{(x, 1^s) \mid \mu_1(x) \leq s\}$
- $\Pi_N = \{(x, 1^s) \mid \mu_2(x) > s\}$

(μ_1 vs. μ_2)* is defined analogously, with the partial complexity measures $(\mu_1)^*$ and $(\mu_2)^*$ in place of the standard ("complete function") ones.

By a proof analogous to that of Theorem 5, we get the following statement.

▶ **Lemma 12.** *Let $t : \mathbb{N} \to \mathbb{N}$ be arbitrary and $t_R : \mathbb{N} \to \mathbb{N}$ a polynomial. If (K^t vs. K) is NP-hard under a randomized many-one reduction running in time $t_R(n)$ and with failure probability at most $1/(t_R(n))^7$, then $\mathsf{NP} \subseteq \mathsf{coAM}$.*

One may contrast Lemma 12 with Hirahara's recent proof that (K^t vs. K)* is in fact NP-hard under a randomized many-one reduction with the same properties. This suggests that the techniques of [23] will not extend to the setting of standard (K^t vs. K) without leveraging some more powerful notion of reducibility. Viewed another way, to obtain NP-hardness of $\mathsf{MK}^t\mathsf{P}$ complexity under randomized many-one reductions, one would need techniques that apply more narrowly to smaller-gap versions of the problem.

Note that the statement gives NP-hardness of $\mathsf{MK}^t\mathsf{P}^*$ under a randomized reduction even when $t \in \mathbb{N}$ is arbitrarily larger than the running time of the reduction. In the case of a randomized reduction, it is not unreasonable to make the assumption that $t \gg t_R$, as is done in [39] and in this work. This is because randomized reductions may easily sample strings of maximum Kolmogorov complexity, so it is easy to generate No-instances of $\mathsf{MK}^t\mathsf{P}$ (or $\mathsf{MK}^t\mathsf{P}^*$) within time t_R. Note that this would be impossible for a deterministic reduction.

▶ **Lemma 13** (Implicit in [23]). *There exists a polynomial $t_R : \mathbb{N} \to \mathbb{N}$ such that for any constant $c \in \mathbb{N}$ and any sufficiently large polynomial $t : \mathbb{N} \to \mathbb{N}$, ($\mathsf{K}^t$ vs. K)* is NP-hard under a randomized many-one reduction running in time $t_R(n)$ and with failure probability at most $1/t_R(n)^c$.*

Proof sketch. One needs to verify that the failure probability of the reduction is at most $1/n^c$ for an arbitrary large constant $c \in \mathbb{N}$. Recall that in the proof of [23] Lemma 8.3, the reduction samples random strings $f_i \sim \{0,1\}^{\lambda \cdot w(i)}$ for $i \in [n]$, where $n \in \mathbb{N}$ is the number of variables in the input CMMSA instance, $w : [n] \to \mathbb{N}$ is a weight function, and λ is some fixed polynomial in n. The reduction succeeds provided, for every $T \subseteq [n]$, for some constant $c \in \mathbb{N}$,

$$\mathsf{K}(f_T) \geq \lambda \cdot w(T) - c \cdot |T| \cdot \log n. \tag{1}$$

This is used in the "soundness" part of the proof to argue that the set $B \subseteq [n]$ is not authorized. In particular, one must prove that $w(B) < \theta$ from the fact that $\mathsf{K}(f_B) \leq o(\lambda \cdot w(B)) + |M|$, where $|M|$ is an arbitrary program of size $\lambda\theta/2$. To see that Eq. (1) is sufficient for this purpose, observe that for any $c \in \mathbb{N}$,

$$\begin{aligned} \lambda \cdot w(B) - c \cdot |B| \cdot \log n &\leq \mathsf{K}(f_B) \\ &\leq o(\lambda \cdot w(B)) + |M| \end{aligned}$$

implies that

$$\begin{aligned} \lambda \cdot w(B) &\leq c \cdot |B| \cdot \log n + o(\lambda \cdot w(B)) + |M| \\ &\leq o(\lambda \cdot w(B)) + |M|, \end{aligned}$$

since $c \cdot |B| \cdot \log n \leq cn \log n = o(\lambda)$. Thus,

$$\begin{aligned} w(B) \cdot \lambda \cdot (1 - o(1)) &\leq |M| \\ &\leq \lambda \cdot \theta/2, \end{aligned}$$

which implies that $w(B) < \theta$, as desired.

Now we will show that, for any $c \in \mathbb{N}$, Eq. (1) holds with probability at least $1 - 1/n^{c-2}$. First observe that by a standard counting argument, with probability $1 - 1/n^{c-2}$,

$$\mathsf{K}(f_{[n]}) \geq \lambda \cdot w([n]) - (c-2) \cdot \log n.$$

Moreover,

$$\mathsf{K}(f_{[n]}) \leq \mathsf{K}(f_T) + \lambda \cdot w([n] \backslash T) + 2 \cdot |T| \cdot \log n,$$

since one may describe $f_{[n]}$ by describing f_T, hard-wiring $f_{[n]\backslash T}$, and describing the set $T \subseteq [n]$ itself. Thus,

$$\begin{aligned} \mathsf{K}(f_T) &\geq \mathsf{K}(f_{[n]}) - \lambda \cdot w([n] \backslash T) - 2 \cdot |T| \cdot \log n \\ &\geq \lambda \cdot w([n]) - (c-2) \cdot \log n - \lambda \cdot w([n] \backslash T) - 2 \cdot |T| \cdot \log n \\ &\geq \lambda \cdot w(T) - c \cdot |T| \cdot \log n, \end{aligned}$$

so the reduction does not fail in this case. ◀

One may wonder why the barrier of Lemma 12 does not apply to the partial K^t setting. The primary issue is that a correspondence between the compressibility of queries and their probability under the query distribution Q_φ appears to be missing. As a result, we cannot apply our central proof technique of reducing meta-complexity to a problem of probability estimation.

Roughly speaking, there is a difference between the Kolmogorov complexity $\mathsf{K}(z)$ of the *description* of a query $z := (x, 1^s)$ with $x \in \{0, 1, *\}^*$ and the *partial complexity* $\mathsf{K}^*(x)$ of x. By the Coding Theorem for K, we still have an approximate correspondence between the logarithm of the inverse probability of (the description of the query) z output by the randomized reduction and the complexity $\mathsf{K}(z)$. However, $\mathsf{K}^*(x)$ can differ significantly from $\mathsf{K}(z)$. For example, consider a string $y = 0^n$, and let y' be a uniformly random string in $\{0, *\}^n$. Since y' is a uniformly random string over the binary alphabet $\{0, *\}$, it's almost certainly true that $\mathsf{K}(y') \geq n - O(\log n)$. On the other hand, $\mathsf{K}^*(y') \leq \mathsf{K}(y) \leq O(\log n)$.

More concretely, for example, consider a reduction from SAT to the problem of approximating $(\mathsf{K}^t)^*$ (with a fixed threshold parameter $s \in \mathbb{N}$). Here, the queries $x \in \{0, 1, *\}^*$ may contain unspecified "$*$" positions. On one hand, we can use a standard coding theorem (adapted appropriately) to show that a query x having probability greater than $\beta \approx 1/(2^s \cdot \mathsf{poly}(n))$ under the query distribution Q_φ would imply that $(\mathsf{K}^t)^*(x) \lesssim s$.

However, the converse does not seem to hold. Previously we showed that, for strings queried in the reduction, it was unlikely for a string to be both of low complexity and low probability. This followed from a counting argument and a union bound: there are roughly at most 2^s strings $x \in \{0, 1\}^*$ with $\mathsf{K}^t(x) \leq s$, so the cumulative probability of strings with both this property and $Q_\varphi(x) \leq \beta$ is at most $1/\mathsf{poly}(n)$. In the case of partial K^t, it is no longer true that there are "few" strings of low complexity. In particular, any one short description $d \in \{0, 1\}^s$ can witness $(\mathsf{K}^t)^*(x) \leq s$ for 2^n distinct strings $x \in \{0, 1, *\}^n$ (unlike standard K^t, where one description only "maps" to one string). Thus, partial K^t complexity is not readily connected to probability under efficiently samplable distributions, which was the key connection exploited in the previous sections.

5.2 Deterministic Reductions

As another point of comparison, in Lemmas 15 and 16, we show that if either of (K^t vs. $\mathsf{K}^{t'}$) or (K^t vs. $\mathsf{K}^{t'}$)* is NP-hard with respect to deterministic adaptive Turing reductions (for a sufficiently large exponential function t'), then one obtains the stronger consequence that $\mathsf{NP} = \mathsf{P}$. This implies that if one-way functions exist, (K^t vs. $\mathsf{K}^{t'}$) and (K^t vs. $\mathsf{K}^{t'}$)* are both NP-intermediate with respect to deterministic Turing reductions.[7]

Note that Lemmas 15 and 16 hold for Turing reductions with *arbitrary* polynomial running time (i.e., less than or greater than the time-bound t), and there is no honesty requirement. After this, we show similar results for honest reductions and superpolynomial t'.

We will use the "dream-breaker" of Bogdanov et al. [10].

▶ **Lemma 14** ([10]). *Suppose* $\mathsf{NP} \neq \mathsf{P}$. *There is an algorithm B and a universal constant d with the following properties. Let A be any poly-time algorithm that attempts to solve search-*SAT *and only errs by incorrectly outputting $\perp$.*[8] *For infinitely many $n \in \mathbb{N}$, $B(A, 1^n)$ outputs a formula $\varphi \in \{0,1\}^n$ and a witness a such that $\varphi(a) = 1$ but $A(\varphi) = \perp$. Moreover, if A runs in time at most n^b on inputs of length n, then $B(A, 1^n)$ runs in time at most $(n^b)^d$.*

▶ **Lemma 15.** *For every constant c, there is a constant c' with the following property. Let $t, t' : \mathbb{N} \to \mathbb{N}$ be such that for all $n \in \mathbb{N}$, $t(n) \leq n^c$ and $t'(n) \geq 2^{c'n}$. Then (K^t vs. $\mathsf{K}^{t'}$) is* NP*-hard under deterministic polynomial-time Turing reductions iff* $\mathsf{NP} = \mathsf{P}$.

Proof. Let M be a Turing reduction from search-SAT to (K^t vs. $\mathsf{K}^{t'}$) running in time at most n^b on inputs of length $n \in \mathbb{N}$. Define a machine M' that on input $\varphi \in \{0,1\}^n$ simulates $M(\varphi)$ and answers its queries as follows. If the query $(x, 1^s)$ is such that $s \leq 4b \log n$ and $s \leq 2|x|$, answer the query by brute force; otherwise simply accept the query. Note that M' runs in time at most n^{6bc}.

Let B be the refuter of Lemma 14, and let $n \in \mathbb{N}$ and $\varphi \in \{0,1\}^n$ be such that $B(M', 1^n) = (\varphi, a)$ with $M'(\varphi) = \perp$ but $\varphi(a) = 1$.

Clearly, if a query $(x, 1^s)$ is such that $s \leq 4b \log n$ or $2|x| < s$, M' answers it correctly. We now claim that for every query $(x, 1^s)$ of $M'(\varphi)$, it holds that $\mathsf{K}^{t'}(x) \leq 4b \log n$. In particular, one may compute x from advice (n, i), where x is the i^{th} query of $M'(\varphi)$, in time at most

$$\left(n^{6bc}\right)^d + n^{6bc} < 2^{c' \cdot |x|},$$

assuming $2|x| \geq s > 4b \log n$ and choosing $c' = 4cd$, where d is the constant from Lemma 14. For $t' : \mathbb{N} \to \mathbb{N}$ such that $t'(m) \geq 2^{c'm}$, this implies

$$\mathsf{K}^{t'}(x) \leq s.$$

Thus, $M'(\varphi)$ answers all of its queries correctly with respect to (K^t vs. $\mathsf{K}^{t'}$), and

$$M'(\varphi) = M^{(\mathsf{K}^t \text{ vs. } \mathsf{K}^{t'})}(\varphi) = \text{search-SAT}(\varphi),$$

a contradiction. ◀

The following statement for (K^t vs. $\mathsf{K}^{t'}$)* indicates that Lemma 13 makes essential use of randomness, unless $\mathsf{NP} = \mathsf{P}$.

[7] Since either of these problems could be used to break a cryptographic PRG, the existence of OWFs means they must not be efficiently decidable.

[8] Note that any poly-time algorithm may be transformed into such an algorithm by verifying any candidate satisfying assignment to the input before returning it.

▶ **Lemma 16.** *For every constant c, there is a constant c' with the following property. Let $t, t' : \mathbb{N} \to \mathbb{N}$ be such that for all $n \in \mathbb{N}$, $t(n) \leq n^c$ and $t'(n) \geq 2^{c'n}$. Then $(\mathsf{K}^t$ vs. $\mathsf{K}^{t'})^*$ is NP-hard under deterministic polynomial-time Turing reductions iff $\mathsf{NP} = \mathsf{P}$.*

Proof sketch. The proof is nearly identical to that of Lemma 15. One may still compute a string *consistent* with x from advice (n, i) by simulating the reduction, obtaining the query x, and replacing any $*$'s in x with 0's. Let $\tilde{x}$ be the string x with all *'s replaced by 0's. It is easy to verify that $(\mathsf{K}^{t'})^*(x) \leq \mathsf{K}^{t'}(\tilde{x}) \leq 4b \log n$. ◀

Note that we could prove the above lemmas for (K^t vs. K) and (K^t vs. K)* (that is, with time-unbounded K and K^* in Π_N) without the use of a dreambreaker. If we additionally assume that the NP-hardness reductions are honest, we obtain the same results but with t' any superpolynomial function.

▶ **Lemma 17.** *Let $t : \mathbb{N} \to \mathbb{N}$ be polynomial and $t' : \mathbb{N} \to \mathbb{N}$ superpolynomial. (K^t vs. $\mathsf{K}^{t'}$) is NP-hard under honest deterministic polynomial-time Turing reductions iff $\mathsf{NP} = \mathsf{P}$.*

Proof. Argue as in Lemma 15. Since the reduction is honest, we have

$$|x| \geq n^\gamma$$

for some constant $\gamma > 0$, for any string x queried in the reduction M. Recall that any such x of M may be computed from advice (n, i) in time at most

$$\begin{aligned}\left(n^{6bc}\right)^d + n^{6bc} &< n^{7bcd} \\ &\leq |x|^{7bcd/\gamma} \\ &< t'(|x|),\end{aligned}$$

as desired. ◀

▶ **Lemma 18.** *Let $t : \mathbb{N} \to \mathbb{N}$ be polynomial and $t' : \mathbb{N} \to \mathbb{N}$ superpolynomial. (K^t vs. $\mathsf{K}^{t'}$)* is NP-hard under honest deterministic polynomial-time Turing reductions iff $\mathsf{NP} = \mathsf{P}$.*

6 Open Questions

We have shown various consequences of (time-bounded) Kolmogorov complexity being NP-hard under randomized notions of reducibility. Some of these consequences may be taken optimistically (Theorem 4), while others may be viewed as barriers to the kinds of NP-hardness in question (Theorems 1, 5), which include kinds of reduction that have previously been used to show NP-hardness of variants of K^t complexity (e.g., [23]).

This work leaves open a number of directions; here, we indicate a few.

1. Can we remove the requirement, in Theorems 1, 2, and 4, that the time bound t in the superscript be larger than the running time of the reduction? Recall that this requirement was due to our use of the coding theorem for pK^t.
2. Can we show consequences of randomized NP-hardness reductions to MKTP or MCSP (i.e., minimization problems for Allender's KT complexity or boolean circuit size)?
3. Can we extend Theorems 1, 2, or 4 to *adaptive* randomized Turing reductions? Note that this kind of extension is unlikely in the case of Theorem 5, given the prior work discussed in Section 3 [4, 21].
4. Can we improve Theorem 5 to hold for randomized many-one reductions with constant failure probability? In particular, can we improve the "robustness" of many-one reductions to K, as in Theorem 8, to hold for constant failure probability and exponentially small failure probability?

References

1 William Aiello and Johan Håstad. Statistical zero-knowledge languages can be recognized in two rounds. *J. Comput. Syst. Sci.*, 42(3):327–345, 1991. doi:10.1016/0022-0000(91)90006-Q.

2 Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on np-hardness. In Jon M. Kleinberg, editor, *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, pages 701–710. ACM, 2006. doi:10.1145/1132516.1132614.

3 Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. Erratum for: on basing one-way functions on np-hardness. In Leonard J. Schulman, editor, *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 795–796. ACM, 2010. doi:10.1145/1806689.1806798.

4 Eric Allender, Harry Buhrman, Michal Koucký, Dieter van Melkebeek, and Detlef Ronneburger. Power from random strings. *SIAM J. Comput.*, 35(6):1467–1493, 2006. doi:10.1137/050628994.

5 Eric Allender and Bireswar Das. Zero knowledge and circuit minimization. *Inf. Comput.*, 256:2–8, 2017. doi:10.1016/J.IC.2017.04.004.

6 Eric Allender and Shuichi Hirahara. New insights on the (non-)hardness of circuit minimization and related problems. *ACM Trans. Comput. Theory*, 11(4):27:1–27:27, 2019. doi:10.1145/3349616.

7 Eric Allender, Shuichi Hirahara, and Harsha Tirumala. Kolmogorov complexity characterizes statistical zero knowledge. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*, volume 251 of *LIPIcs*, pages 3:1–3:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPICS.ITCS.2023.3.

8 Eric Allender, Dhiraj Holden, and Valentine Kabanets. The minimum oracle circuit size problem. *Comput. Complex.*, 26(2):469–496, 2017. doi:10.1007/S00037-016-0124-0.

9 Benny Applebaum, Boaz Barak, and David Xiao. On basing lower-bounds for learning on worst-case assumptions. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 211–220. IEEE Computer Society, 2008. doi:10.1109/FOCS.2008.35.

10 Andrej Bogdanov, Kunal Talwar, and Andrew Wan. Hard instances for satisfiability and quasi-one-way functions. In Andrew Chi-Chih Yao, editor, *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 5-7, 2010. Proceedings*, pages 290–300. Tsinghua University Press, 2010. URL: http://conference.iiis.tsinghua.edu.cn/ICS2010/content/papers/23.html.

11 Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for NP problems. *SIAM J. Comput.*, 36(4):1119–1159, 2006. doi:10.1137/S0097539705446974.

12 Ravi B. Boppana, Johan Håstad, and Stathis Zachos. Does co-np have short interactive proofs? *Inf. Process. Lett.*, 25(2):127–132, 1987. doi:10.1016/0020-0190(87)90232-8.

13 Joan Feigenbaum and Lance Fortnow. Random-self-reducibility of complete sets. *SIAM J. Comput.*, 22(5):994–1005, 1993. doi:10.1137/0222061.

14 Lance Fortnow. The complexity of perfect zero-knowledge. *Adv. Comput. Res.*, 5:327–343, 1989.

15 Halley Goldberg and Valentine Kabanets. Consequences of randomized reductions from SAT to time-bounded Kolmogorov complexity. *Electron. Colloquium Comput. Complex.*, TR24-120, 2024. URL: https://eccc.weizmann.ac.il/report/2024/120/.

16 Halley Goldberg, Valentine Kabanets, Zhenjian Lu, and Igor C. Oliveira. Probabilistic Kolmogorov complexity with applications to average-case complexity. In Shachar Lovett, editor, *37th Computational Complexity Conference, CCC 2022, July 20-23, 2022, Philadelphia, PA, USA*, volume 234 of *LIPIcs*, pages 16:1–16:60. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICS.CCC.2022.16.

17 Oded Goldreich, Amit Sahai, and Salil P. Vadhan. Can statistical zero knowledge be made non-interactive? or on the relationship of SZK and NISZK. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 467–484. Springer, 1999. doi:10.1007/3-540-48405-1_30.

18 Dan Gutfreund and Amnon Ta-Shma. Worst-case to average-case reductions revisited. In Moses Charikar, Klaus Jansen, Omer Reingold, and José D. P. Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 10th International Workshop, APPROX 2007, and 11th International Workshop, RANDOM 2007, Princeton, NJ, USA, August 20-22, 2007, Proceedings*, volume 4627 of *Lecture Notes in Computer Science*, pages 569–583. Springer, 2007. doi:10.1007/978-3-540-74208-1_41.

19 Shuichi Hirahara. Non-black-box worst-case to average-case reductions within NP. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 247–258. IEEE Computer Society, 2018. doi:10.1109/FOCS.2018.00032.

20 Shuichi Hirahara. Non-disjoint promise problems from meta-computational view of pseudorandom generator constructions. In Shubhangi Saraf, editor, *35th Computational Complexity Conference, CCC 2020, July 28-31, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 169 of *LIPIcs*, pages 20:1–20:47. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.CCC.2020.20.

21 Shuichi Hirahara. Unexpected hardness results for kolmogorov complexity under uniform reductions. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 1038–1051. ACM, 2020. doi:10.1145/3357713.3384251.

22 Shuichi Hirahara. Average-case hardness of NP from exponential worst-case hardness assumptions. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 292–302. ACM, 2021. doi:10.1145/3406325.3451065.

23 Shuichi Hirahara. Np-hardness of learning programs and partial MCSP. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 - November 3, 2022*, pages 968–979. IEEE, 2022. doi:10.1109/FOCS54457.2022.00095.

24 Shuichi Hirahara. Capturing one-way functions via np-hardness of meta-complexity. In Barna Saha and Rocco A. Servedio, editors, *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023*, pages 1027–1038. ACM, 2023. doi:10.1145/3564246.3585130.

25 Shuichi Hirahara and Osamu Watanabe. Limits of minimum circuit size problem as oracle. *Electron. Colloquium Comput. Complex.*, TR15-198, 2015. arXiv:TR15-198.

26 Shuichi Hirahara and Osamu Watanabe. Limits of minimum circuit size problem as oracle. In Ran Raz, editor, *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan*, volume 50 of *LIPIcs*, pages 18:1–18:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPICS.CCC.2016.18.

27 John M. Hitchcock and Aduri Pavan. On the np-completeness of the minimum circuit size problem. In Prahladh Harsha and G. Ramalingam, editors, *35th IARCS Annual Conference on Foundation of Software Technology and Theoretical Computer Science, FSTTCS 2015, December 16-18, 2015, Bangalore, India*, volume 45 of *LIPIcs*, pages 236–245. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2015. doi:10.4230/LIPICS.FSTTCS.2015.236.

28 Rahul Ilango. Approaching MCSP from above and below: Hardness for a conditional variant and $AC^0[p]$. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference, ITCS 2020, January 12-14, 2020, Seattle, Washington, USA*, volume 151 of *LIPIcs*, pages 34:1–34:26. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.ITCS.2020.34.

29 Rahul Ilango. SAT reduces to the minimum circuit size problem with a random oracle. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 733–742. IEEE, 2023. doi:10.1109/FOCS57990.2023.00048.

30 Rahul Ilango, Bruno Loff, and Igor C. Oliveira. Np-hardness of circuit minimization for multi-output functions. In Shubhangi Saraf, editor, *35th Computational Complexity Conference, CCC 2020, July 28-31, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 169 of *LIPIcs*, pages 22:1–22:36. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.CCC.2020.22.

31 Russell Impagliazzo and Leonid A. Levin. No better ways to generate hard NP instances than picking uniformly at random. In *31st Annual Symposium on Foundations of Computer Science, St. Louis, Missouri, USA, October 22-24, 1990, Volume II*, pages 812–821. IEEE Computer Society, 1990. doi:10.1109/FSCS.1990.89604.

32 Valentine Kabanets and Jin-yi Cai. Circuit minimization problem. In F. Frances Yao and Eugene M. Luks, editors, *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA*, pages 73–79. ACM, 2000. doi:10.1145/335305.335314.

33 Yanyi Liu and Rafael Pass. One-way functions and the hardness of (probabilistic) time-bounded kolmogorov complexity w.r.t. samplable distributions. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part II*, volume 14082 of *Lecture Notes in Computer Science*, pages 645–673. Springer, 2023. doi:10.1007/978-3-031-38545-2_21.

34 Zhenjian Lu, Igor C. Oliveira, and Marius Zimand. Optimal coding theorems in time-bounded kolmogorov complexity. In Mikolaj Bojanczyk, Emanuela Merelli, and David P. Woodruff, editors, *49th International Colloquium on Automata, Languages, and Programming, ICALP 2022, July 4-8, 2022, Paris, France*, volume 229 of *LIPIcs*, pages 92:1–92:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICS.ICALP.2022.92.

35 Cody D. Murray and Richard Ryan Williams. On the (non) np-hardness of computing circuit complexity. In David Zuckerman, editor, *30th Conference on Computational Complexity, CCC 2015, June 17-19, 2015, Portland, Oregon, USA*, volume 33 of *LIPIcs*, pages 365–380. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2015. doi:10.4230/LIPICS.CCC.2015.365.

36 Mikito Nanashima. On basing auxiliary-input cryptography on np-hardness via nonadaptive black-box reductions. In James R. Lee, editor, *12th Innovations in Theoretical Computer Science Conference, ITCS 2021, January 6-8, 2021, Virtual Conference*, volume 185 of *LIPIcs*, pages 29:1–29:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.ITCS.2021.29.

37 Rafail Ostrovsky and Avi Wigderson. One-way fuctions are essential for non-trivial zero-knowledge. In *Second Israel Symposium on Theory of Computing Systems, ISTCS 1993, Natanya, Israel, June 7-9, 1993, Proceedings*, pages 3–17. IEEE Computer Society, 1993. doi:10.1109/ISTCS.1993.253489.

38 Michael E. Saks and Rahul Santhanam. Circuit lower bounds from np-hardness of MCSP under turing reductions. In Shubhangi Saraf, editor, *35th Computational Complexity Conference, CCC 2020, July 28-31, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 169 of *LIPIcs*, pages 26:1–26:13. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.CCC.2020.26.

39 Michael E. Saks and Rahul Santhanam. On randomized reductions to the random strings. In Shachar Lovett, editor, *37th Computational Complexity Conference, CCC 2022, July 20-23, 2022, Philadelphia, PA, USA*, volume 234 of *LIPIcs*, pages 29:1–29:30. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICS.CCC.2022.29.

40 Boris A. Trakhtenbrot. A survey of russian approaches to perebor (brute-force searches) algorithms. *IEEE Ann. Hist. Comput.*, 6(4):384–400, 1984. doi:10.1109/MAHC.1984.10036.