NP-Hardness of Approximating Meta-Complexity: A Cryptographic Approach

NP-Hardness of Approximating Meta-Complexity: A Cryptographic Approach

Yizhi Huang
huangyizhi01@gmail.com
IIIS, Tsinghua University
Beijing, China

Rahul Ilango
rilango@mit.edu
Massachusetts Institute of Technology
Cambridge, United States

Hanlin Ren
h4n1in.r3n@gmail.com
University of Oxford
Oxford, United Kingdom

ABSTRACT

It is a long-standing open problem whether the Minimum Circuit Size Problem (MCSP) and related meta-complexity problems are NP-complete. Even for the rare cases where the NP-hardness of meta-complexity problems are known, we only know very weak hardness of approximation.

In this work, we prove NP-hardness of approximating meta-complexity with nearly-optimal approximation gaps. Our key idea is to use *cryptographic constructions* in our reductions, where the security of the cryptographic construction implies the correctness of the reduction. We present both conditional and unconditional hardness of approximation results as follows.

1. Assuming subexponentially-secure witness encryption exists, we prove essentially optimal NP-hardness of approximating conditional time-bounded Kolmogorov complexity ($\mathrm{K}^t(x \mid y)$) in the regime where $t \gg |y|$. Previously, the best hardness of approximation known was a $|x|^{1/\mathrm{poly}(\log\log|x|)}$ factor and only in the sublinear regime ($t \ll |y|$).

2. Unconditionally, we show that for any constant $c > 1$, the Minimum Oracle Circuit Size Problem (MOCSP) is NP-hard to approximate, where Yes instances have circuit complexity at most s, and No instances have circuit complexity at least s^c. Our reduction builds on a witness encryption construction proposed by Garg, Gentry, Sahai, and Waters (STOC'13). Previously, it was unknown whether it is NP-hard to distinguish between oracle circuit complexity s versus $10s \log N$.

3. Finally, we define a "multi-valued" version of MCSP, called mvMCSP, and show that w.p. 1 over a random oracle O, it is NP-hard to approximate mvMCSP^O under quasi-polynomial-time reductions with an O oracle. Intriguingly, this result follows almost directly from the security of Micali's CS Proofs (Micali, SICOMP'00).

In conclusion, we give three results convincingly demonstrating the power of cryptographic techniques in proving NP-hardness of approximating meta-complexity.

CCS CONCEPTS

• **Theory of computation → Circuit complexity**; *Cryptographic primitives*; **Problems, reductions and completeness**.

STOC '23, June 20–23, 2023, Orlando, FL, USA

ACM ISBN 978-1-4503-9913-5/23/06…$15.00
https://doi.org/10.1145/3564246.3585154

KEYWORDS

meta-complexity, Minimum Circuit Size Problem, cryptography, witness encryption, NP-hardness

ACM Reference Format:
Yizhi Huang, Rahul Ilango, and Hanlin Ren. 2023. NP-Hardness of Approximating Meta-Complexity: A Cryptographic Approach. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing (STOC '23), June 20–23, 2023, Orlando, FL, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3564246.3585154

1 INTRODUCTION

Given an object (such as a string or a Boolean function), how hard is it to compute the "computational complexity" of this object? Such questions can be formalised by *meta-complexity* problems which aim to capture the "complexity of complexity" [4]. A prominent example of a meta-complexity problem is the *Minimum Circuit Size Problem* (MCSP) [57]. In MCSP, one is given the length-2^n truth table of a Boolean function $f : \{0,1\}^n \to \{0,1\}$ as well as a size parameter s, and the goal is to determine whether f can be computed by a circuit of size at most s.

Characterising the precise computational complexity of many meta-complexity problems, especially MCSP, remains elusive. It is easy to see that MCSP is a NP (simply guess a circuit of size at most s and check, by brute force[1], that it computes the given truth table). On the other hand, building on the natural proofs framework [36, 39, 74], Kabanets and Cai [57] showed that if one-way functions exist, then MCSP is not in P. Therefore, MCSP is an intractable problem in NP under standard cryptographic assumptions. However, the question of whether MCSP is NP-complete remains wide open. Indeed, Levin is reported to have delayed publishing his theory of NP-completeness [66] in hopes of showing MCSP is NP-complete.[2] Since then, there have been many works investigating whether MCSP and related problems are NP-complete (e.g., [6–8, 10–13, 31, 41–44, 46–50, 53, 57, 59, 70, 72, 78]).

1.1 Why Care About NP-Hardness of Meta-Complexity?

Since we already know that MCSP and other meta-complexity problems are intractable under standard cryptographic assumptions, one may wonder what the motivation is for showing these problems are NP-hard. Perhaps surprisingly, researchers have discovered

[1] This guess and check are non-deterministically efficient since every Boolean function on n-bits has a trivial circuit of size $O(n2^n)$ and we are given the length 2^n truth table as input.
[2] Allender and Das [8] cite a personal communication from Levin regarding this and a discussion can be found on Levin's webpage (https://www.cs.bu.edu/fac/lnd/research/hard.htm, accessed March 26, 2023).

a growing number of important motivations for showing meta-complexity problems are actually NP-hard. We list some that we find compelling:

Eliminating Heuristica. Heuristica is the name Impagliazzo [52] gives to a world where P ≠ NP but NP is easy on average. Unlike other complexity classes such as EXP, PSPACE, or NC^1 [16, 18, 30, 86], there is no known worst-case to average-case reduction for NP. Indeed, there are barrier results against any NP-complete problem having a "black-box" worst-case to average-case reduction [21, 30]. In a breakthrough result, Hirahara [40] overcomes this barrier by giving a non-black-box worst-case to average-case reduction for approximating MCSP. If one could show this approximation version of MCSP is NP-hard, then this would imply that NP *does have* a worst-case to average-case reduction and thus rule out Heuristica.

Later work of Hirahara [43] further extends this result by showing that, to eliminate Heuristica, it suffices to show that a certain additive approximation to GapMINcKT (roughly speaking, a "conditional" version of meta-complexity) is NP-hard.

Basing One-way Functions on P ≠ NP. A longstanding goal in cryptography is to base the existence of one-way functions on worst-case assumptions such as P ≠ NP (or rather NP ⊈ BPP). Recently, an approach to showing this has emerged using meta-complexity [7, 51, 67–70, 75]. In a breakthrough paper, Liu and Pass [67] show that one-way functions exist if and only if time-bounded Kolmogorov complexity is mildly hard on average over the uniform distribution. As mentioned previously, Hirahara's worst-case to average-case reduction [40] also holds for approximating time-bounded Kolmogorov complexity. Thus, if one "just" combines these two results and also shows that approximating time-bounded Kolmogorov complexity is NP-hard, then we would have that one-way functions exist if and only if P ≠ NP. Unfortunately, the results of [40] and [67] do not yet compose, as the types of average-case hardness that they consider are different ([40] considered errorless heuristics while [67] considered error-prone heuristics).

Proving Circuit Lower Bounds. Any reduction from SAT to MCSP needs to generate No instances of MCSP, which is equivalent to circuit lower bounds; therefore, NP-hardness of meta-complexity has a strong connection to circuit lower bounds. This argument was formalized by Kabanets and Cai [57], who show that if MCSP is NP-complete under "natural" reductions[3], then E does not have polynomial-size circuits. Murray and Williams [72] show that any deterministic many-one reduction from SAT to MCSP implies a breakthrough complexity separation: EXP ≠ ZPP. Note that both these results have consequences that we believe but seem hard to show.

We also mention an instance where new circuit lower bounds are proved along the way of pursuing the NP-hardness of meta-complexity. Ilango [49] showed that for every constant d there is a constant $\varepsilon > 0$ and a function whose depth-d and depth-$(d+1)$ *formula* complexity are $2^{\varepsilon n}$ apart. This follows from the techniques used to prove the NP-hardness of MCSP for constant-depth formulas; note that the standard switching lemma arguments [38] are unable to prove such strongly-exponential ($2^{\Omega(n)}$) size lower bounds.

Curiosity. MCSP and its time-bounded Kolmogorov complexity variants are simple and important computational problems that have been studied since at least the 1960s [85]. It is remarkable that despite this long history of study, these problems (unlike thousands of other problems) have thoroughly eluded attempts at classifying their complexity (in particular, completeness for some natural complexity class). Indeed, we lack compelling evidence either for or against the existence of a polynomial-time mapping reduction from SAT to MCSP or many other meta-complexity problems. The situation is especially lacklustre when considering hardness of approximation. Essentially no NP-hardness is known for any even moderately strong model (e.g. depth-3 formulas) beyond logarithmic factors[4] in the truth table [43, 49, 59]. Are these problems NP-complete or not? Are they NP-hard to approximate or not?

1.2 Can Cryptography Help?

The starting point of our work is the following question:

> Can *cryptography* be useful in showing the NP-*hardness* of meta-complexity?

In some sense, prior work already shows that the answer to this question is yes. For example, a trivial corollary of Kabanets and Cai [57] is that if one-way functions exist, then MCSP ∈ P if and only if P = NP. One can view this as a kind of NP-completeness result, but the proof is somewhat unsatisfying: if one-way functions exist, then both P ≠ NP and MCSP ∉ P.

Another (more satisfying) example is a result by Impagliazzo, Kabanets, and Volkovich [53], who show that if indistinguishability obfuscation ($i\mathcal{O}$) exists, then MCSP ∈ ZPP if and only if NP = ZPP. Their proof can be viewed as a *non-black box* reduction from SAT to MCSP. However, one drawback is that assuming $i\mathcal{O}$ exists is very close to assuming that one-way functions exist. In particular, if $i\mathcal{O}$ exists and NP is not in BPP infinitely often, then one-way functions exist [64].

Thus, while these results are interesting, in both cases it is somewhat unclear what the takeaway should be. Do these results really suggest that MCSP is NP-hard, or rather perhaps just that MCSP is intractable based on plausible cryptographic and complexity-theoretic assumptions?

To address this, one can refine the original question.

> Can *cryptography* be useful in showing *black-box* NP-*hardness* of meta-complexity?

Here by a black-box reduction, we mean showing, for example, that one can solve SAT in polynomial time given an oracle to MCSP. Such a result would constitute perhaps the strongest evidence yet that MCSP is indeed NP-complete under the usual definition of NP-completeness.

[3] That is, deterministic reductions whose output length and numerical parameters only depend on the input length (instead of the particular input), and the sizes of the inputs and the outputs are polynomially related. Almost all known NP-complete problems are NP-hard under "natural" reductions.

[4] The only exception to this that we are aware of is Hirahara's recent result that it is NP-hard to compute an $n^{1/\mathrm{poly}\log\log n}$ factor approximation to the conditional time-bounded Kolmogorov complexity. But even this is in a weaker sublinear-time model.

It may seem counter-intuitive that cryptography could be helpful in proving black-box NP-completeness results. While the existence of one-way functions implies that problems like MCSP are intractable [39, 57, 74], it is not at all clear how to turn this into a black-box reduction from say SAT to MCSP.[5]

Intriguingly, a recent breakthrough result by Hirahara [42] uses tools from information-theoretic cryptography, such as secret sharing schemes and one-time encryptions, to show the NP-hardness of many important meta-complexity problems. Indeed, Hirahara's result convincingly demonstrates the power of information-theoretic cryptography for proving NP-hardness of meta-complexity problems.

In this paper, we focus on notions from *computational* cryptography, instead of information-theoretic cryptography. There is a natural intuition for why such cryptography could be useful: it gives *structured computational hardness* one could hope to exploit. In more detail, one potential reason it is difficult to prove the NP-hardness of MCSP is that we lack strong enough circuit lower bounds. Indeed, just deterministically generating a No instance of MCSP requires proving circuit lower bounds! It is hard to imagine giving an NP-hardness result when we cannot even generate an explicit No instance. Moreover, this argument is made formal by several works [57, 72, 78], who showed that NP-hardness of MCSP under certain types of reductions would imply separations in complexity theory such as $\mathrm{EXP} \not\subseteq \mathrm{P}_{/\mathrm{poly}}$.

Thus, since NP-hardness of MCSP (at least in some settings) implies circuit lower bounds, it is natural to wonder whether we can go in the opposite direction: assuming we have circuit lower bounds, can we show meta-complexity problems are NP-hard? So far the answer appears to be *no*. For example, we have subexponential-size lower bounds against AC^0 [1, 32, 38, 88] and $\mathrm{AC}^0[p]$ where p is a prime [73, 83, 84], but the NP-hardness of AC^0-MCSP and $\mathrm{AC}^0[p]$-MCSP remain important open problems.[6] Apparently, to show that MCSP is NP-complete, one needs hardness with some "structure." Can cryptography give such structured hardness?

2 OUR RESULTS

We show three main results, each one using a cryptographic construction (i.e. JLS's indistinguishability obfuscation[7] [54], GGSW's witness encryption [35], or Micali's CS proofs [71]) to get either a conditional or an unconditional NP-hardness result in meta-complexity. Moreover, our results imply NP-hardness of approximation with large approximation gaps. In our view, the central conceptual takeaway from our results is a strongly positive answer to the question above:

> Cryptography is indeed a powerful tool for showing black-box NP-hardness of meta-complexity!

[5] One potential way of doing this is to show that there is a one-way function that is NP-hard to invert. But, as discussed earlier, constructing such a one-way function remains a major open question.

[6] Ilango [49] showed that the *formula* version of AC^0-MCSP is NP-hard under quasi-polynomial-time randomised Turing reductions, but the *circuit* versions of AC^0-MCSP is not known to be NP-hard [24]. Prior to these results, the largest circuit class $\mathscr{C}$ for which NP-hardness of $\mathscr{C}$-MCSP was known is only DNF ∘ XOR [44].

[7] More specifically, we use that the JLS construction implies the existence of witness encryption from well-founded assumptions.

2.1 Witness Encryption and Conditional Time-Bounded Kolmogorov Complexity

The *t-time-bounded Kolmogorov complexity* of a string $x \in \{0,1\}^n$, denoted $\mathrm{K}^t(x)$, is the minimum length of any program that outputs x in time at most t [61, 63, 82]. Similarly, the *conditional t-time-bounded Kolmogorov complexity* of a string $x \in \{0,1\}^n$ given a string $y \in \{0,1\}^m$, denoted $\mathrm{K}^t(x \mid y)$, is the minimum length of any program that outputs x in time t when given oracle access to y. (See Section 2.4 of the full version for formal definitions of $\mathrm{K}^t(\cdot)$ and $\mathrm{K}^t(\cdot \mid \cdot)$.)

In a recent work Hirahara [43] shows that it is NP-hard to approximate $\mathrm{K}^t(x \mid y)$ to a factor of $n^{1/\mathrm{poly}\log\log n}$. This improves on prior work, which could only show an $O(\log n)$ factor hardness of approximation for conditional time-bounded Kolmogorov complexity and related problems [7, 48, 70].

In all of the above NP-hardness results, the instances of $\mathrm{K}^t(x \mid y)$ are in the *sublinear* time regime, i.e. where $t \ll |y|$ and thus one does not even have enough time to read all the bits of y. Intriguingly, Hirahara [43] shows that if one could improve these NP-hardness results to show a certain additive hardness of approximation in the *superlinear* regime where $t \gg |y|$ (so one has time to read all of y), then this would eliminate Heuristica!

This strongly motivates understanding the complexity of conditional time-bounded Kolmogorov complexity in the superlinear regime. Should we expect this problem to be NP-hard? Even if it is, is it NP-hard in the rather specific approximation regime Hirahara needs?

We show that, conditioned on a widely believed cryptographic assumption, this problem is indeed NP-hard with essentially optimal hardness of approximation.

Theorem 2.1 (Informal). *Assume subexponentially-secure witness encryption exists. Then the following promise problem is* NP*-hard under randomized polynomial-time (black-box) reductions: given strings* (x, y) *where* $|x| = n$ *and* $|y| = \mathrm{poly}(n)$*, output*

- *Yes if* $\mathrm{K}^{\mathrm{poly}(n)}(x \mid y) \le n^{.01}$*;*
- *No if* $\mathrm{K}^{2^{n^2}}(x \mid y) \ge n - O(1)$.

We will discuss the notion of witness encryption and its plausibility in a few paragraphs, but before we do that we make some remarks about this theorem. First, we emphasize that, under the assumption, we get a standard, black-box, randomized many-one reduction from NP to the promise problem stated above. To our knowledge, this is the first time an NP-hardness result has been proven conditioned on a cryptographic assumption.

Next, we note that the gap in Theorem 2.1 is essentially maximal NP-hardness of approximation. The complexity of the Yes instances is at most $n^{.01}$ and the constant .01 can be made arbitrarily small (one cannot hope for Yes instances with complexity subpolynomial in n without giving a subexponential time algorithm for SAT). On the other hand, the No instances have complexity at least $n - O(1)$, which is an additive constant away from the maximum complexity of any n-bit string. Moreover, the gap in the time bound is extremely large: $\mathrm{poly}(n)$ in the Yes case versus 2^{n^2} in the No case. (In fact, the choice of n^2 in the exponent is for brevity. In the full version, we show the n^2 in the exponent can be made into an arbitrary polynomial in n)

Finally, we return to Hirahara's approach to eliminating Heuristica. Despite the strong hardness of approximation Theorem 2.1 gives, it does not give the hardness of approximation needed to eliminate Heuristica. The precise reason is somewhat technical (we refer a curious reader to Section 3 of the full version for the details). At a high level, the reason is that the specific additive hardness of approximation Hirahara needs has a somewhat non-standard dependence on the instance $(x \mid y)$, in particular on the "computational depth" of y. The upshot of this is that one needs to give hardness of approximation on instances $(x \mid y)$, where y has low computational depth. It is unclear whether the instances produced by the reduction in Theorem 2.1 have this property. In fact, they likely do not.

Nevertheless, we overcome this, under a further assumption. Assuming the existence of subexponentially secure injective one-way functions, we can modify the reduction in Theorem 2.1 so that with high probability an output $(x \mid y)$ of the reduction will have a y with low computational depth.

Theorem 2.2 (Informal). *Assume subexponentially secure injective one-way functions and subexponentially secure witness encryption exists. Then the promise problem, whose* NP*-completeness was shown to exclude Heuristica by Hirahara [43], is in fact* NP*-complete (under randomized polynomial-time many-one reductions).*

We find Theorem 2.2 rather surprising. One can interpret this result as saying that, under widely believed assumptions in cryptography, Hirahara's approach to eliminating Heuristica *provably* works! Of course, for the purpose of eliminating Heuristica this Theorem 2.2 by itself is not so interesting since if subexponentially secure one-way functions exist, then NP is (automatically) hard on average. Even so, we find this result enlightening, especially because the "ground truth" of whether this problem was in fact NP-hard was not at all clear.

Before we continue, we discuss the notion of witness encryption informally (see Section 2 of the full version for a formal definition). Introduced in [35], witness encryption is a cryptographic primitive that encrypts a message using a (public) instance of some NP-complete language. Let φ be a formula (for example, any satisfying assignment of φ is a proof of Riemann Hypothesis of at most 10,000 pages long). We can encrypt a secret message m (e.g., a Bitcoin address for a prize awarded to whoever proves Riemann Hypothesis) using φ such that:

(1) If φ is satisfiable, then any party with a satisfying assignment of φ (e.g., any mathematician with a valid proof of Riemann Hypothesis) could decrypt the message in polynomial time, and
(2) if φ is unsatisfiable, then the encryption of two different messages should be computationally indistinguishable.

Witness encryption turns out to be a very powerful primitive. It was shown in [35] that witness encryption can be used to build public-key encryption [25, 37], Identity-Based Encryption [22, 80], and Attribute-Based Encryption [77] for circuits. The witness encryption in [35] also yielded the first candidate for Rudich-type secret sharing scheme [19, 65]. In this work, we show an unexpected application of witness encryption in complexity theory: it implies the NP-hardness of meta-complexity problems!

Finally, we discuss the plausibility of witness encryption. Subexponentially secure witness encryption is implied [34] as a special case by the existence of subexponentially secure indistinguishability obfuscators ($i\mathcal{O}$) [17]. In a recent breakthrough paper, Jain, Lin, and Sahai [54] show that subexponentially secure[8] $i\mathcal{O}$ exists assuming four standard "well-founded" assumptions (later work of Jain, Lin, and Sahai reduced this to three assumptions [56]). As a result, the existence of the witness encryption used in Theorem 2.1 has now become a widely-believed assumption.

We also remark that witness encryption is a plausibly weaker assumption than $i\mathcal{O}$. However, the only known constructions (from well-founded assumptions) of subexponentially-secure witness encryption are from $i\mathcal{O}$ (see the recent paper of Vaikuntanathan, Wee, and Wichs [87] for a discussion of this).

2.2 Oracle Witness Encryption and MOCSP

Our second result is about MOCSP, the conditional variant of MCSP. In MOCSP, we are given a truth table of a function $f : \{0,1\}^n \to \{0,1\}$ and a truth table of an oracle $O : \{0,1\}^{O(n)} \to \{0,1\}$, and we are asked to compute the minimum size of any oracle circuit $C : \{0,1\}^n \to \{0,1\}$ that computes f with oracle access to O; we denote by $\mathrm{CC}^O(f)$ this minimum size. Ilango [48] showed that MOCSP is NP-hard to approximate to roughly a logarithmic factor in the input length. Uncertain as to whether or not current techniques could prove stronger hardness of approximation, Ilango left as an open question to either show an N^ϵ factor hardness of approximation for MOCSP for some constant $\epsilon > 0$, where N is the length of the input to MOCSP, or to show a barrier against proving such strong inapproximability results [48, Open Question 1.5].

We resolve this open question by *unconditionally* showing that MOCSP is NP-hard to approximate with a very large approximation factor as follows.

Theorem 2.3 (Informal). *For any $\varepsilon > 0$, the following promise problem is* NP*-hard under polynomial-time randomised mapping reduction: given a truth table f of length ℓ and an oracle truth table O of length* $\mathrm{poly}(\ell)$*, distinguish between the following two cases:*
(Yes instances) $\mathrm{CC}^O(f) \le \ell^\varepsilon$;
(No instances) $\mathrm{CC}^O(f) \ge \ell^{1/2-\varepsilon}$.

Before we discuss the proof techniques, we comment a bit more on the problem MOCSP. As Ilango [48] suggested, MOCSP is a nice "testing ground" for hardness results we conjecture for MCSP. Similar to MCSP, MOCSP is also in NP; it is easy to see that MOCSP is no easier than MCSP. And it is also pointed out by [48] that, essentially the same proof as in [72] shows that if MOCSP is NP-hard under *deterministic* polynomial-time reductions, then EXP $\neq$ ZPP. We will see another example of MOCSP being a "testing ground" for MCSP later (Theorem 2.6). We hope that our results shed some light on the complexity of MCSP.

Perhaps surprisingly, the key idea underlying our proof is again *witness encryption*, despite our proof being *unconditional*. In more detail, our proof utilises the notion of witness encryption *in oracle worlds*, where both the encryption and decryption algorithms have

[8] We note that the definition of subexponentially secure that we need is slightly different from the one explicitly used by Jain, Lin, and Sahai [56], although their result readily generalizes to our definition [55]. See Remark 2.5 of the full version for a detailed discussion of this.

access to an oracle. We show that exponentially-secure witness encryption exists, *unconditionally*, in a carefully constructed oracle world. We also show that such a secure oracle witness encryption scheme implies Theorem 2.3: roughly speaking, we map a formula φ to a function f and an oracle O, where O contains the oracle world as well as a lot of ciphertexts encrypted using φ. If φ is satisfiable, then a small circuit with a satisfying assignment hardcoded can compute f from these ciphertexts easily; if φ is unsatisfiable, then any small circuit computing f would violate the security of witness encryption.

We now discuss how we construct a witness encryption scheme in oracle worlds. A natural approach is to consider candidate witness encryption schemes in the literature and build oracles that make them secure. Fortunately, the original candidate proposed by [35] already suffices. As this candidate uses multilinear maps [23, 33], we replace it with an oracle implementing the *generic multilinear map model* (which is the multilinear map version of the generic group model [81]). It turns out that the security of [35] is provable in the generic multilinear map model! See Sections 4.3 and 4.4 of the full version for details.

A lesson from this result is that unconditional security results in idealised models are not only heuristic arguments that certain cryptographic protocols "seem secure"; they also have (rigorous) implications in complexity theory.

One last aspect we find interesting and worth noting is that, unlike previous results (e.g., [42, 48]), our proof of Theorem 2.3 does not rely on the PCP theorem [14, 15]. Nevertheless, we obtain much stronger hardness of approximation results! The construction in [35] works directly for the NP-complete language ExactCover [58], so our result is also a direct reduction from ExactCover to GapMOCSP. This is in contrast to previous results (e.g., [42, 48]) that need to start with a hardness-of-approximation result (e.g., set cover [27] or the Minimum Monotone Satisfying Assignment problem [2, 26]), which relies on the PCP theorem.

2.3 CS Proofs and A Multi-Valued Version of MCSP with Random Oracles

Our third result is about a "multi-valued" version of MCSP, which we denote as mvMCSP. In mvMCSP, we are given the truth table of a "multi-valued" function $f \subseteq \{0,1\}^n \times \{0,1\}^m$, where for each input $x \in \{0,1\}^n$, any $y \in \{0,1\}^m$ such that $(x,y) \in f$ is a valid output. The goal is to compute the size of the smallest circuit $C : \{0,1\}^n \to \{0,1\}^m$ that computes f, i.e.,

$$\forall x \in \{0,1\}^n, (x, C(x)) \in f.$$

Let $k \geq 1$ be a constant, Gap_k-mvMCSP denotes the following promise problem: given the length-2^{n+m} truth table of a "multi-valued" function $f \subseteq \{0,1\}^n \times \{0,1\}^m$ and a parameter s, distinguish between the following two cases:

(Yes instances) there is a circuit C of size s such that

$$\Pr_{x \leftarrow \{0,1\}^n}[(x, C(x)) \in f] = 1;$$

(No instances) for any circuit C of size s^k,

$$\Pr_{x \leftarrow \{0,1\}^n}[(x, C(x)) \in f] \leq 1/s^k.$$

Theorem 2.4. *For every constant $k > 1$, with probability 1 over a random oracle O, the problem Gap_k-mvMCSP^O is NP-hard under $\mathsf{TIME}[2^{\mathrm{polylog}(n)}]^O$ (deterministic quasi-polynomial time with an O oracle) mapping reductions.*

Perhaps intriguingly, Theorem 2.4 essentially follows from the security of Micali's CS proofs [71] in the random oracle model. We think this is the interesting aspect of Theorem 2.4, as it illustrates the connection between cryptography and NP-hardness of meta-complexity in a *direct and straightforward* way.

Let $L \in \mathsf{NP}$, an *argument system* for L involves a prover and a verifier, where both parties know an instance $x \stackrel{?}{\in} L$ and the prover wants to convince the verifier that $x \in L$. If x is indeed in L, then an efficient prover (with a witness of $x \in L$) could convince the verifier with certainty; if $x \notin L$, then any prover of a certain size could only convince the verifier with small probability.

If one looks carefully at this definition, one realises that this is nothing but a reduction from L to a "meta-complexity" problem! In particular, this is a "meta-complexity" problem about the complexity of convincing the verifier. If $x \in L$, then this complexity should be small, while if $x \notin L$, then this complexity should be large. Therefore, if every language in NP admits an argument system (of some kind), then some meta-complexity problem (related to this argument system) is NP-complete. This is exactly what happens in Theorem 2.4: since every problem in NP has a SNARG (succinct non-interactive argument) in the random oracle model [71], a certain meta-complexity problem should be NP-complete. When we work out the definition of this meta-complexity problem, it becomes exactly mvMCSP.

Moreover, this approach gives us NP-hardness of approximation with "the largest gap possible". If $x \in L$, then the complexity of "convincing the verifier" is a fixed polynomial of $|x|$, since the prover essentially needs to hardwire a witness for x; if $x \notin L$, then by the security of the argument system, the complexity of "convincing the verifier" can be made arbitrarily large (by adjusting the security parameter).

This idea also shows that if (subexponentially-secure) SNARGs exist (in the unrelativised world), then mvMCSP is NP-hard to approximate.

Corollary 2.5. *Suppose that subexponentially-secure SNARGs exist. Then for every $k \in \mathbb{N}$, Gap_k-mvMCSP is NP-hard under deterministic quasi-polynomial time reductions.*

One technical complication of Theorem 2.4 is that, in order to transform a SNARG into NP-hardness of mvMCSP, one needs the security of the SNARG in the *common random string* (CRS) model. That is, both the prover and the verifier receives a (short) CRS before the protocol starts; w.h.p. over the random oracle, for every efficient malicious prover, the probability over a random CRS that the malicious prover proves a false statement successfully is negligible. In contrast, [71] only showed the (weaker) security guarantee in the (plain) random oracle model: for every efficient malicious prover, the probability over a random oracle that the malicious prover proves a false statement successfully is negligible. (Notice the quantifier change here: in the plain model, we fix an adversary and require that a random oracle is secure against this particular

adversary. In contrast, in the CRS model, we want the random oracle to be secure against *every* efficient adversary.) For this reason, a large part of Section 5 of the full version is devoted to proving the security of CS proofs in the CRS model, which we view as an additional technical contribution. See Section 5.1.5 of the full version for more details.

2.4 Applications

Using the ideas developed in this paper, we also make progress on two other problems: pseudorandom self-reductions for NP-complete languages and heuristics for Complexity.

Pseudorandom self-reductions for NP*-complete languages.* In 2017, Hirahara and Santhanam [45] observed that if exponentially-hard one-way functions exist, then MCSP admits a *pseudorandom self-reduction*: a self-reduction that maps a worst-case instance to a distribution that is indistinguishable from the uniform distribution. In contrast, if PH does not collapse, then NP-complete problems do not admit (non-adaptive) *random* self-reductions [21]. Hirahara and Santhanam viewed this result as a property that "distinguishes the MCSP problem from natural NP-complete problems" [45].

Thus it may come as a surprise when Elrazik, Robere, Schuster, and Yehuda [28] showed that NP-complete problems could also admit pseudorandom self-reductions. In particular, under a non-uniform version of the Planted Clique Conjecture, the Clique problem admits a non-adaptive pseudorandom self-reduction. There might be some property that distinguishes MCSP from natural NP-complete problems, but having pseudorandom self-reductions is not one of them!

One weakness of the results in [28] is that they need to assume the Planted Clique Conjecture, which is much stronger than the existence of one-way functions. Moreover, the Planted Clique problem can be solved in $n^{O(\log n)}$ time, which means their distributions are not pseudorandom against adversaries of quasi-polynomial size.

Our NP-hardness results on MOCSP allow us to achieve the best of both worlds: assuming the existence of one-way functions, there is an NP-complete problem with pseudorandom self-reductions.

Theorem 2.6 (Informal). *Assuming one-way functions exist, there is an* NP*-complete problem (namely* GapMOCSP*) that admits pseudorandom self-reductions.*

We remark that hardness of *approximation* is crucial for this application, as our self-reduction blows up the circuit complexity of the input truth tables in MOCSP by a multiplicative factor. (The hardness of approximating Clique is also crucial to the results in [28].)

Heuristics for Complexity. The Complexity problem [60] asks the following: given the truth table of an oracle O, find a truth table f such that the O-oracle circuit complexity of f is large. A random truth table is always hard w.h.p., so there is a trivial randomised algorithm solving Complexity. On the other hand, a deterministic algorithm for Complexity, even only for the case that O is the all-zero truth table, is equivalent to circuit lower bounds for E. Thus deterministic algorithms solving Complexity are of great interest.

We consider *deterministic heuristics* for this problem. We say a deterministic algorithm $\mathcal{A}$ is a *heuristic* for Complexity under the uniform distribution if

$$\Pr_O[\mathrm{CC}^O(f) > 2^n/10n \mid f = \mathcal{A}(O)] \geq 1 - o(1),$$

where f is a truth table of length 2^n.

Inspired by the NP-hardness of Gap-mvMCSP^O for a random oracle O, we design an *unconditional* heuristic for Complexity:

Theorem 2.7 (Informal). *There is,* unconditionally*, a deterministic heuristic for Complexity in certain parameter regimes.*

The idea is simple: if O is a uniformly random input, then solving Complexity means proving circuit lower bounds *in the random oracle model.* Therefore, we can take any proof that E requires large circuits relative to a random oracle, and turn it into a heuristic for Complexity. In fact, our construction is extremely simple: Suppose $O : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is a random oracle over $2n$ bits, then the function $f : \{0,1\}^n \to \{0,1\}$ is defined as

$$f(x) = \bigoplus_{y \in \{0,1\}^n} O(x, y).$$

It is not hard to show that for a random oracle O, the O-oracle circuit complexity of f is exponential.

3 RELATED WORK

For a general survey of meta-complexity, we point the reader to Allender's recent surveys [4, 5] and the references therein. Below we discuss the prior works that are mostly related to our results.

NP*-hardness of meta-complexity problems.* We first discuss works related to Theorems 2.1 and 2.3, several NP-hardness results have been shown for conditional meta-complexity problems. Ilango [48] introduced the problem MOCSP and proved that MOCSP is NP-hard. Allender, Cheraghchi, Myrisiotis, Tirumula, and Volkovich [7] proved the NP-hardness of McKTP, the problem of computing conditional KT-complexity,[9] and Liu and Pass [70] showed that MINcKT, the problem of computing conditional time-bounded Kolmogorov complexity, is NP-complete. In all three aforementioned results, the hardness of approximation given is relatively weak, namely at most a logarithmic factor. This logarithmic factor arises because the reductions begin from set cover, where a logarithmic factor is optimal [27, 29]. A recent exciting work by Hirahara [43] greatly improves the hardness of approximation known for MINcKT, showing it is NP-hard to approximate to a $n^{1/\mathrm{poly}\log\log n}$ factor.

Related to Theorem 2.4, Ilango, Loff, and Oliveira [50] showed that Multi-MCSP is NP-hard under randomised reductions. Here, Multi-MCSP is the problem of computing the circuit complexity of a multi-output function. It is easy to see that Multi-MCSP reduces to mvMCSP. In [50], the number of output bits of the function is *exponential* in the number of input bits, but the hard function is fixed (i.e., any input corresponds to a unique output). On the other hand, in mvMCSP, the number of output bits is only polynomially larger than the number of input bits, but there might be many valid outputs for each input. Thus the two results are not directly comparable.

[9] The KT-complexity is a notion of resource-bounded Kolmogorov complexity defined in [3, 6].

Hirahara [42] proved that MCSP* is NP-hard under randomised reductions. Here, MCSP* is the problem of computing the circuit complexity of a *partial* truth table. Since MCSP* reduces to mvMCSP, it follows that mvMCSP is also NP-hard under randomised reductions.

However, we emphasise that our NP-hardness results hold for *very large* approximation gaps: the Yes instances are computable in size s, while the No instances are inapproximable by size $2^{\mathrm{polylog}(s)}$. The results in [50] only proved the NP-hardness of approximating Multi-MCSP within a small additive factor, and the results in [42] only proved the NP-hardness of approximating MCSP* within a multiplicative factor of n^α for some constant $\alpha < 1$.

Using cryptography to prove hardness of meta-complexity. It is already known from Kabanets and Cai [57] that we can use an MCSP oracle to invert any candidate one-way function. By building concrete (auxiliary-input) one-way function candidates, it was shown that MCSP is hard for discrete logarithm [6, 76], graph isomorphism [9], and actually the whole class SZK [8].

Impagliazzo, Kabanets, and Volkovich [53] show that, assuming indistinguishability obfuscation exists, then NP = ZPP if and only if MCSP $\in$ ZPP. We stress that this is a logical equivalence, not a black-box reduction. We also note that assuming strong cryptographic objects like indistinguishability obfuscation exist is very close to assuming MCSP $\notin$ ZPP (since if one-way functions do exist, then MCSP is not in ZPP).

Intriguingly, we note that Hirahara's recent NP-hardness results for MCSP* [42] and MINcKT [43] utilizes secure secret sharing schemes, a tool from *information theoretic* cryptography. In contrast, our results utilize cryptographic objects that are *computationally secure* either based on a computational assumption or given access to a specifically designed oracle.

Finally, Allender and Hirahara [11] showed that under cryptographic assumptions, a gap version of MCSP is NP-intermediate (i.e., neither in P nor NP-hard). However, the gap they consider is so large that if their version of GapMCSP were NP-hard, then SAT would be in subexponential time.

More comparison with [42]. A recent exciting breakthrough by Hirahara [42] proved the NP-hardness of many meta-complexity problems, including MCSP* and AveMCSP. Here, MCSP* is the problem of determining the circuit complexity of a partial function $f : \{0,1\}^n \to \{0,1,\star\}$, and AveMCSP is the problem of determining the average-case hardness of a (total) function $f : \{0,1\}^n \to \{0,1\}$. These results look tantalisingly close to the NP-hardness of MCSP!

The current paper addresses a few drawbacks of the results in [42]. First, we show a much stronger hardness of approximation than [42]. We prove that MOCSP is NP-hard to approximate within a factor of $N^{\Omega(1)}$, where N is the length of the input truth table. For comparison, [42] only showed that $(\log N)^\alpha$-approximation is NP-hard for some absolute constant $\alpha > 0$. Second, Hirahara's techniques do not seem to yield NP-hardness of circuit minimisation for *total* functions,[10] while we proved NP-hardness of meta-complexity for total strings (MINcKT) and total functions (MOCSP).

[10] Although AveMCSP is a problem about circuit minimisation for total functions, the Yes instances in Hirahara's results are only $(1/2+\varepsilon)$-approximated by small circuits for some small factor $\varepsilon > 0$. In contrast, in Theorem 2.1 and 2.3, the Yes instances are worst-case computable by a small circuit.

It is also interesting to compare our techniques with Hirahara's techniques. To establish his results, Hirahara used secret sharing schemes [20, 79] which is a cryptographic primitive. Both our first result (conditional time-bounded Kolmogorov complexity) and second result (MOCSP) rely on witness encryption [35]. Witness encryption is equivalent to a computational version of secret sharing [65], but it is unclear if there is a unifying framework behind Hirahara's results and our results. We leave this intriguing question for future research.

4 DISCUSSIONS ON BARRIERS RESULTS

There are mainly two barriers to showing NP-hardness of meta-complexity problems: relativisation [62] and oracle independence [46].

Ko [62] showed that any NP-hardness result for MINLT (which is some meta-complexity problem that we do not define here) must be non-relativising. This relativisation barrier was overcome by [42] using non-relativising techniques such as the PCP theorem. Our results are also non-relativising:

- Due to the use of *cryptographic assumptions*, Theorem 2.1 could show consequences that might be impossible to prove unconditionally in a relativising way. However, the proof of Theorem 2.1 (that witness encryption implies NP-hardness of MINcKT) is relativising.
- Theorem 2.3 uses the non-relativising fact that ExactCover is NP-complete. Indeed, the main technical ingredient of Theorem 2.3 is a witness encryption scheme for ExactCover.
- Theorem 2.4 uses the PCP theorem, which is non-relativising. We also note that Theorem 2.4 does *not* show that mvMCSP^O is NP^O-complete, as we could only reduce NP (instead of NP^O) to mvMCSP^O.

It was observed in [46] that most reductions (at their time) to MCSP are *oracle-independent*, i.e., they also work for MCSP^A for *every* oracle A. Then, [46] showed that under plausible assumptions, NP-hardness of MCSP cannot be established via oracle-independent reductions. Hirahara's results [42] are subject to this barrier since they showed the NP-hardness of $(\mathrm{MKTP}^\star)^A$ for every oracle A.

Unfortunately, Theorems 2.3 and 2.4 are also subject to this barrier.[11] In particular, to prove the soundness of our reduction (i.e., the No instances we generated are indeed No instances), we proved strong circuit lower bounds in certain oracle worlds O. These lower bounds hold for not only O-oracle circuits, but also *programs* of bounded query complexity to O (and possibly unbounded time). Therefore, for every fixed (additional) oracle A, these lower bounds also extend to A-oracle circuits. It is a very intriguing question to obtain NP-hardness of (approximating) meta-complexity problems via reductions that are not oracle-independent.

ACKNOWLEDGMENTS

We thank Rahul Santhanam for helpful discussions during the initial stage of this research. We also thank Yilei Chen, Aayush Jain, Huijia Lin, Amit Sahai, Neekon Vafa, and Vinod Vaikuntanathan for

[11] This barrier does not apply to Theorem 2.1 due to the use of cryptographic assumptions.

answering questions about our cryptographic assumptions. During this work, Rahul Ilango was supported by an NSF Graduate Research Fellowship and NSF CCF-1909429.

REFERENCES

[1] Miklós Ajtai. 1983. Σ_1^1-Formulae on finite structures. *Ann. Pure. Appl. Log.* 24, 1 (1983), 1–48. https://doi.org/10.1016/0168-0072(83)90038-6

[2] Michael Alekhnovich, Samuel R. Buss, Shlomo Moran, and Toniann Pitassi. 2001. Minimum Propositional Proof Length Is NP-Hard to Linearly Approximate. *J. Symb. Log.* 66, 1 (2001), 171–191. https://doi.org/10.2307/2694916

[3] Eric Allender. 2001. When Worlds Collide: Derandomization, Lower Bounds, and Kolmogorov Complexity. In *Proc. 21st Foundations of Software Technology and Theoretical Computer Science (FSTTCS) (Lecture Notes in Computer Science, Vol. 2245)*. 1–15. https://doi.org/10.1007/3-540-45294-X_1

[4] Eric Allender. 2017. The Complexity of Complexity. In *Computability and Complexity (Lecture Notes in Computer Science, Vol. 10010)*. Springer, 79–94. https://doi.org/10.1007/978-3-319-50062-1_6

[5] Eric Allender. 2021. Vaughan Jones, Kolmogorov Complexity, and the New Complexity Landscape around Circuit Minimization. *New Zealand Journal of Mathematics* 52 (2021), 585–604. https://doi.org/10.53733/148

[6] Eric Allender, Harry Buhrman, Michal Koucký, Dieter van Melkebeek, and Detlef Ronneburger. 2006. Power from Random Strings. *SIAM Journal of Computing* 35, 6 (2006), 1467–1493. https://doi.org/10.1137/050628994

[7] Eric Allender, Mahdi Cheraghchi, Dimitrios Myrisiotis, Harsha Tirumala, and Ilya Volkovich. 2021. One-Way Functions and a Conditional Variant of MKTP. In *FSTTCS (LIPIcs, Vol. 213)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 7:1–7:19. https://doi.org/10.4230/LIPIcs.FSTTCS.2021.7

[8] Eric Allender and Bireswar Das. 2017. Zero knowledge and circuit minimization. *Information and Computation* 256 (2017), 2–8. https://doi.org/10.1016/j.ic.2017.04.004

[9] Eric Allender, Joshua A. Grochow, Dieter van Melkebeek, Cristopher Moore, and Andrew Morgan. 2018. Minimum Circuit Size, Graph Isomorphism, and Related Problems. *SIAM J. Comput.* 47, 4 (2018), 1339–1372. https://doi.org/10.1137/17M1157970

[10] Eric Allender, Lisa Hellerstein, Paul McCabe, Toniann Pitassi, and Michael E. Saks. 2008. Minimizing Disjunctive Normal Form Formulas and AC^0 Circuits Given a Truth Table. *SIAM J. Comput.* 38, 1 (2008), 63–84. https://doi.org/10.1137/060664537

[11] Eric Allender and Shuichi Hirahara. 2019. New Insights on the (Non-)Hardness of Circuit Minimization and Related Problems. *ACM Transactions on Computation Theory* 11, 4 (2019), 27:1–27:27. https://doi.org/10.1145/3349616

[12] Eric Allender, Dhiraj Holden, and Valentine Kabanets. 2017. The Minimum Oracle Circuit Size Problem. *Comput. Complex.* 26, 2 (2017), 469–496. https://doi.org/10.1007/s00037-016-0124-0

[13] Eric Allender, Rahul Ilango, and Neekon Vafa. 2019. The Non-hardness of Approximating Circuit Size. In *Proc. 14th International Computer Science Symposium in Russia (CSR) (Lecture Notes in Computer Science, Vol. 11532)*. 13–24. https://doi.org/10.1007/978-3-030-19955-5_2

[14] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. 1998. Proof Verification and the Hardness of Approximation Problems. *Journal of the ACM* 45, 3 (1998), 501–555. https://doi.org/10.1145/278298.278306

[15] Sanjeev Arora and Shmuel Safra. 1998. Probabilistic Checking of Proofs: A New Characterization of NP. *Journal of the ACM* 45, 1 (1998), 70–122. https://doi.org/10.1145/273865.273901

[16] László Babai, Lance Fortnow, Noam Nisan, and Avi Wigderson. 1993. BPP Has Subexponential Time Simulations Unless **EXPTIME** has Publishable Proofs. *Computatioanl Complexity* 3 (1993), 307–318. https://doi.org/10.1007/BF01275486

[17] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. 2012. On the (im)possibility of obfuscating programs. *Journal of the ACM* 59, 2 (2012), 6:1–6:48. https://doi.org/10.1145/2160158.2160159

[18] David A. Mix Barrington. 1989. Bounded-Width Polynomial-Size Branching Programs Recognize Exactly Those Languages in NC^1. *J. Comput. Syst. Sci.* 38, 1 (1989), 150–164. https://doi.org/10.1016/0022-0000(89)90037-8

[19] Amos Beimel. 2011. Secret-Sharing Schemes: A Survey. In *IWCC (Lecture Notes in Computer Science, Vol. 6639)*. Springer, 11–46. https://doi.org/10.1007/978-3-642-20901-7_2

[20] George Robert Blakley. 1979. Safeguarding cryptographic keys. In *International Workshop on Managing Requirements Knowledge (MARK)*. IEEE, 313–318. https://doi.org/10.1109/MARK.1979.8817296

[21] Andrej Bogdanov and Luca Trevisan. 2006. On Worst-Case to Average-Case Reductions for NP Problems. *SIAM Journal of Computing* 36, 4 (2006), 1119–1159. https://doi.org/10.1137/S0097539705446974

[22] Dan Boneh and Matthew K. Franklin. 2003. Identity-Based Encryption from the Weil Pairing. *SIAM J. Comput.* 32, 3 (2003), 586–615. https://doi.org/10.1137/S0097539701398521

[23] Dan Boneh and Alice Silverberg. 2003. Applications of Multilinear Forms to Cryptography. In *Contemporary Mathematics*. Vol. 324. American Mathematical Society, 71–90. https://doi.org/10.1090/conm/324/05731

[24] Marco Carmosino, Kenneth Hoover, Russell Impagliazzo, Valentine Kabanets, and Antonina Kolokolova. 2021. Lifting for Constant-Depth Circuits and Applications to MCSP. In *ICALP (LIPIcs, Vol. 198)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 44:1–44:20. https://doi.org/10.4230/LIPIcs.ICALP.2021.44

[25] Whitfield Diffie and Martin E. Hellman. 1976. New directions in cryptography. *IEEE Transactions on Information Theory* 22, 6 (1976), 644–654. https://doi.org/10.1109/TIT.1976.1055638

[26] Irit Dinur and Shmuel Safra. 2004. On the hardness of approximating label-cover. *Inf. Process. Lett.* 89, 5 (2004), 247–254. https://doi.org/10.1016/j.ipl.2003.11.007

[27] Irit Dinur and David Steurer. 2014. Analytical approach to parallel repetition. In *STOC*. ACM, 624–633. https://doi.org/10.1145/2591796.2591884

[28] Reyad Abed Elrazik, Robert Robere, Assaf Schuster, and Gal Yehuda. 2022. Pseudorandom Self-Reductions for NP-Complete Problems. In *ITCS (LIPIcs, Vol. 215)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 65:1–65:12. https://doi.org/10.4230/LIPIcs.ITCS.2022.65

[29] Uriel Feige. 1998. A Threshold of $\ln n$ for Approximating Set Cover. *J. ACM* 45, 4 (1998), 634–652. https://doi.org/10.1145/285055.285059

[30] Joan Feigenbaum and Lance Fortnow. 1993. Random-Self-Reducibility of Complete Sets. *SIAM J. Comput.* 22, 5 (1993), 994–1005. https://doi.org/10.1137/0222061

[31] Vitaly Feldman. 2009. Hardness of approximate two-level logic minimization and PAC learning with membership queries. *J. Comput. Syst. Sci.* 75, 1 (2009), 13–26. https://doi.org/10.1016/j.jcss.2008.07.007

[32] Merrick L. Furst, James B. Saxe, and Michael Sipser. 1984. Parity, Circuits, and the Polynomial-Time Hierarchy. *Math. Syst. Theory* 17, 1 (1984), 13–27. https://doi.org/10.1007/BF01744431

[33] Sanjam Garg, Craig Gentry, and Shai Halevi. 2012. Candidate Multilinear Maps from Ideal Lattices and Applications. *IACR Cryptol. ePrint Arch.* (2012), 610. http://eprint.iacr.org/2012/610

[34] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. 2016. Candidate Indistinguishability Obfuscation and Functional Encryption for All Circuits. *SIAM J. Comput.* 45, 3 (2016), 882–929. https://doi.org/10.1137/14095772X

[35] Sanjam Garg, Craig Gentry, Amit Sahai, and Brent Waters. 2013. Witness encryption and its applications. In *STOC*. ACM, 467–476. https://doi.org/10.1145/2488608.2488667

[36] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. 1986. How to construct random functions. *Journal of the ACM* 33, 4 (1986), 792–807. https://doi.org/10.1145/6490.6503

[37] Shafi Goldwasser and Silvio Micali. 1984. Probabilistic Encryption. *J. Comput. Syst. Sci.* 28, 2 (1984), 270–299. https://doi.org/10.1016/0022-0000(84)90070-9

[38] Johan Håstad. 1986. Almost Optimal Lower Bounds for Small Depth Circuits. In *STOC*. ACM, 6–20. https://doi.org/10.1145/12130.12132

[39] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. 1999. A Pseudorandom Generator from any One-way Function. *SIAM Journal of Computing* 28, 4 (1999), 1364–1396. https://doi.org/10.1137/S0097539793244708

[40] Shuichi Hirahara. 2018. Non-Black-Box Worst-Case to Average-Case Reductions within NP. In *FOCS*. 247–258. https://doi.org/10.1109/FOCS.2018.00032

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

[42] Shuichi Hirahara. 2022. NP-Hardness of learning programs and partial MCSP. In *FOCS*. https://eccc.weizmann.ac.il/report/2022/119

[43] Shuichi Hirahara. 2022. Symmetry of Information from Meta-Complexity. In *CCC (LIPIcs, Vol. 234)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 26:1–26:41. https://doi.org/10.4230/LIPIcs.CCC.2022.26

[44] Shuichi Hirahara, Igor Carboni Oliveira, and Rahul Santhanam. 2018. NP-hardness of Minimum Circuit Size Problem for OR-AND-MOD Circuits. In *CCC (LIPIcs, Vol. 102)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 5:1–5:31. https://doi.org/10.4230/LIPIcs.CCC.2018.5

[45] Shuichi Hirahara and Rahul Santhanam. 2017. On the Average-Case Complexity of MCSP and Its Variants. In *CCC (LIPIcs, Vol. 79)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 7:1–7:20. https://doi.org/10.4230/LIPIcs.CCC.2017.7

[46] Shuichi Hirahara and Osamu Watanabe. 2016. Limits of Minimum Circuit Size Problem as Oracle. In *Proc. 31st Computational Complexity Conference (CCC) (LIPIcs, Vol. 50)*. 18:1–18:20. https://doi.org/10.4230/LIPIcs.CCC.2016.18

[47] John M. Hitchcock and Aduri Pavan. 2015. 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) (LIPIcs, Vol. 45)*. 236–245. https://doi.org/10.4230/LIPIcs.FSTTCS.2015.236

[48] Rahul Ilango. 2020. Approaching MCSP from Above and Below: Hardness for a Conditional Variant and $AC^0[p]$. In *Proc. 11th Conference on Innovations in Theoretical Computer Science (ITCS) (LIPIcs, Vol. 151)*. 34:1–34:26. https://doi.org/10.4230/LIPIcs.ITCS.2020.34

[49] Rahul Ilango. 2020. Constant Depth Formula and Partial Function Versions of MCSP are Hard. In *Proc. 61st Annual IEEE Symposium on Foundations of Computer*

Science (FOCS). 424–433. https://doi.org/10.1109/FOCS46700.2020.00047
[50] Rahul Ilango, Bruno Loff, and Igor Carboni Oliveira. 2020. NP-Hardness of Circuit Minimization for Multi-Output Functions. In *CCC (LIPIcs, Vol. 169)*. 22:1–22:36. https://doi.org/10.4230/LIPIcs.CCC.2020.22
[51] Rahul Ilango, Hanlin Ren, and Rahul Santhanam. 2022. Robustness of average-case meta-complexity via pseudorandomness. In *STOC*. ACM, 1575–1583. https://doi.org/10.1145/3519935.3520051
[52] Russell Impagliazzo. 1995. A Personal View of Average-Case Complexity. In *Proc. 10th Annual Structure in Complexity Theory Conference*. 134–147. https://doi.org/10.1109/SCT.1995.514853
[53] Russell Impagliazzo, Valentine Kabanets, and Ilya Volkovich. 2018. The Power of Natural Properties as Oracles. In *Proc. 33rd Computational Complexity Conference (CCC) (LIPIcs, Vol. 102)*. 7:1–7:20. https://doi.org/10.4230/LIPIcs.CCC.2018.7
[54] Aayush Jain, Huijia Lin, and Amit Sahai. 2021. Indistinguishability obfuscation from well-founded assumptions. In *STOC*. ACM, 60–73. https://doi.org/10.1145/3406325.3451093
[55] Aayush Jain, Huijia Lin, and Amit Sahai. 2022. Personal Communication.
[56] Aayush Jain, Huijia Lin, and Amit Sahai. 2022. Indistinguishability Obfuscation from LPN over $\mathbb{F}_p$, DLIN, and PRGs in NC^0. In *EUROCRYPT (1) (Lecture Notes in Computer Science, Vol. 13275)*. Springer, 670–699. https://doi.org/10.1007/978-3-031-06944-4_23
[57] Valentine Kabanets and Jin-Yi Cai. 2000. Circuit minimization problem. In *Proc. 32nd Annual ACM Symposium on Theory of Computing (STOC)*. 73–79. https://doi.org/10.1145/335305.335314
[58] Richard M. Karp. 1972. Reducibility Among Combinatorial Problems. In *Complexity of Computer Computations (The IBM Research Symposia Series)*. 85–103. https://doi.org/10.1007/978-1-4684-2001-2_9
[59] Subhash Khot and Rishi Saket. 2008. Hardness of Minimizing and Learning DNF Expressions. In *FOCS*. IEEE Computer Society, 231–240. https://doi.org/10.1109/FOCS.2008.37
[60] Robert Kleinberg, Oliver Korten, Daniel Mitropolsky, and Christos H. Papadimitriou. 2021. Total Functions in the Polynomial Hierarchy. In *ITCS (LIPIcs, Vol. 185)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 44:1–44:18. https://doi.org/10.4230/LIPIcs.ITCS.2021.44
[61] Ker-I Ko. 1986. On the Notion of Infinite Pseudorandom Sequences. *Theor. Comput. Sci.* 48, 3 (1986), 9–33. https://doi.org/10.1016/0304-3975(86)90081-2
[62] Ker-I Ko. 1991. On the Complexity of Learning Minimum Time-Bounded Turing Machines. *SIAM Journal of Computing* 20, 5 (1991), 962–986. https://doi.org/10.1137/0220059
[63] Andrei N Kolmogorov. 1965. Three approaches to the quantitative definition of information. *Problems of information transmission* (1965). https://doi.org/10.1080/00207166808803030
[64] Ilan Komargodski, Tal Moran, Moni Naor, Rafael Pass, Alon Rosen, and Eylon Yogev. 2014. One-Way Functions and (Im)Perfect Obfuscation. In *FOCS*. IEEE Computer Society, 374–383. https://doi.org/10.1109/FOCS.2014.47
[65] Ilan Komargodski, Moni Naor, and Eylon Yogev. 2017. Secret-Sharing for NP. *J. Cryptol.* 30, 2 (2017), 444–469. https://doi.org/10.1007/s00145-015-9226-0
[66] Leonid Anatolevich Levin. 1973. Universal sequential search problems. *Problemy peredachi informatsii* 9, 3 (1973), 115–116.
[67] Yanyi Liu and Rafael Pass. 2020. On One-way Functions and Kolmogorov Complexity. In *Proc. 61st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 1243–1254. https://doi.org/10.1109/FOCS46700.2020.00118
[68] Yanyi Liu and Rafael Pass. 2021. Cryptography from sublinear-time average-case hardness of time-bounded Kolmogorov complexity. In *STOC*. ACM, 722–735. https://doi.org/10.1145/3406325.3451121
[69] Yanyi Liu and Rafael Pass. 2021. On the Possibility of Basing Cryptography on EXP ≠ BPP. In *Proc. 41st Annual International Cryptology Conference (CRYPTO) (Lecture Notes in Computer Science, Vol. 12825)*. Springer, 11–40. https://doi.org/10.1007/978-3-030-84242-0_2
[70] Yanyi Liu and Rafael Pass. 2022. On One-Way Functions from NP-Complete Problems. In *CCC (LIPIcs, Vol. 234)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 36:1–36:24. https://doi.org/10.4230/LIPIcs.CCC.2022.36
[71] Silvio Micali. 2000. Computationally Sound Proofs. *SIAM J. Comput.* 30, 4 (2000), 1253–1298. https://doi.org/10.1137/S0097539795284959
[72] Cody D. Murray and R. Ryan Williams. 2017. On the (Non) NP-Hardness of Computing Circuit Complexity. *Theory of Computing* 13, 1 (2017), 1–22. https://doi.org/10.4086/toc.2017.v013a004
[73] Alexander A. Razborov. 1987. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical Notes of the Academy of Sciences of the USSR* 41, 4 (1987), 333–338.
[74] Alexander A. Razborov and Steven Rudich. 1997. Natural Proofs. *Journal of Computer and System Sciences* 55, 1 (1997), 24–35. https://doi.org/10.1006/jcss.1997.1494
[75] Hanlin Ren and Rahul Santhanam. 2021. Hardness of KT Characterizes Parallel Cryptography. In *Proc. 36th Computational Complexity Conference (CCC) (LIPIcs, Vol. 200)*. 35:1–35:58. https://doi.org/10.4230/LIPIcs.CCC.2021.35
[76] Michael Rudow. 2017. Discrete Logarithm and Minimum Circuit Size. *Inf. Process. Lett.* 128 (2017), 1–4. https://doi.org/10.1016/j.ipl.2017.07.005
[77] Amit Sahai and Brent Waters. 2005. Fuzzy Identity-Based Encryption. In *EUROCRYPT (Lecture Notes in Computer Science, Vol. 3494)*. Springer, 457–473. https://doi.org/10.1007/11426639_27
[78] Michael Saks and Rahul Santhanam. 2020. Circuit Lower Bounds from NP-Hardness of MCSP Under Turing Reductions. In *Proc. 35th Computational Complexity Conference (CCC) (LIPIcs, Vol. 169)*. 26:1–26:13. https://doi.org/10.4230/LIPIcs.CCC.2020.26
[79] Adi Shamir. 1979. How to Share a Secret. *Commun. ACM* 22, 11 (1979), 612–613. https://doi.org/10.1145/359168.359176
[80] Adi Shamir. 1984. Identity-Based Cryptosystems and Signature Schemes. In *CRYPTO (Lecture Notes in Computer Science, Vol. 196)*. Springer, 47–53. https://doi.org/10.1007/3-540-39568-7_5
[81] Victor Shoup. 1997. Lower Bounds for Discrete Logarithms and Related Problems. In *EUROCRYPT (Lecture Notes in Computer Science, Vol. 1233)*. Springer, 256–266. https://doi.org/10.1007/3-540-69053-0_18
[82] Michael Sipser. 1983. A Complexity Theoretic Approach to Randomness. In *STOC*. ACM, 330–335. https://doi.org/10.1145/800061.808762
[83] Roman Smolensky. 1987. Algebraic Methods in the Theory of Lower Bounds for Boolean Circuit Complexity. In *STOC*. ACM, 77–82. https://doi.org/10.1145/28395.28404
[84] Roman Smolensky. 1993. On Representations by Low-Degree Polynomials. In *FOCS*. IEEE Computer Society, 130–138. https://doi.org/10.1109/SFCS.1993.366874
[85] Boris A. Trakhtenbrot. 1984. A Survey of Russian Approaches to Perebor (Brute-Force Searches) Algorithms. *IEEE Annals of the History of Computing* 6, 4 (1984), 384–400. https://doi.org/10.1109/MAHC.1984.10036
[86] Luca Trevisan and Salil P. Vadhan. 2007. Pseudorandomness and Average-Case Complexity Via Uniform Reductions. *Comput. Complex.* 16, 4 (2007), 331–364. https://doi.org/10.1007/s00037-007-0233-x
[87] Vinod Vaikuntanathan, Hoeteck Wee, and Daniel Wichs. 2022. Witness Encryption and Null-IO from Evasive LWE. *IACR Cryptol. ePrint Arch.* (2022), 1140. https://eprint.iacr.org/2022/1140
[88] Andrew Chi-Chih Yao. 1985. Separating the Polynomial-Time Hierarchy by Oracles (Preliminary Version). In *FOCS*. IEEE Computer Society, 1–10. https://doi.org/10.1109/SFCS.1985.49

Received 2022-11-07; accepted 2023-02-06