A Relativization Perspective on Meta-Complexity

Hanlin Ren
University of Oxford, UK

Rahul Santhanam
University of Oxford, UK

Abstract

Meta-complexity studies the complexity of computational problems about complexity theory, such as the Minimum Circuit Size Problem (MCSP) and its variants. We show that a relativization barrier applies to many important open questions in meta-complexity. We give relativized worlds where:

1. MCSP can be solved in deterministic polynomial time, but the search version of MCSP cannot be solved in deterministic polynomial time, even approximately. In contrast, Carmosino, Impagliazzo, Kabanets, Kolokolova [CCC'16] gave a randomized approximate search-to-decision reduction for MCSP with a relativizing proof.
2. The complexities of MCSP$[2^{n/2}]$ and MCSP$[2^{n/4}]$ are different, in both worst-case and average-case settings. Thus the complexity of MCSP is not "robust" to the choice of the size function.
3. Levin's time-bounded Kolmogorov complexity $\mathrm{Kt}(x)$ can be approximated to a factor $(2+\epsilon)$ in polynomial time, for any $\epsilon > 0$.
4. Natural proofs do not exist, and neither do auxiliary-input one-way functions. In contrast, Santhanam [ITCS'20] gave a relativizing proof that the non-existence of natural proofs implies the existence of one-way functions under a conjecture about optimal hitting sets.
5. DistNP does not reduce to GapMINKT by a family of "robust" reductions. This presents a technical barrier for solving a question of Hirahara [FOCS'20].

2012 ACM Subject Classification Theory of computation → Oracles and decision trees; Theory of computation → Circuit complexity; Theory of computation → Problems, reductions and completeness

Keywords and phrases meta-complexity, relativization, minimum circuit size problem

Digital Object Identifier 10.4230/LIPIcs.STACS.2022.54

Related Version *Full Version*: `https://eccc.weizmann.ac.il/report/2021/089` [40]

1 Introduction

Meta-complexity refers to the complexity of computing complexity. A prominent example of a meta-complexity problem is the Minimum Circuit Size Problem (MCSP): Given as input the (length-2^n) truth table of a function $f : \{0,1\}^n \to \{0,1\}$, output the size of the smallest circuit that computes f. MCSP was recognized as a fundamental problem in the Soviet Union since 1950s [43], and has received a lot of attention in the last two decades since the seminal work of Kabanets and Cai [28]. Other examples include computing variants of Kolmogorov complexity such as polynomial-time bounded Kolmogorov complexity and Levin's time-bounded Kolmogorov complexity Kt [2, 29]. Questions about the circuit size of Boolean functions are closely related to Kolmogorov complexity and incompressibility, because a circuit is essentially a *compressed representation* of the truth table of the function it computes.

There has been plenty of interplay between meta-complexity and other areas of complexity theory such as average-case complexity [15, 16, 18, 19], cryptography [32, 38, 39, 42], learning theory [10, 36] and pseudorandomness [2, 17, 28, 36].

We highlight a couple of recent breakthrough results. The first gives a non-black-box worst-case to average-case reduction for a problem about Kolmogorov complexity ("GapMINKT") that many believe to be NP-hard.

39th International Symposium on Theoretical Aspects of Computer Science (STACS 2022).
Editors: Petra Berenbrink and Benjamin Monmege; Article No. 54; pp. 54:1–54:13
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

▶ **Theorem 1** ([15], building on [10]). *There is a randomized polynomial-time worst-case to average-case reduction for* GapMINKT*.*

The second gives an *equivalence* between the existence of one-way functions and the bounded-error average-case hardness over the uniform distribution of the functional version of MINKT. This result *characterizes* the most fundamental primitive in cryptography by a notion in meta-complexity.

▶ **Theorem 2** ([32]). *One-way functions exist if and only if there is a polynomial p such that the $p(n)$-time bounded Kolmogorov complexity of a string x of length n cannot be computed in polynomial time on average, when x is chosen uniformly at random from n-bit strings.*

Results such as these give hope for a rich theory connecting complexity lower bounds, meta-complexity, average-case complexity, learning theory and cryptography, among other fields. However, despite much effort, many basic questions about meta-complexity remain elusive. In addition, the recent advances on meta-complexity also propose new questions, some of which are seemingly beyond our reach. (See Section 1.1 for a sample of these questions.)

In this work, we seek a more fine-grained understanding of the current landscape of meta-complexity by using the classical perspective of *relativization* [9]. It is noteworthy that Theorem 1 and Theorem 2 relativize. Of course, we need to be careful here to define what relativization means, as the notion typically applies to complexity classes and not to computational problems. However, meta-computational problems do indeed have natural notions of relativizations, where the algorithms solving the problem as well as the algorithms defining the problem get access to the same oracle A. Results such as Theorem 1 and Theorem 2 use techniques from the theory of pseudorandomness [27, 34, 44], which typically relativize, and it is worth asking how much these techniques can achieve. Can they be used to solve the major open problems in the area?

We give a largely negative answer to this question, by giving oracles relative to which many of the questions in the area have answers opposite to what we expect. However, we do not necessarily infer that there are fundamental barriers to solving the major open questions; we can only say that new techniques will be required in many cases. Our perspective also contributes to formulating new notions and questions which might still be approachable using current techniques. We also note that there are some exciting recent works in meta-complexity by Ilango and others (e.g. [22–25]) using gate elimination and related ideas. It is not clear yet whether relativization is a barrier to these techniques.

1.1 Our Questions

We first introduce the questions with which we are concerned.

1.1.1 Easiness or Hardness of Meta-Complexity Problems

Arguably, the most important and fundamental problem about MCSP is whether MCSP is easy or hard. Is MCSP in polynomial time, or if not, is MCSP NP-complete? It is reported in [5, 30] that Levin delayed the publication of his NP-completeness results [31] because he wanted to show NP-hardness for MCSP. A long line of research [3, 4, 12, 20, 21, 28, 33, 41] showed that the NP-completeness of MCSP implies breakthrough results in complexity theory. For instance, if MCSP is NP-complete under polynomial-time Karp reductions, then $\mathsf{EXP} \neq \mathsf{ZPP}$ [33]. However, these results do not indicate whether MCSP is or is not NP-complete; they merely suggest that this problem will be hard to solve.

▶ **Question 3.** *Is* MCSP NP*-complete under polynomial-time Karp reductions?*

Just as with MCSP, it is open to show the NP-hardness of MINKT. A further motivation for this problem is the recent "non-black-box" worst-case to average-case reduction for MINKT [15]. As a consequence, if GapMINKT is NP-hard, then the worst-case and average-case complexities of NP are equivalent. As there are serious obstacles to showing the NP-completeness of MINKT by "weak" reductions, [15] proposed, as a weakening of Question 3, that MINKT could be NP-hard via very powerful reductions:

▶ **Question 4.** *Is* GapMINKT NP*-hard under* $\mathsf{coNP}_{/\mathrm{poly}}$*-Turing reductions?*

In terms of unconditional lower bounds, there is an intriguing question about the meta-complexity of Levin's Kt complexity, raised in [2]. It is known that MKtP is EXP-complete, but only under rather powerful reductions such as $\mathsf{P}_{/\mathrm{poly}}$-truth-table reductions or NP-Turing reductions. Therefore, it is reasonable to conjecture that MKtP is not in P. However, the aforementioned reducibilities are too strong, so we cannot apply the time hierarchy theorem directly to prove that MKtP $\notin$ P. Still, it may be surprising that this problem has been open for almost 20 years:[1]

▶ **Question 5.** *Is* MKtP *computable (or at least approximable) in polynomial-time?*

(We note that a randomized version of MKtP, called MrKtP, is known to be not in BPP unconditionally [35].)

1.1.2 Structural Properties of Meta-Complexity Problems

Every NP-complete problem admits a *search-to-decision* reduction. For instance, given an oracle that decides SAT, for every input formula φ that is satisfiable, we can find a satisfying assignment of φ in polynomial time. However, it is unknown whether MCSP has this property.

▶ **Question 6.** *Does* MCSP *admit a search-to-decision reduction?*

We remark that there has been some progress on Question 6: [10] showed that if MCSP is in BPP, then a certain "weak" version of search-MCSP can be solved in probabilistic polynomial time; [23] presented a "non-trivial" search-to-decision reduction for the problem of minimizing formulas.

Another mystery about MCSP is whether its various *parameterized* versions are equivalent. Specifically, let MCSP$[s(n)]$ denote the problem that given a truth table of a function $f : \{0,1\}^n \to \{0,1\}$, determine whether f can be computed by a circuit of size $s(n)$. It is easy to see that MCSP$[2^{n/2}]$ reduces to MCSP$[2^{n/4}]$,[2] but the converse direction is unknown:

▶ **Question 7.** *Is* MCSP$[2^{n/4}]$ *reducible to* MCSP$[2^{n/2}]$ *under polynomial-time Karp reductions?*

The average-case version of Question 7 is also open. It is observed in [19] that any errorless heuristic for MCSP$[2^{n/2}]$ can be transformed into an errorless heuristic for MCSP$[2^{n/4}]$, but the converse is unknown.

[1] The conference version of [2] was published in 2002.

[2] Given an input truth table f of length 2^n, let f' be the concatenation of 2^n copies of f, then $f' : \{0,1\}^{2n} \to \{0,1\}$ is a function that only depends on half of its input bits, and the circuit complexities of f and f' are exactly the same. Therefore $f \in$ MCSP$[2^{n/2}]$ if and only if $f' \in$ MCSP$[2^{n/4}]$.

▶ **Question 8.** *If* $\mathrm{MCSP}[2^{n/4}]$ *is easy on average, does this imply that* $\mathrm{MCSP}[2^{n/2}]$ *is also easy on average?*

One drawback of the worst-case to average-case reduction of [15] is that it only works for *zero-error* average-case complexity. Ideally, we would like to establish a worst-case to *two-sided-error* average-case reduction for MINKT. Can we extend the results in [15] to the two-sided-error setting?

▶ **Question 9.** *Is there a natural distribution such that, if* MINKT *is easy on this distribution with two-sided error, then* GapMINKT *is solvable in the worst case? In particular, does the* uniform *distribution satisfy the above condition?*

1.1.3 Meta-Complexity, Average-Case Complexity and Cryptography

Some of the most compelling questions around meta-complexity relate to connections with average-case complexity and cryptography. A partial converse of [15] was established in [16,17], where it was shown that if $\mathrm{GapMINKT}^{\mathrm{SAT}} \in \mathsf{P}$, then $\mathsf{DistNP} \subseteq \mathsf{AvgP}$, i.e. NP is easy on average. Here $\mathrm{GapMINKT}^{\mathrm{SAT}}$ is the problem of determining the (time-bounded) Kolmogorov complexity of a string with a SAT oracle. Based on this result, [16] characterized the average-case complexity of the polynomial hierarchy by the worst-case complexity of meta-complexity. An important open question, a positive answer to which would imply a characterization of the average-case complexity for NP, is whether the SAT oracle can be removed, that is:

▶ **Question 10.** *Does* $\mathrm{GapMINKT} \in \mathsf{P}$ *imply* $\mathsf{DistNP} \subseteq \mathsf{AvgP}$*?*

There seems to be strong correspondences between the hardness of MCSP and problems in cryptography. For example, if MCSP is easy, then one-way functions (OWFs) do not exist [28, 38]. Under the unproven Universality Conjecture, [42] established the converse direction, i.e. if MCSP is zero-error average-case hard, then OWFs exist. Of course, an *unconditional* answer would be much more interesting:

▶ **Question 11.** *Can we base the existence of OWF from the nonexistence of natural proofs?*

A recent exciting work [32] established the equivalence between the two-sided error average-case hardness of MINKT and the existence of one-way functions. Given the result in [32], it is perhaps natural to conjecture that $\mathrm{GapMINKT} \in \mathsf{CZK}$ unconditionally, where CZK is the set of languages with a computational zero-knowledge proof system [14]. One could imagine a win-win argument as follows: If MINKT is easy, then of course it is in CZK; on the other hand, if MINKT is hard, then one-way functions exist, and by the result of [14], every language in NP is in CZK. However, there are some gaps between the "easy" and "hard" in the above argument, as we do not know what happens if MINKT is only worst-case hard and one-way functions do not exist.

▶ **Question 12.** *Does (some gap version of)* MCSP *or* MINKT *admit a computational zero knowledge proof system?*

2 Our Results

In this work, we investigate the above questions in the perspective of *relativization*. Due to page limits, we only describe our results in this section and provide a proof overview in Section 3. The detailed proofs can be found in the full version of this paper [40].

2.1 Meta-Complexity Problems Are Not Robust in Relativized Worlds

In our first set of results, we present evidence for the following hypothesis: A *slight change* in the definition of a meta-complexity problem could result in a *completely different* problem. For example, we show that there are relativized worlds where MCSP is significantly easier than search-MCSP, and relativized worlds where MCSP$[2^{n/2}]$ and MCSP$[2^{n/4}]$ have dramatically different complexities.

▶ **Theorem 13** (Informal version). *For each of the following items, there is a relativized world where it becomes true.*

- MCSP $\in$ P*, but* search-MCSP *is very hard.*
- MCSP$[2^{n/2}] \in$ P*, but* MCSP$[2^{n/4}]$ *is very hard.*
- MCSP$[2^{n/4}]$ *admits a polynomial-time errorless heuristic, but* MCSP$[2^{n/2}]$ *does not.*

As direct consequences of Theorem 13, we have the following nonreducibility results: For example, unless nonrelativizing techniques are used, MCSP does not admit a search-to-decision reduction, and MCSP$[2^{n/4}]$ does not reduce to MCSP$[2^{n/2}]$.

2.2 Barriers for Proving Hardness of Kt Complexity

Our second result concerns Question 5.

▶ **Theorem 14** (Informal version). *There is a relativized world where Levin's* Kt *complexity can be* $(2+\epsilon)$*-approximated in polynomial time.*

We note that Question 5 also appeared in a stronger form in literature. In particular, let R_{Kt} be the set of strings x such that $\mathrm{Kt}(x) \geq |x|/3$, it is conjectured that any "dense enough" subset of R_{Kt} is not in polynomial time. Our result shows that this conjecture needs nonrelativizing techniques to prove.

Actually, our message is even stronger than the above statement of Theorem 14. We define a nonstandard variant of Levin's Kt complexity, and denote it as $\widetilde{\mathrm{Kt}}$, such that $\widetilde{\mathrm{Kt}}$ approximates Kt, i.e. for every string x, $\widetilde{\mathrm{Kt}}(x) \leq \mathrm{Kt}(x) \leq (2+o(1))\widetilde{\mathrm{Kt}}(x)$. Then we construct a relativized world where $\widetilde{\mathrm{Kt}}$ is computable in polynomial time *exactly*, and Theorem 14 follows directly.

However, non-relativizing techniques already play an important role in characterizing the complexity of R_{Kt}. It was shown that any dense subset of R_{Kt} is EXP-complete under $\mathsf{P}_{/\mathrm{poly}}$-truth-table reductions and NP-Turing reductions [2], and these results use the non-relativizing "instance checkers" for EXP-complete problems [7,8]. An *algebrization* barrier would be more satisfying for showing limitations of such techniques. However, we could not extend our oracle world to an algebrizing one in the sense of either [1], [26], or [6].

Nevertheless, we managed to construct an oracle world where $\widetilde{\mathrm{Kt}}$ is computable in polynomial time, and EXP = ZPP holds simultaneously.

▶ **Theorem 15.** *There is a relativized world where* $\widetilde{\mathrm{Kt}}$ *complexity is computable in deterministic polynomial time, and* EXP = ZPP*.*

In this world, EXP-complete problems have trivial instance checkers, since they are in ZPP. We also get some other non-relativizing theorems such as IP = PSPACE for free, since PSPACE $\subseteq$ EXP = ZPP $\subseteq$ IP. As a result, we cannot prove that $\widetilde{\mathrm{Kt}}$ is not in polynomial time, even if we combine IP = PSPACE or the instance checkers for EXP-complete problems with relativizing techniques. We believe that this oracle world serves as a "fundamental obstacle" ([2]) to proving MKtP $\notin$ P.

We think our new complexity measure $\widetilde{\mathrm{Kt}}$ is of independent interest. Understanding $\widetilde{\mathrm{Kt}}$ using nonrelativizing techniques may serve as the first step towards solving Question 5.

2.3 Natural Proofs Versus Cryptography

Our third set of results is motivated by Question 11. Under the so-called "Universality Conjecture", [42] answered Question 11 affirmatively, i.e. the non-existence of natural proofs is equivalent to the existence of one-way functions. In contrast, we show that the answer of Question 11 is false in some relativized world, establishing a barrier for constructing one-way functions from nonexistence of natural proofs. We can even rule out *auxiliary-input* one-way functions (a primitive weaker than one-way functions) in our world.

Consequently, the Universality Conjecture fails in this world. As we will discuss in Section 3.3, in this world, the Universality Conjecture actually fails in a *very intuitive way.*

▶ **Theorem 16** (informal version). *There is a relativized world where* $\mathsf{P}_{/\mathrm{poly}}$*-natural properties useful against* $\mathsf{SIZE}[2^{\delta n}]$ *do not exist, and auxiliary-input one-way functions do not exist either.*

The non-existence of natural proofs corresponds to the zero-error average-case hardness of MCSP [19]. We also extend our results by showing a relativized world where MCSP or MINKT is hard even for two-sided error heuristics.

▶ **Theorem 17** (informal version). *There is a relativized world where* GapMCSP *is hard on average under some samplable distribution, and auxiliary-input one-way functions do not exist.*

▶ **Theorem 18** (informal version). *There is a relativized world where* GapMINKT *is hard on average under some samplable distribution, and auxiliary-input one-way functions do not exist.*

Besides Question 11, we also show the following consequences based on our relativized worlds:

- (Question 9) Extending the results in [15] to the bounded-error case requires nonrelativizing techniques, if the underlying distribution for MINKT is still the uniform distribution. (This is because [32] showed the equivalence between the existence of one-way functions and the bounded-error average-case hardness of MINKT under the uniform distribution.)
- (Question 12) It requires nonrelativizing techniques to show that GapMINKT $\in \mathsf{CZK}$, or even that GapMINKT can be solved on average by a CZK protocol, on infinitely many input lengths. This is because [37] showed that if auxiliary-input one-way functions do not exist, then $\mathsf{CZK} = \mathsf{BPP}$.

  Note that the proof that if one-way functions exist then $\mathsf{NP} \subseteq \mathsf{CZK}$ [14] is already nonrelativizing. On the other hand, we show that basing GapMINKT $\in \mathsf{CZK}$ on the *non*existence of one-way functions also requires a nonrelativizing proof.

2.4 Limits of GapMINKT as an Oracle

We also present technical barriers for showing *stronger* reductions to the GapMINKT oracle, such as coNP-Turing reductions or $\mathsf{P}_{/\mathrm{poly}}$-Turing reductions.

We view (Turing) reductions to a promise problem $L = (L.\textsc{Yes}, L.\textsc{No})$ as machines that interact with an (adversarial) oracle, and tries to solve a problem L'. We say a reduction is *robust*, if it works even if the adversary is *inconsistent* on queries not in the promise. That is, on queries outside ($L.\textsc{Yes} \cup L.\textsc{No}$), the adversary can sometimes return 0 and sometimes return 1. Furthermore, the adversary is allowed to see the input of L' or the nondeterministic branch the reduction is running on, and decide whether to return 0 or 1 accordingly.

We show that a reduction that is both robust and relativizing cannot solve Question 10 or (a harder version of) Question 4. However, as the requirement of robust reductions seem very strong, we mainly treat these results as *technical* barriers rather than *conceptual* barriers. It is also worth mentioning that we use the "Gap" in GapMINKT in a very crucial way.

▶ **Theorem 19** (informal version)**.** *Each of the following items* cannot *be proved by a reduction that is both robust and relativizing.*

- *Either* GapMINKT $\in$ coNP*, or* GapMINKT *is* NP*-complete under* coNP*-Turing reductions.*
- *Every problem in* DistNP *has a polynomial-size two-sided error heuristic with* GapMINKT *oracles.*

We did not manage to prove non-hardness results under $\mathsf{coNP}_{/\mathrm{poly}}$-Turing reductions, as mentioned in Question 4. We leave it as an open problem.

▶ **Open Problem 20.** *Is there a relativized world where* GapMINKT $\notin \mathsf{coNP}_{/\mathrm{poly}}$*, and* GapMINKT *is not* NP*-complete under robust* $\mathsf{coNP}_{/\mathrm{poly}}$*-Turing reductions?*

3 Technical Overview

3.1 Meta-Complexity Problems Are Not Robust in Relativized Worlds

We briefly discuss the proof techniques of the first bullet of Theorem 13 here, i.e. there is an oracle world such that MCSP is easy but search-MCSP is hard. The framework for the other two bullets will be similar.

Making MCSP easy. We can add an MCSP oracle in our oracle world, but the circuit minimization problem in our world becomes $\mathrm{MCSP}^{\mathrm{MCSP}}$. Then we also need to add an $\mathrm{MCSP}^{\mathrm{MCSP}}$ oracle, but again, the circuit minimization problem becomes $\mathrm{MCSP}^{\mathrm{MCSP}^{\mathrm{MCSP}}}$ now. Therefore, a natural approach is to add the "limit" of

$$\mathrm{MCSP}^{\mathrm{MCSP}^{\mathrm{MCSP}^{\cdots}}}$$

into our oracle world. Indeed, this is what we do: We add an oracle itrMCSP (which stands for "iterated MCSP") into our world, such that (roughly speaking)

$$\mathrm{itrMCSP}[k,x,s] = \underbrace{\mathrm{MCSP}^{\mathrm{MCSP}^{\mathrm{MCSP}^{\cdots}}}}_{\text{iterate } k \text{ times}}[x,s].$$

(Recall that $\mathrm{MCSP}^{\mathcal{O}}[x,s] = 1$ if and only if in the oracle world with oracle $\mathcal{O}$, the circuit complexity of the truth table x is at most s.)

In our world, MCSP is indeed easy. Actually, let x be a truth table of length 2^n, then the circuit complexity of x is at most s in our world if and only if $\mathrm{itrMCSP}[2^n,x,s] = 1$.

Making search-MCSP hard. We define an oracle $\mathcal{O}$ that diagonalizes against every polynomial time Turing machine M, and define itrMCSP relative to $\mathcal{O}$. (That is, for example, $\mathrm{itrMCSP}[1,x,s] = \mathrm{MCSP}^{\mathcal{O}}[x,s]$ and $\mathrm{itrMCSP}[2,x,s] = \mathrm{MCSP}^{\mathrm{MCSP}^{\mathcal{O}}}[x,s]$.) For every Turing machine M, we find a large enough integer N and a hard truth table x_{hard}

of length poly(N). Then we feed x_{hard} to M. How we answer the $\mathcal{O}$ queries of M is not important, but each time M makes a query itrMCSP$[k, x, s]$, we *pretend x has the lowest possible circuit complexity*, and answer this query accordingly.

To be more precise, we fix the oracle $\mathcal{O}$ up to input length $N - 1$ before we simulate M on input x_{hard}. This has the effect that for every integer k, truth table x, and parameter $s \leq N - 1$, we already know whether itrMCSP$[k, x, s] = 1$ regardless of how we fix the rest of $\mathcal{O}$; see Claim 3.3 of the full version. Then upon every query itrMCSP$[k, x, s]$, if $s \leq N - 1$ we already know how to reply to it; otherwise we simply reply 1.

At last, for every query itrMCSP$[k, x, s]$ where $s \geq N$ and we returned 1, we need to put the truth table x in the length-N slice of $\mathcal{O}$ so that its circuit complexity is indeed at most N. Since M only runs in polynomial time, and only probes very few positions of $\mathcal{O}$, we can indeed put it somewhere in $\mathcal{O}$ without letting M notice. We do not need to care about the parameter k here, as MCSP$[x, N] = 1$ implies itrMCSP$[k, x, N] = 1$ for every k.[3] To diagonalize against M, we also put x_{hard} into the length-N slice of $\mathcal{O}$, but in a place that M did not probe at all. In this way, we can guarantee that there is a size-N circuit for x_{hard}, but M fails to find it.

3.2 Barriers for Proving Hardness of Kt Complexity

We first define the complexity $\widetilde{\mathrm{Kt}}$. For a string x, let $\widetilde{\mathrm{Kt}}(x)$ denote the minimum possible value of $|M| + \lfloor \log t \rfloor$, where after we run the machine M on the empty input for t steps, the content of some tape of M is exactly x. The difference between Kt and $\widetilde{\mathrm{Kt}}$ is that in the definition of Kt, we require M to halt after outputting x; while in the definition of $\widetilde{\mathrm{Kt}}$, x can be an intermediate step of the computation.

A fixed-point oracle. Our approach will be to find a "fixed-point" of $\widetilde{\mathrm{Kt}}$: an oracle $\mathcal{O}$ such that $\mathcal{O}[x] = \widetilde{\mathrm{Kt}}^{\mathcal{O}}(x)$ for every string x. Then, in the world with oracle $\mathcal{O}$, we can compute $\widetilde{\mathrm{Kt}}(x)$ by simply calling $\mathcal{O}[x]$.

We proceed in stages, and in stage n, we fix the strings that have $\widetilde{\mathrm{Kt}}$ complexity exactly n. We enumerate every (M, t) such that $|M| + \lfloor \log t \rfloor = n$, and run M for t steps. For every intermediate tape content x, if $\mathcal{O}[x]$ is not fixed yet, then we fix $\mathcal{O}[x] = n$. A natural problem is: how to respond to the $\mathcal{O}$ queries made by M? The answer is surprisingly simple: for every query $\mathcal{O}[y]$ that M makes, we already have $\widetilde{\mathrm{Kt}}(y) \leq n$ by definition, so if $\mathcal{O}[y]$ is not fixed to a value smaller than n yet, then we can return $\mathcal{O}[y] = n$ confidently! It is not hard to show that the oracle $\mathcal{O}$ is indeed a "fixed-point" of $\widetilde{\mathrm{Kt}}$.

Achieving EXP = ZPP. It is also simple to achieve $\mathsf{EXP} = \mathsf{ZPP}$ in the above oracle. To simulate exponential time, we give the zero-error probabilistic polynomial-time machine a "cheat" oracle Cheat that embeds the truth tables of a certain EXP-complete problem. It is natural to choose the EXP-complete problem as

$$L = \{(M, t) : M \text{ on empty input outputs 1 in time } t\},$$

since we can construct $\mathcal{O}$ and obtain the truth tables of L at the same time. We can reply arbitrarily when M queries the Cheat oracle.

Now we have a "fixed-point" oracle $\mathcal{O}$ such that $\mathcal{O}[x] = \widetilde{\mathrm{Kt}}^{\mathcal{O},\mathsf{Cheat}}(x)$ for every x. We also have a length-2^n truth table (of L), which we want to "embed" into Cheat. We can simply embed it into the length-$3n$ (say) slice of Cheat, as there are still many empty slots

[3] It is possible to define itrMCSP such that this is satisfied.

not asked in the construction of $\mathcal{O}$. Actually, the number of empty slots is so large (around $2^{3n} - 2^n\text{poly}(n)$) that we can embed it "everywhere we can". A ZPP algorithm can simply guess a pointer in the length-$3n$ slice of Cheat, and it will likely point to the truth table of L.

3.3 Natural Proofs Versus Cryptography

We only discuss how we prove Theorem 16. Our starting point is an oracle world in [45, Section 5], in which there is a hard-on-average problem but no auxiliary-input one-way functions. Given a function $f : \{0,1\}^n \to \{0,1\}^n$ (think of f as a uniformly random function), the world consists of two oracles: A PSPACE-complete oracle, and a "verification" oracle for f:

$$V_f[x, y] = \begin{cases} 1 & \text{if } f(x) = y, \\ 0 & \text{otherwise.} \end{cases}$$

Inverting auxiliary-input one-way functions. We use essentially the same argument as in [45]. Roughly speaking, given any circuit C of size s, it is possible to "eliminate" every V_f gate in C, and obtain a circuit C' of size poly(s), such that C and C' agree on a $1 - 1/s$ fraction of inputs, but C' does not use V_f at all. This is because V_f behaves like an oracle that is both random and sparse. Therefore, for each V_f gate, we only need to store its answers to the inputs that appear frequently, and V_f is likely zero on other inputs.

Now, given any circuit C, we want to "invert" C, i.e. given $C(\mathbf{z})$ for a uniformly random input $\mathbf{z}$, output any string in $C^{-1}(C(\mathbf{z}))$. We simply find a circuit C' that is close to C, uses no V_f gates, and is only polynomially larger than C. Then we use the PSPACE-complete oracle to invert C'.

Ruling out natural proofs. It suffices to show there is a *succinct pseudorandom distribution*, i.e. a distribution $\mathcal{D}$ over truth tables with small circuits, such that $\mathcal{D}$ is indistinguishable from the uniform distribution by small circuits. (Actually, this approach is inspired by recent circuit lower bounds [11, 19] for MCSP.)

Let $\mathcal{D}$ be any distribution over poly(s) strings, that fools PSPACE-oracle circuits of size s. The existence of $\mathcal{D}$ can be proven by the probabilistic method. For each $x \in \{0,1\}^{O(\log s)}$, let D_x be the x-th truth table in $\mathcal{D}$. We "embed" D_x into the oracle $V_f[x, f(x)]$, as follows:

$$V_f[x, y, \beta] = \begin{cases} D_x[\beta] & \text{if } f(x) = y, \\ \perp & \text{otherwise.} \end{cases}$$

Here, $D_x[\beta]$ is the β-th bit of D_x. Now we have artificially made $\mathcal{D}$ a *succinct* distribution: the circuit complexity of every string in $\mathcal{D}$ is small. We also need to prove $\mathcal{D}$ is *pseudorandom*, i.e. it fools every size $s^{o(1)}$ circuit. For every circuit C with V_f gates and PSPACE gates, we use the same method as above to eliminate every V_f gate in C, to obtain a circuit C' that is close to C. Note that the distribution under which we measure the closeness of C and C' is a hybrid of $\mathcal{D}$ and the uniform distribution. After that, we can use the fact that $\mathcal{D}$ fools C' to also show that $\mathcal{D}$ fools C, therefore C cannot be a natural proof.

How did the Universality Conjecture fail? The Universality Conjecture of [42] roughly says that if there are succinct pseudorandom distributions, then there are *efficiently samplable* succinct pseudorandom distributions. However, in our oracle world, the succinct pseudorandom distribution $\mathcal{D}$ does not appear to be efficiently samplable: to sample from $\mathcal{D}$, it seems that we need be able to compute f, which is hard when f is a random function.

3.4 Limits of GapMINKT as an Oracle

At the core of our proofs is the following weakness of GapMINKT: *It may hide a small change of the oracle.* In particular, suppose we have two oracles $\mathcal{O}$ and $\mathcal{O}'$, such that they only differ at one input, then the "Gap" in GapMINKT allows us to choose an instantiation of GapMINKT that is both consistent with $\text{GapMINKT}^{\mathcal{O}}$ and $\text{GapMINKT}^{\mathcal{O}'}$. (See Lemma 6.2 in the full version.) This instantiation of GapMINKT would not help the reduction distinguish between $\mathcal{O}$ and $\mathcal{O}'$ at all; however, an NP problem on $\mathcal{O}$ and $\mathcal{O}'$ may have very different answers.

NP-intermediateness under coNP-Turing reductions. It is not hard to construct a relativized world where GapMINKT $\notin$ coNP (see, e.g. [29, Theorem 4.1]). For the "non-completeness" part, we construct a diagonalizing oracle $\mathcal{O}$ such that there is no robust reduction from the NP problem

$$L = \{0^n : \mathcal{O} \cap \{0,1\}^n \neq \varnothing\}$$

to GapMINKT. On input length N, we construct a GapMINKT oracle that is both consistent with "$\mathcal{O} \cap \{0,1\}^N = \varnothing$" and "$|\mathcal{O} \cap \{0,1\}^N| = 1$". This oracle does not reveal whether $0^N \in L$, and we can still use the standard method to diagonalize against every co-nondeterministic Turing machine. In particular, we run this machine and reply 0 to all its queries to $\mathcal{O}$. If it rejects some branch, we put a string of length N that is not probed in this branch into $\mathcal{O}$; otherwise we do nothing.

Non-DistNP-hardness under $\mathsf{P}_{/\mathsf{poly}}$-Turing reductions. [13] showed that a random permutation $\pi : \{0,1\}^n \to \{0,1\}^n$ cannot be computed on average by circuits of size $2^{o(n)}$, even with a verification oracle

$$\Pi[\alpha, \beta] = \begin{cases} 1 & \text{if } \pi(\alpha) = \beta, \\ 0 & \text{otherwise.} \end{cases}$$

We show the same thing for (robust) circuits with Π and GapMINKT oracle gates. To oversimplify, the argument boils down to the following task: Given an input α, a circuit C that computes π correctly on α, and every value $\{\pi(\beta)\}_{\beta \neq \alpha}$, recover $\pi(\alpha)$. Without GapMINKT gates, it suffices to use $\log |C|$ bits to store a number k, such that on input α, the k-th Π gate of C contains the correct answer $\pi(\alpha)$. (For comparison, the trivial solution needs to record $n \gg \log |C|$ bits.)

Now, the circuit C has GapMINKT gates, and it is robust in the sense that $C^{\Pi,B}(\alpha) = \pi(\alpha)$ for *every* oracle B consistent with GapMINKT. Now we let B' be the MINKT oracle in the world where $\Pi[\alpha, \pi(\alpha)] = 0$, and other entries of Π are not changed. As the new oracle Π does not depend on $\pi(\alpha)$ at all, we can simulate $C^{\Pi,B'}(\alpha)$ without knowing $\pi(\alpha)$. On the other hand, we only modified one entry in Π, therefore B' is still consistent with GapMINKT. We still record the number k defined above for the simulation $C^{\Pi,B'}(\alpha)$, which suffices to recover $\pi(\alpha)$.

4 Related Works

In the paper that defined MINKT, Ko [29] studied the properties of MINKT in relativized worlds. Among other results, [29] showed that there is a relativized world where MINKT is neither in coNP, nor NP-complete under polynomial-time Turing reductions. This result indicates that the MINKT counterpart of Question 3 cannot be shown affirmatively using relativizing techniques. Also, [29] constructed a relativized world where $\mathsf{NP} \neq \mathsf{coNP}$, but MINKT *is* NP-complete under coNP-Turing reductions ("$\leq_T^{\mathsf{SNP}}$-reductions"). This leads to the conjecture [15, 29] that MINKT might be NP-complete under coNP-Turing reductions in the unrelativized world (Question 4).

Our third set of results build upon the results of Wee [45]. The motivation of [45] was to show that a certain cryptographic object (succinct noninteractive argument, SNARG) does not imply one-way functions in a relativizing way. The framework of [45] was very helpful for us, as we also need to rule out (auxiliary-input) one-way functions.

Xiao [46] presented a relativized world where learning is hard against circuits and auxiliary-input one-way functions do not exist either. It may seem that our results are direct corollaries of this result, since [10] proved that natural proofs imply learning algorithms. However, [46] only ruled out learning algorithms that use *uniform samples*, while the learning algorithms in [10] need *membership queries*. It seems that our results and [46] are incomparable. However, we remark that the techniques underlying [45, 46] and our results are quite similar.

We also mention the negative results of Hirahara and Watanabe [20] that has a different but similar setting compared to ours. In particular, they consider reductions to MCSP (in the unrelativized world) that are *oracle-independent*, i.e. work for MCSP^A for every oracle A. Two particular results in [20] are that deterministic oracle-independent reductions cannot reduce problems outside P to MCSP, and that randomized oracle-independent reductions that only make one query cannot reduce problems outside $\mathsf{AM} \cap \mathsf{coAM}$ to MCSP. As discussed in [20], the difference between relativization and their model is that in the relativized world with A oracle, a Turing reduction has access to not only MCSP^A but also A itself; however in their model, the reduction does not have access to A.

References

1 Scott Aaronson and Avi Wigderson. Algebrization: A new barrier in complexity theory. *ACM Transactions on Computation Theory*, 1(1):2:1–2:54, 2009. doi:10.1145/1490270.1490272.

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

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

4 Eric Allender, Rahul Ilango, and Neekon Vafa. The non-hardness of approximating circuit size. In *Proc. 14th International Computer Science Symposium in Russia (CSR)* , volume 11532 of *Lecture Notes in Computer Science*, pages 13–24, 2019. doi:10.1007/978-3-030-19955-5_2.

5 Eric Allender, Michal Koucký, Detlef Ronneburger, and Sambuddha Roy. The pervasive reach of resource-bounded Kolmogorov complexity in computational complexity theory. *Journal of Computer and System Sciences*, 77(1):14–40, 2011. doi:10.1016/j.jcss.2010.06.004.

6 Barış Aydınlıoğlu and Eric Bach. Affine relativization: Unifying the algebrization and relativization barriers. *ACM Transactions on Computation Theory*, 10(1):1:1–1:67, 2018. doi:10.1145/3170704.

7 László Babai, Lance Fortnow, and Carsten Lund. Non-deterministic exponential time has two-prover interactive protocols. *Computational Complexity*, 1:3–40, 1991. doi:10.1007/BF01200056.

8 László Babai, Lance Fortnow, Noam Nisan, and Avi Wigderson. BPP has subexponential time simulations unless EXPTIME has publishable proofs. *Computatioanl Complexity*, 3:307–318, 1993. doi:10.1007/BF01275486.

9 Theodore P. Baker, John Gill, and Robert Solovay. Relativizations of the P =?NP question. *SIAM Journal of Computing*, 4(4):431–442, 1975. doi:10.1137/0204037.

10 Marco L. Carmosino, Russell Impagliazzo, Valentine Kabanets, and Antonina Kolokolova. Learning algorithms from natural proofs. In *Proc. 31st Computational Complexity Conference (CCC)*, volume 50 of *LIPIcs*, pages 10:1–10:24, 2016. doi:10.4230/LIPIcs.CCC.2016.10.

11 Mahdi Cheraghchi, Valentine Kabanets, Zhenjian Lu, and Dimitrios Myrisiotis. Circuit lower bounds for MCSP from local pseudorandom generators. *ACM Transactions on Computation Theory*, 12(3):21:1–21:27, 2020. doi:10.1145/3404860.

12 Bin Fu. Hardness of sparse sets and minimal circuit size problem. In *Proc. 26th International Computing and Combinatorics Conference (COCOON)* , volume 12273 of *Lecture Notes in Computer Science*, pages 484–495, 2020. doi:10.1007/978-3-030-58150-3_39.

13 Rosario Gennaro, Yael Gertner, Jonathan Katz, and Luca Trevisan. Bounds on the efficiency of generic cryptographic constructions. *SIAM Journal of Computing*, 35(1):217–246, 2005. doi:10.1137/S0097539704443276.

14 Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(3):691–729, 1991. doi:10.1145/116825.116852.

15 Shuichi Hirahara. Non-black-box worst-case to average-case reductions within NP. In *Proc. 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 247–258, 2018. doi:10.1109/FOCS.2018.00032.

16 Shuichi Hirahara. Characterizing average-case complexity of PH by worst-case meta-complexity. In *Proc. 61st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 50–60, 2020. doi:10.1109/FOCS46700.2020.00014.

17 Shuichi Hirahara. Unexpected hardness results for Kolmogorov complexity under uniform reductions. In *Proc. 52nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 1038–1051, 2020. doi:10.1145/3357713.3384251.

18 Shuichi Hirahara. Average-case hardness of NP from exponential worst-case hardness assumptions. In *Proc. 53rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 292–302, 2021. doi:10.1145/3406325.3451065.

19 Shuichi Hirahara and Rahul Santhanam. On the average-case complexity of MCSP and its variants. In *Proc. 32nd Computational Complexity Conference (CCC)*, volume 79 of *LIPIcs*, pages 7:1–7:20, 2017. doi:10.4230/LIPIcs.CCC.2017.7.

20 Shuichi Hirahara and Osamu Watanabe. Limits of minimum circuit size problem as oracle. In *Proc. 31st Computational Complexity Conference (CCC)*, volume 50 of *LIPIcs*, pages 18:1–18:20, 2016. doi:10.4230/LIPIcs.CCC.2016.18.

21 John M. Hitchcock and Aduri Pavan. On the NP-completeness of the minimum circuit size problem. In *Proc. 35th Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, volume 45 of *LIPIcs*, pages 236–245, 2015. doi:10.4230/LIPIcs.FSTTCS.2015.236.

22 Rahul Ilango. Approaching MCSP from above and below: Hardness for a conditional variant and $\mathsf{AC}^0[p]$. In *Proc. 11th Conference on Innovations in Theoretical Computer Science (ITCS)*, volume 151 of *LIPIcs*, pages 34:1–34:26, 2020. doi:10.4230/LIPIcs.ITCS.2020.34.

23 Rahul Ilango. Connecting perebor conjectures: Towards a search to decision reduction for minimizing formulas. In *Proc. 35th Computational Complexity Conference (CCC)*, volume 169 of *LIPIcs*, pages 31:1–31:35, 2020. doi:10.4230/LIPIcs.CCC.2020.31.

24 Rahul Ilango. Constant depth formula and partial function versions of MCSP are hard. In *Proc. 61st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 424–433, 2020. doi:10.1109/FOCS46700.2020.00047.

25 Rahul Ilango, Bruno Loff, and Igor Carboni Oliveira. NP-hardness of circuit minimization for multi-output functions. In *Proc. 35th Computational Complexity Conference (CCC)*, volume 169 of *LIPIcs*, pages 22:1–22:36, 2020. doi:10.4230/LIPIcs.CCC.2020.22.

26 Russell Impagliazzo, Valentine Kabanets, and Antonina Kolokolova. An axiomatic approach to algebrization. In *Proc. 41st Annual ACM Symposium on Theory of Computing (STOC)*, pages 695–704, 2009. doi:10.1145/1536414.1536509.

27 Russell Impagliazzo and Avi Wigderson. Randomness vs time: Derandomization under a uniform assumption. *Journal of Computer and System Sciences*, 63(4):672–688, 2001. doi:10.1006/jcss.2001.1780.

28 Valentine Kabanets and Jin-Yi Cai. Circuit minimization problem. In *Proc. 32nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 73–79, 2000. doi:10.1145/335305.335314.

29 Ker-I Ko. On the complexity of learning minimum time-bounded Turing machines. *SIAM Journal of Computing*, 20(5):962–986, 1991. doi:10.1137/0220059.

30 Leonid A. Levin. Hardness of search problems. Accessed 12-June-2021. URL: https://www.cs.bu.edu/fac/lnd/research/hard.htm.

31 Leonid A. Levin. Universal sequential search problems. *Problemy peredachi informatsii*, 9(3):115–116, 1973.

32 Yanyi Liu and Rafael Pass. On one-way functions and Kolmogorov complexity. In *Proc. 61st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 1243–1254, 2020. doi:10.1109/FOCS46700.2020.00118.

33 Cody D. Murray and R. Ryan Williams. On the (non) NP-hardness of computing circuit complexity. *Theory of Computing*, 13(1):1–22, 2017. doi:10.4086/toc.2017.v013a004.

34 Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149–167, 1994. doi:10.1016/S0022-0000(05)80043-1.

35 Igor Carboni Oliveira. Randomness and intractability in Kolmogorov complexity. In *Proc. 46th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 132 of *LIPIcs*, pages 32:1–32:14, 2019. doi:10.4230/LIPIcs.ICALP.2019.32.

36 Igor Carboni Oliveira and Rahul Santhanam. Conspiracies between learning algorithms, circuit lower bounds, and pseudorandomness. In *Proc. 32nd Computational Complexity Conference (CCC)*, volume 79 of *LIPIcs*, pages 18:1–18:49, 2017. doi:10.4230/LIPIcs.CCC.2017.18.

37 Rafail Ostrovsky and Avi Wigderson. One-way functions are essential for non-trivial zero-knowledge. In *Proc. Second Israel Symposium on Theory of Computing Systems, (ISTCS)*, pages 3–17, 1993. doi:10.1109/ISTCS.1993.253489.

38 Alexander A. Razborov and Steven Rudich. Natural proofs. *Journal of Computer and System Sciences*, 55(1):24–35, 1997. doi:10.1006/jcss.1997.1494.

39 Hanlin Ren and Rahul Santhanam. Hardness of KT characterizes parallel cryptography. In *Proc. 36th Computational Complexity Conference (CCC)*, volume 200 of *LIPIcs*, pages 35:1–35:58, 2021. doi:10.4230/LIPIcs.CCC.2021.35.

40 Hanlin Ren and Rahul Santhanam. A relativization perspective on meta-complexity. *Electron. Colloquium Comput. Complex.*, page 89, 2021. URL: https://eccc.weizmann.ac.il/report/2021/089.

41 Michael Saks and Rahul Santhanam. Circuit lower bounds from NP-hardness of MCSP under Turing reductions. In *Proc. 35th Computational Complexity Conference (CCC)*, volume 169 of *LIPIcs*, pages 26:1–26:13, 2020. doi:10.4230/LIPIcs.CCC.2020.26.

42 Rahul Santhanam. Pseudorandomness and the minimum circuit size problem. In *Proc. 11th Conference on Innovations in Theoretical Computer Science (ITCS)*, volume 151 of *LIPIcs*, pages 68:1–68:26, 2020. doi:10.4230/LIPIcs.ITCS.2020.68.

43 Boris A. Trakhtenbrot. A survey of Russian approaches to perebor (brute-force searches) algorithms. *IEEE Annals of the History of Computing*, 6(4):384–400, 1984. doi:10.1109/MAHC.1984.10036.

44 Luca Trevisan and Salil P. Vadhan. Pseudorandomness and average-case complexity via uniform reductions. *Computational Complexity*, 16(4):331–364, 2007. doi:10.1007/s00037-007-0233-x.

45 Hoeteck Wee. Finding Pessiland. In *Proc. 3rd Theory of Cryptography Conference (TCC)*, volume 3876 of *Lecture Notes in Computer Science*, pages 429–442, 2006. doi:10.1007/11681878_22.

46 David Xiao. On basing ZK $\neq$ BPP on the hardness of PAC learning. In *Proc. 24th Annual IEEE Conference on Computational Complexity (CCC)*, pages 304–315, 2009. doi:10.1109/CCC.2009.11.