Electronic Colloquium on Computational Complexity, Report No. 79 (2023)

Mutual Empowerment between Circuit Obfuscation and Circuit Minimization

Russell Impagliazzo [*] Valentine Kabanets [†] Ilya Volkovich [‡]

May 30, 2023

Abstract

We study close connections between Indistinguishability Obfuscation (IO) and the Minimum Circuit Size Problem (MCSP), and argue that algorithms for one of MCSP or IO would empower the other one. Some of our main results are:

- If there exists a perfect (imperfect) IO that is computationally secure against nonuniform polynomial-size circuits, then for all $k \in \mathbb{N}$: $\mathsf{NP} \cap \mathsf{ZPP}^{\mathsf{MCSP}} \not\subseteq \mathsf{SIZE}[n^k]$ $(\mathsf{MA} \cap \mathsf{ZPP}^{\mathsf{MCSP}} \not\subseteq \mathsf{SIZE}[n^k])$.
- In addition, if there exists a perfect IO that is computationally secure against nonuniform polynomial-size circuits, then $\mathsf{NEXP} \cap \mathsf{ZPEXP}^{\mathsf{MCSP}} \not\subseteq \mathsf{P/poly}$.
- If $\mathsf{MCSP} \in \mathsf{BPP}$, then statistical security and computational security for IO are equivalent.
- If computationally-secure perfect IO exists, then $\mathsf{MCSP} \in \mathsf{BPP}$ iff $\mathsf{NP} = \mathsf{ZPP}$.
- If computationally-secure perfect IO exists, then $\mathsf{ZPEXP} \neq \mathsf{BPP}$.

To the best of our knowledge, this is the first consequence of strong circuit lower bounds from the existence of an IO. The results are obtained via a construction of an optimal *universal distinguisher*, computable in randomized polynomial time with access to the MCSP oracle, that will distinguish any two circuit-samplable distributions with the advantage that is the statistical distance between these two distributions minus some negligible error term. This is our main technical contribution. As another immediate application, we get a simple proof of the result by Allender and Das (*Inf. Comput.*, 2017) that $\mathsf{SZK} \subseteq \mathsf{BPP}^{\mathsf{MCSP}}$.

[*] Department of Computer Science, University of California San Diego, La Jolla, CA. Email: russell@cs.ucsd.edu

[†] School of Computing Science, Simon Fraser University, Burnaby, BC, Canada. Email: kabanets@cs.sfu.ca

[‡] Computer Science Department, Boston College, Chestnut Hill, MA. Email: ilya.volkovich@bc.edu

ISSN 1433-8092

Contents

1 Introduction

Circuit Obfuscation. The main purpose of program obfuscation is to transform a given program into an "unintelligible" one, while preserving the program's original functionality. A natural way to represent a program is via a Boolean circuit. Given that, the most common notion of obfuscation is the notion of *indistinguishability obfuscation*, introduced in [Bar+12]. Roughly speaking, a (potentially) randomized procedure IO is an *indistinguishability obfuscator*, if the obfuscations of two circuits C_1 and C_2 of the same size and functionality are "indistiguishable". In other words, no algorithm can "distinguish" between the outputs of $\mathsf{IO}(C_1)$ and $\mathsf{IO}(C_2)$ with a "noticeable" advantage.

The kind of security provided by the IO is defined by the class of the allowed distinguishing algorithms. More formally, consider a particular class of algorithms $\mathcal{A}$ and ask whether IO is "secure against" $\mathcal{A}$. For example, if $\mathcal{A}$ is the class of *all* (possibly inefficient) algorithms, we say that IO is *statistically* secure. On the other hand, if $\mathcal{A}$ is the class of *efficient* (i.e. randomized polynomial-time) algorithms, we say that IO is *computationally* secure.

The correctness of an IO procedure is called *perfect* if the functionality of the input circuit is preserved with probability one (over the internal randomness of the IO), or *imperfect* if the functionality is preserved with high probability only.

Circuit obfuscation turned out to be a very useful tool in many cryptographic and complexity-theoretic applications, see, e.g., [Gar+16; SW21; GP15; BZ17; KNY17]. The past decade saw numerous candidate constructions, culminating with the work of [JLS21]. Yet, identifying the exact necessary and sufficient conditions for the existence of indistinguishability obfuscators remains an important open question. One reason for that is that unlike the vast majority of cryptographic primitives, obfuscators could still exist even if $\mathsf{P} = \mathsf{NP}$! In fact, in this case we get an "ultimate" obfuscator: for each circuit C, the IO will output some canonical equivalent $\hat{C}$[1].

The place of IO within the Five Worlds. Thus, in the language of Impagliazzo's Five Worlds [Imp95], an IO exists in Algorithmica. The work of [JLS21], on the other hand, makes a good argument that an IO may exist in Cryptomania. What about the other three worlds: Heuristica, Pessiland, and Minicrypt? It turns out that none of these remaining three worlds can accommodate an IO. The results of [SW21] show that an IO plus a one-way function imply public key encryption (and more), and hence IO cannot exist in Minicrypt. The results of [Kom+14] essentially show that an (even imperfect) IO cannot exist in Pessiland: if there are no one-way functions but an (imperfect) IO exists, then $\mathsf{NP} \subseteq \mathsf{io\text{-}BPP}$. This result also rules out Heuristica as a possible home for an IO. We will prove a stronger connection: if an (imperfect) IO exists in Heuristica (where $\mathsf{DistNP} \subseteq \mathsf{AvgP}$), then $\mathsf{NP} = \mathsf{P}$ (see Theorem 8.1 below).

So IO can exist in either Algorithmica or Cryptomania. Many of the results that we shall present in this paper can be viewed as instantiations of this fact, for various settings of parameters of IO: *If you assume* IO *exists, and assume something that threatens the existence of cryptography, then you find yourself in* Algorithmica.

Circuit Minimization. Minimum Circuit Size Problem (MCSP) [Tra84; KC00] asks for a given truth table of an n-variate Boolean function $f\colon \{0,1\}^n \to \{0,1\}$ and a parameter $0 \leq s \leq 2^n$, if f is computable by a Boolean circuit of size at most s. It is easy to see that $\mathsf{MCSP} \in \mathsf{NP}$.

[1] For example, given a circuit C one can find the lexicographically-smallest, equivalent circuit $\hat{C}$ in PH.

Yet, it is unknown if MCSP is NP-hard, or if MCSP is easy, say in BPP. What is known is that MCSP is powerful enough to "kill" cryptography. That is, any one-way function candidate can be efficiently inverted on average by a randomized polynomial-time algorithm with access to the MCSP oracle [RR97; All+06]. Hence, an efficient algorithm for MCSP cannot exist in $\mathsf{Minicrypt}$ or $\mathsf{Cryptomania}$.[2]

Interplay between IO and MCSP. By the preceding discussion, if we assume that both an IO exists and that MCSP is "easy", then we should get that NP is also "easy" (as we must be in some version of $\mathsf{Algorithmica}$)[3]. In fact, we shall argue that MCSP and IO *mutually empower each other*. We show results where assumed existence of an appropriate version of IO makes MCSP more powerful that it is known to be. And, conversely, we show results where assumed "easiness" of MCSP makes an IO stronger (i.e., more secure). We state some of our main results next.

1.1 Our Main Results

We show that the existence of an (even imperfect) IO secure against $\mathsf{P/poly}$ implies new circuit lower bounds.

Theorem 1. *Suppose there exists a perfect* IO *secure against* $\mathsf{P/poly}$. *Then:*

1. $\mathsf{NEXP} \cap \mathsf{ZPEXP}^{\mathsf{MCSP}} \not\subseteq \mathsf{P/poly}$.
2. *For all* $k \in \mathbb{N}$: $\mathsf{NP} \cap \mathsf{ZPP}^{\mathsf{MCSP}} \not\subseteq \mathsf{SIZE}[n^k]$.

Theorem 2. *Suppose there exists an imperfect* IO *secure against* $\mathsf{P/poly}$. *Then for all* $k \in \mathbb{N}$: $\mathsf{MA} \cap \mathsf{ZPP}^{\mathsf{MCSP}} \not\subseteq \mathsf{SIZE}[n^k]$.

The two preceding theorems should be contrasted with the unconditional circuit lower bounds proved in [San09] and [IKV18]. There it is shown that $\mathsf{ZPEXP}^{\mathsf{MCSP}} \not\subseteq \mathsf{P/poly}$ and that, for every $k > 0$, $\mathsf{ZPP}^{\mathsf{MCSP}}/1 \not\subseteq \mathsf{SIZE}[n^k]$ and $\mathsf{MA}/1 \not\subseteq \mathsf{SIZE}[n^k]$. Although removing the extra bit of advice from the lower bounds may seem incremental, it actually has been a long standing open problem that resisted many attempts! Indeed, the same issue arises in other instances involving lower bounds for randomized complexity classes; see, e.g., [Bar02; FS04; MP07; Vol14]. Additionally, while widely *believed* to be true, showing that $\mathsf{NEXP} \not\subseteq \mathsf{P/poly}$ seems to require techniques beyond our current reach. For a further discussion, see the seminal paper of Williams [Wil14] where it was shown that $\mathsf{NEXP} \not\subseteq \mathsf{ACC}$. In conclusion, the two new theorems above prove stronger circuit lower bounds, but under an assumption that a certain IO exists. One interpretation of that is that a construction of these kinds of IO will require novel techniques.

Our next result is a uniform version of Theorem 1.

Theorem 3. *Suppose there exists a computationally-secure perfect* IO. *Then* $\mathsf{ZPEXP} \neq \mathsf{BPP}$.

While we do not have hierarchy theorems for randomized complexity classes, one can show that $\mathsf{ZPEXP} \neq \mathsf{ZPP}$ (see Appendix C). Yet, separating ZPEXP (or even NEXP and $\mathsf{EXP}^{\mathsf{NP}}$) from BPP is

[2] In fact, even an efficient one-sided average-case algorithm for MCSP (i.e., an efficiently computable natural property in the sense of [RR97], which is useful against exponential-size circuits) would "kill" one-way functions.

[3] This observation was made in [IKV18] and previously a similar observation was made in [Kom+14].

a longstanding open problem (see e.g. [BT00; Wil13]). In that sense our result resolves the problem under the assumption that a computationally-secure perfect IO exists.

The following theorems are examples of results where MCSP empowers IO, and where IO empowers MCSP. While, for simplicity, we state our results below for MCSP, it could be replaced with any other natural property $\mathcal{R}$ in the sense of Razborov and Rudich [RR97] (i.e. having largeness and "exponential" usefulness against $\mathsf{P/poly}$, with constructivity being in the premise or replaced with oracle access). Such properties include: approximations of MCSP, average-case approximations to MCSP with one-sided error, "gap" (promise) versions of MCSP, MCSP^B for B-oracle circuits, and others.

Theorem 4. *An IO (both imperfect and perfect) is statistically-secure if and only if it is secure against $\mathsf{FBPP}^{\mathsf{MCSP}}$. Hence, assuming $\mathsf{MCSP} \in \mathsf{BPP}$, statistically-secure IO exists if and only if computationally-secure IO exists.*

Theorem 5. *Let $\Gamma \in \{\mathsf{ZPP}, \mathsf{BPP}\}$. Suppose there exists a computationally-secure imperfect IO. Then $\mathsf{MCSP} \in \Gamma$ iff $\mathsf{NP} \subseteq \Gamma$.*

Note that Theorem 5 strengthens a similar result of [IKV18] to the imperfect setting.

Theorem 6. *Suppose there exists a computationally-secure perfect IO. Then $\mathsf{MCSP} \in \mathsf{BPP}$ iff $\mathsf{NP} = \mathsf{ZPP}$.*

Remark 1.1. *Note that all the results still hold true if we only have an obfuscator IO for a class of circuits $\mathcal{C}$ for which the* equivalence *problem (i.e., testing if two given circuits $C_0, C_1 \in \mathcal{C}$ agree on all inputs) is coNP-hard such as: 3-CNFs (even read-thrice 3-CNFs), read-twice depth-3 formulas, monotone depth-3 formulas[4] and others. All these circuit classes are small subsets of NC^1, which is the starting points of most candidate IO constructions.*

Remark 1.2. *Recall that the results of [Kom+14] show that if there are no one-way functions yet an imperfect IO exists, then $\mathsf{NP} \subseteq \mathsf{io}\text{-}\mathsf{BPP}$. The authors subsequently pose an open problem to get a similar result only relying on an obfuscator for 3-CNFs. While we do not solve their open problem, we believe that Theorem 5, adjusted according to the previous remark, can be viewed as partial progress towards the resolution of the problem especially in light of the recent characterizations of one-way functions in terms of "MCSP"-like problems [LP20; All+21].*

1.2 Our Techniques

Our main technical tool is a *universal distinguisher* that, given any two circuits C_0 and C_1 that are samplers for some distributions D_0 and D_1, will distinguish between D_0 and D_1 essentially as best as information-theoretically possible (with the distinguishing advantage equal to the statistical distance between D_0 and D_1 minus a negligible error term). We show (see Corollary 3.6) that such a universal distinguisher is computable in $\mathsf{FBPP}^{\mathsf{MCSP}}$[5]. The main idea is to use a distributional inverter and a connection between one-way functions and distributional one-way functions

[4] Monotone depth-3 formulas are the only class on the list for which the equivalence problem is coNP-hard [EG95], but the satisfiability problem is trivial. See Section B in Appendix.

[5] $\mathsf{FBPP}^{\mathsf{MCSP}}$ denotes the class of randomized polynomial-time algorithms with MCSP oracle. As was mentioned earlier, MCSP oracle can be replaced with an oracle to any (at least) inverse-exponentially large natural property $\mathcal{R}$ that is exponentially useful against $\mathsf{P/poly}$.

from [IL89]. In particular, we argue (see Lemma 3.1) that a distributional inverter suffices to get a distinguisher for **any** two circuit-samplable distributions D_0 and D_1. We then use the result of [All+06] that allows to invert any candidate one-way (and, in fact, any polynomial-time computable) function in randomized-polynomial time given an MCSP oracle (see Lemma 2.13 for more details). Indeed, we generalize the inverter of [All+06] to get a *distributional* inverter for any candidate distributional one-way function (see Lemma 2.14). We believe that this extension could be of independent interest.

We note, however, that it is fairly easy to construct a universal distinguisher in $\mathsf{FBPP}^{\mathsf{SAT}}$ by approximating the "maximum likelihood" distinguisher using the well-known fact that approximate counting can be done in $\mathsf{FBPP}^{\mathsf{NP}}$ [JVV86]. For completeness, we provide the full proof in Theorem A.2 of the appendix. From this perspective, our construction constitutes another example of a computational task that can still be performed with the MCSP oracle instead of the SAT oracle. See [IKV18] for further discussion.

With this universal distinguisher in hand, we immediately get Theorem 4. We then obtain the circuit lower bounds in Theorems 1 and 2 by a "win-win" argument on the circuit complexity of MCSP. If $\mathsf{MCSP} \notin \mathsf{P/poly}$, we are done. Otherwise (i.e., if $\mathsf{MCSP} \in \mathsf{P/poly}$) security against $\mathsf{P/poly}$ implies security against $\mathsf{FBPP}^{\mathsf{MCSP}}$ and hence, by our universal distinguisher result above, is equivalent to statistical security for our IO. Then we leverage this very secure IO to get into $\mathsf{Algorithmica}$ where NP is "easy" by extending some ideas from [GR14; Vol23]. The latter leads to certain "collapses" of high complexity classes (such as $\mathsf{NEXP}^{\mathsf{NP}}$), which are known to contain languages outside $\mathsf{P/poly}$, to smaller complexity classes (such as NEXP). Hence we get circuit lower bounds for these smaller complexity classes, as required. Theorem 6 is proved using similar ideas.

1.3 Relation to Previous Work

The results in [Gol90; All+06] imply the following: For any two samplable distribution ensembles $\{A_n\}, \{B_n\}$, we have that $\{A_n\}, \{B_n\}$ are statistically indistinguishable if and only if they are indistinguishable by $\mathsf{FBPP}^{\mathsf{MCSP}}$ algorithms. While this result says that statistical indistinguishability and $\mathsf{FBPP}^{\mathsf{MCSP}}$-computable indistinguishability are the same for efficiently *uniformly* computable distribution ensembles, we need a stronger result applicable also to efficiently *nonuniformly* computable distributions. That is, we need a *universal* distinguisher that will distinguish any two distributions given by sampler circuits, with the distinguishing advantage close to the statistical distance between these distributions.

In [NR06], Naor and Rothblum used similar techniques to prove a similar result, yet with **different** quantifier order: for any *uniformly* computable distribution ensembles there exist a $\mathsf{FBPP}^{\mathsf{MCSP}}$-computable distinguisher with the distinguishing advantage close to the statistical distance between these distributions. Yet, by using the MCSP oracle as a *universal* inverter, one can "extract" a universal distinguisher from their proofs. For completeness we include a self-contained proof in the universal setting.

In [GR14], Goldwasser and Rothblum showed that the existence of statistically-secure obfuscators IO implies that $\mathsf{NP} \subseteq \mathsf{coAM}$, which in turn results in a collapse of the polynomial hierarchy by [BHZ87]. In particular, their idea was to solve SAT in SZK[6]. This is done by leveraging the IO to reduce SAT to *Statistical Difference* (SD) - the standard SZK-complete promise problem of [SV03].

[6] The class of decision problems for which a "yes" answer can be verified by a statistical zero-knowledge proof protocol.

The result then follows from [For89; AH91] where it was shown that $\mathsf{SZK} \subseteq \mathsf{AM} \cap \mathsf{coAM}$. In [Vol23] a simplified and quantified proof of the result was presented. We use some of these ideas as a part our "win-win" argument (see Lemma 5.1 for more details).

Finally, it follows from the definition that the existence of non-uniform one-way functions (i.e. secure against $\mathsf{P/poly}$) already implies very strong circuit lower bounds. Namely, $\mathsf{NP} \not\subseteq \mathsf{P/poly}$. However, this approach cannot be used to derive lower bounds from the existence of an IO since the very same lower bound is already required in order to obtain a one-way function from an IO! In other words, given an IO, one-way functions exist iff $\mathsf{NP} \not\subseteq \mathsf{P/poly}$. Our results allows to obtain a weaker, but still strong circuit lower bound $\mathsf{NEXP} \not\subseteq \mathsf{P/poly}$ from the existence of an IO, thus avoiding this circular reference.

The rest of the paper. The necessary background is given in Section 2. Our main technical contribution (a universal distinguisher) is given in Section 3. In Section 4, we give a simple proof that $\mathsf{SZK} \subseteq \mathsf{BPP}^{\mathsf{MCSP}}$ [AD17]. We give some consequences for MCSP from IO assumptions (including Theorems 5 and 6) in Section 5, and those for IO from MCSP assumptions (including Theorem 4) in Section 6. We prove Theorems 1 and 2 in Section 7. In Section 8, we prove that even an imperfect IO cannot exist in $\mathsf{Heuristica}$. We conclude with some open questions in Section 9. Some auxiliary results are stated in the appendix.

2 Preliminaries

2.1 Definitions

A function $\mathrm{negl}(n)$ is *negligible* if for any $k \in \mathbb{N}$ there exists $n_k \in \mathbb{N}$ such that, for all $n > n_k$, $\mathrm{negl}(n) < 1/n^k$.

Definition 2.1 (Statistical Distance)**.** *Let X_0 and X_1 be two random variables taking values in some finite universe $\mathcal{U}$. The* Statistical Distance *between X_0 and X_1 is defined as*

$$\Delta(X_0, X_1) \triangleq \max_{A\colon \mathcal{U}\to\{0,1\}} \left\{\mathbf{Pr}_{u\sim X_0}[A(u)=1] - \mathbf{Pr}_{u\sim X_1}[A(u)=1]\right\},$$

where $A : \mathcal{U} \to \{0,1\}$ is an arbitrary statistical test (distinguisher).[7] *Another equivalent definition is that*

$$\Delta(X_0, X_1) = (1/2) \cdot \sum_{u\in\mathcal{U}} \left|\mathbf{Pr}_{X_0}[X_0 = u] - \mathbf{Pr}_{X_1}[X_1 = u]\right|.$$

*We say that X_0 and X_1 are δ-*close, *if $\Delta(X_0, X_1) \leq \delta$.*

Definition 2.2 (Indistinguishability Obfuscator [Bar+12; Kom+14; BBF16])**.** *We say that a randomized procedure $\mathsf{IO}(C; r)$ (with randomness r) is an* Indistinguishability Obfuscator *for a circuit class $\mathcal{C}$ with the following:*

1. ***(Perfect/Imperfect) Correctness:*** *IO is ε-*imperfect *if for every circuit $C \in \mathcal{C}$:*

$$\mathbf{Pr}_r[C \equiv \mathsf{IO}(C; r)] \geq 1 - \varepsilon(|C|).$$

If $\varepsilon = 0$, then we say that IO is perfect.

[7] Note that the maximum is attained by the statistical test A such that $A(u) = 1 \iff \mathbf{Pr}[X_0 = u] \geq \mathbf{Pr}[X_1 = u]$.

2. ***Polynomial slowdown:*** *There are $a, k \in \mathbb{N}$ such that, for every circuit $C \in \mathcal{C}$ and every r,*

$$|\mathsf{IO}(C;r)| \leq a \cdot |C|^k .$$

3. ***Security:***

   (a) ***Statistical:*** IO *is* statistically $(1-\delta)$-secure *if for all pairs of circuits $C_1, C_2 \in \mathcal{C}$ such that $C_1 \equiv C_2$ and $|C_1| = |C_2| = s$, we have*

   $$\Delta(\mathsf{IO}(C_1;r), \mathsf{IO}(C_2;r')) \leq \delta(s),$$

   where $\mathsf{IO}(C;r)$ is a distribution over the outputs of $\mathsf{IO}(C;r)$ for random r. We say that IO *is* statistically secure, *if $\delta(s)$ is a negligible function.*

   (b) ***Computational:*** *Let $\mathcal{A}$ be a class of (randomized) algorithms. We say that IO is $(1-\delta)$-*secure against $\mathcal{A}$, *if for every algorithm $A \in \mathcal{A}$, for all pairs of sufficiently large circuits $C_1, C_2 \in \mathcal{C}$ such that $C_1 \equiv C_2$ and $|C_1| = |C_2| = s$, we have*

   $$|\mathbf{Pr}_{r,A}[A(\mathsf{IO}(C_1;r)) = 1] - \mathbf{Pr}_{r,A}[A(\mathsf{IO}(C_2;r)) = 1]| \leq \delta(s),$$

   where the probabilities are over the internal randomness r of IO as well as over possible internal randomness of A. If $\delta(s)$ is negligible, we say that IO is secure against $\mathcal{A}$. *We say that IO is* computationally secure *if it is secure against the class FBPP.*

Remark 2.3 (Efficiency of IO)**.** *By default, we assume $\mathsf{IO}(C;r)$ is computable by a randomized polynomial-time algorithm with internal randomness r. We consider IO computable in other complexity classes, e.g., $\mathsf{FBPP}^{\mathsf{MCSP}}$. In such a case, we shall explicitly say that an IO is $\mathsf{FBPP}^{\mathsf{MCSP}}$-computable.*

Remark 2.4. *Some definitions in the literature also contain a security parameter. In the above definition it is incorporated in the circuit size. Any reasonable encoding scheme for Boolean circuits allows to represent a circuit of size s as a circuit of larger size.*

We will need to following definition and result for our proofs.

Definition 2.5 (Statistical Difference [SV03])**.** *Let $\alpha(n) : \mathbb{N} \to \mathbb{N}$ and $\beta(n) : \mathbb{N} \to \mathbb{N}$ be computable functions, such that $\alpha(n) > \beta(n)$. Then $\mathsf{SD}^{(\alpha(n)\,,\,\beta(n))}$ is promise problem defined as $\mathsf{SD}^{(\alpha(n)\,,\,\beta(n))} \triangleq (\mathsf{SD}_{\text{YES}}^{(\alpha(n)\,,\,\beta(n))}, \mathsf{SD}_{\text{NO}}^{(\alpha(n)\,,\,\beta(n))})$, where*

$$\mathsf{SD}_{\text{YES}}^{(\alpha(n)\,,\,\beta(n))} = \{(C_0, C_1) \mid \Delta(C_0, C_1) \geq \alpha(n)\}, \;\; \mathsf{SD}_{\text{NO}}^{(\alpha(n)\,,\,\beta(n))} = \{(C_0, C_1) \mid \Delta(C_0, C_1) \leq \beta(n)\}.$$

Here, C_0 and C_1 are Boolean circuits $C_0, C_1 : \{0,1\}^n \to \{0,1\}^m$ of size $\mathsf{poly}(n)$ that are samplers for some distributions D_0 and D_1, respectively.
For the standard parameters, we define $\mathsf{SD} \triangleq \mathsf{SD}^{(2/3\,,\,1/3)}$.
For an oracle O, we define the relativized version of the problem $\mathsf{SD}^{O\;(\alpha(n)\,,\,\beta(n))}$ as above, when C_0 and C_1 are O-oracle circuits.

Lemma 2.6 ([SV03])**.** *Suppose $\alpha(n)^2 - \beta(n) \geq 1/\mathsf{poly}(n)$. Then for any oracle O, the problem $\mathsf{SD}^{O\;(\alpha(n)\,,\,\beta(n))}$ is SZK^O-complete. In particular, SD is SZK-complete.*

2.2 Useful Lemmas

Let $\mathsf{FBPP}^{\mathsf{MCSP}}$ denote the class of randomized polynomial-time algorithms with MCSP oracle.

Lemma 2.7 (implicit in [IKV18])**.** *If there exists an* IO $(1-\delta)$*-secure against* $\mathsf{FBPP}^{\mathsf{MCSP}}$*, for some* $\delta \leq 1 - 1/n^{\ell}$ *for a constant* $\ell > 0$*, then* $\mathsf{NP} \subseteq \mathsf{ZPP}^{\mathsf{MCSP}}$ *and hence* $\mathsf{ZPP}^{\mathsf{NP}} = \mathsf{ZPP}^{\mathsf{MCSP}}$*.*

Lemma 2.8 ([Vol23])**.** *If there exists an* IO *statistically* $(1-\delta)$*-secure, for some* $\delta < 1$*, then* $\mathsf{NP} \subseteq \mathsf{coNP}$ *and hence* $\mathsf{PH} = \mathsf{NP} \cap \mathsf{coNP}$*.*

Lemma 2.9 ([Vol23])**.** *Let* IO *be an* ε*-imperfect obfuscator and let* C_1, C_2 *be such that* $C_1 \not\equiv C_2$*. Then* $\Delta(\mathsf{IO}(C_1; r), \mathsf{IO}(C_2; r')) \geq 1 - 2\varepsilon$*, over the internal randomness* r, r' *of the* IO*.*

Lemma 2.10 ([Kan82])**.** *For any* $k \in \mathbb{N} : \mathsf{NP}^{\mathsf{NP}} \not\subseteq \mathsf{SIZE}[n^k]$*. In addition,* $\mathsf{NEXP}^{\mathsf{NP}} \not\subseteq \mathsf{P/poly}$*.*

Lemma 2.11 ([Bsh+96; KW98])**.** *If* $\mathsf{SAT} \in \mathsf{P/poly}$*, then* $\mathsf{PH} = \mathsf{ZPP}^{\mathsf{SAT}}$*, and polynomial-size circuits for* SAT *can be constructed in* $\mathsf{ZPP}^{\mathsf{SAT}}$*.*

Lemma 2.12 ([IKV18])**.** *If* $\mathsf{MCSP} \in \mathsf{P/poly}$*, then* $\mathsf{BPP}^{\mathsf{MCSP}} = \mathsf{ZPP}^{\mathsf{MCSP}}$*.*

We require the following result of [All+06] that allows to find preimages of functions computable in polynomial time.

Lemma 2.13 ([All+06])**.** *Let* $f_y(x) = f(y, x)$ *be a function computable uniformly in time polynomial in* $|x|$*. There exists a polynomial-time probabilistic oracle Turing machine* M *such that for any* $n, K \in \mathbb{N}$ *and any* y*:*

$$\mathbf{Pr}_{|x|=n,r}\left[f_y\left(M^{\mathsf{MCSP}}(1^K, y, f_y(x), r)\right) = f_y(x)\right] \geq 1/K,$$

where $x \in \{0,1\}^n$ *is chosen uniformly at random and* r *denotes the internal randomness of* M*.*

We generalize this result to get a *distributional* inverter for any candidate distributional one-way function in the sense of [IL89]. Roughly speaking, such a distributional inverter finds uniformly random preimages of a given polynomial-time computable function. More precisely, we have the following.

Lemma 2.14. *Let* $f_y(x) = f(y, x)$ *be a function computable uniformly in time polynomial in* $|x|$*. There exists a polynomial-time probabilistic oracle Turing machine* M *such that, for any* $n, K \in \mathbb{N}$ *and any* y*, the following two distributions*

$$(x,\ f_y(x)) \qquad \text{and} \qquad \left(M^{\mathsf{MCSP}}(1^K, y, f_y(x), r),\ f_y(x)\right),$$

for $x \in \{0,1\}^n$ *chosen uniformly at random, and* r *the internal uniform randomness of* M*, are at most* $(1/K)$*-far in statistical distance.*

Proof. We combine Lemma 2.13 with the reduction from [IL89] showing that an inverter for candidate one-way functions can be used to get a distributional inverter for every distributional one-way function candidate $f_y(x)$ computable in polynomial time. □

3 From Computational to Statistical Security

Below we will argue the existence of a universal distinguisher. We will describe an algorithm $\mathcal{D}(C_0, C_1; 1^{1/\gamma})$ in $\mathsf{FBPP}^{\mathsf{MCSP}}$ that, given any pair of circuits C_0 and C_1 that are samplers for some distributions D_0 and D_1, and a parameter $0 < \gamma < 1$, will distinguish D_0 and D_1 with advantage at least $\delta - \gamma$, where δ is the statistical distance between D_0 and D_1.

We will first argue that such a distinguisher for a pair of distributions sampled by circuits C_0 and C_1 can be obtained given oracle access to a distributional inverter (in the sense of Impagliazzo and Luby [IL89]) for a function defined in terms of C_0 and C_1 (see Lemma 3.1 below). Then we appeal to Lemma 2.14 to get a universal distributional inverter.

Lemma 3.1. *There is an oracle* FBPP *algorithm* $\hat{\mathcal{D}}$ *satisfying the following. Let* C_0 *and* C_1 *be two circuits that are samplers for distributions* D_0 *and* D_1 *over some finite universe* $\mathcal{U}$, *and let* δ *be the statistical distance between* D_0 *and* D_1. *Let* $F(b, r)$ *use* r *to sample from* D_b. *Let* A *be a distributional inverter for* F *so that the distributions*

$$((b, r),\, F(b, r)) \qquad \textit{and} \qquad (A(F(b, r)),\, F(b, r))$$

are at most α^2*-close in statistical distance, where* $0 \le \alpha \le \delta/28$. *Then* $\hat{\mathcal{D}}^A(C_0, C_1; 1^{1/\alpha})$ *is a distinguisher for* D_0 *and* D_1 *with advantage at least* $\delta - 14\alpha \ge \delta/2$.

Proof. Let $B(x)$ be the first bit of $A(x)$, and let $Q(x)$ be the probability that $B(x)$ is 0, i.e.,

$$Q(x) = \mathbf{Pr}_A[B(x) = 0],$$

where the probability is over the internal randomness of A. Let K be such that an empirical estimate of K iid $\{0, 1\}$-valued random variables is within α of its expectation with probability $1 - (\alpha/2)$; by the Chernoff bounds, we have that $K = O((\log 1/\alpha)/\alpha^2)$. Let $\tilde{Q}(x; \rho)$ be the random variable where we use randomness ρ to sample from $A(x)$ independently K times and use these to create an empirical estimate of $Q(x)$; so ρ is K times the internal randomness of A. Let $C(x; \rho)$ be the probabilistic Boolean algorithm where we accept x if $\tilde{Q}(x; \rho) \ge 1/2$. We will show that

$$\mathbf{Pr}_{x\sim D_0,\rho}\left[C(x; \rho) = 1\right] - \mathbf{Pr}_{x\sim D_1,\rho}\left[C(x; \rho) = 1\right] \ge \delta - 14\alpha. \tag{1}$$

Let $p_0(x)$ be the probability of x for D_0, and $p_1(x)$ that for D_1. Note that

$$q(x) = p_0(x)/(p_0(x) + p_1(x))$$

is the conditional probability that $b = 0$ given that $F(b, r) = x$. Then

$$\mathbf{Pr}_{x\sim D_0,\rho}\left[C(x; \rho) = 1\right] - \mathbf{Pr}_{x\sim D_1,\rho}\left[C(x; \rho) = 1\right] = \mathbf{Exp}_\rho\left[\sum_{x:\ \tilde{Q}(x;\rho)\ge 1/2} (p_0(x) - p_1(x))\right], \tag{2}$$

and

$$\delta = \sum_{x:\ q(x)\ge 1/2} (p_0(x) - p_1(x)). \tag{3}$$

Note that if, for "typical" randomness ρ used by $\tilde{Q}$, we had for all $x \in \mathcal{U}$ that $\tilde{Q}(x; \rho) \ge 1/2 \Leftrightarrow q(x) \ge 1/2$, then the right-hand sides of (2) and (3) would be identical (for that randomness ρ of

$\tilde{Q}$), and we would get our goal of (1) minus the error term for "atypical" randomness of $\tilde{Q}$. We formalize this argument next.

For given internal randomness ρ of $\tilde{Q}$, let the error set $E = E(\rho)$ be the set of those $x \in \mathcal{U}$ so that exactly one of $\tilde{Q}(x;\rho)$ and $q(x)$ is at least $1/2$, i.e.,

$$E(\rho) = \{x \in \mathcal{U} \mid \tilde{Q}(x;\rho) \geq 1/2 \not\Leftrightarrow q(x) \geq 1/2\}.$$

Then

$$\begin{aligned}
&\mathbf{Pr}_{x\sim D_0,\rho}\left[C(x;\rho)=1\right] - \mathbf{Pr}_{x\sim D_1,\rho}\left[C(x;\rho)=1\right] \\
&= \mathbf{Exp}_\rho\left[\sum_{x:\ \tilde{Q}(x;\rho)\geq 1/2} (p_0(x)-p_1(x))\right] \\
&= \mathbf{Exp}_\rho\left[\sum_{x\notin E(\rho):\ \tilde{Q}(x;\rho)\geq 1/2} (p_0(x)-p_1(x)) + \sum_{x\in E(\rho):\ \tilde{Q}(x;\rho)\geq 1/2} (p_0(x)-p_1(x))\right] \\
&= \mathbf{Exp}_\rho\left[\sum_{x\notin E(\rho):\ q(x)\geq 1/2} (p_0(x)-p_1(x)) + \sum_{x\in E(\rho):\ q(x)< 1/2} (p_0(x)-p_1(x))\right] \\
&= \mathbf{Exp}_\rho\left[\sum_{x:\ q(x)\geq 1/2} (p_0(x)-p_1(x)) + \sum_{x\in E(\rho):\ q(x)< 1/2} (p_0(x)-p_1(x)) - \sum_{x\in E(\rho):\ q(x)\geq 1/2} (p_0(x)-p_1(x))\right] \\
&\geq \delta - \mathbf{Exp}_\rho\left[\sum_{x\in E(\rho)} |p_0(x)-p_1(x)|\right],
\end{aligned}$$

where we used (3) to get the last line.

We bound the sum under the expectation in the last line above by looking at three sets whose union contains $E = E(\rho)$:

$$\begin{aligned}
E_1(\rho) &= \{x \mid |\tilde{Q}(x;\rho) - Q(x)| \geq \alpha\}, \\
E_2 &= \{x \mid |Q(x) - q(x)| \geq 2\alpha, \\
E_3 &= \{x \mid |q(x) - 1/2| \leq 3\alpha\}.
\end{aligned}$$

Claim 3.2. *For every ρ, $E(\rho) \subseteq E_1(\rho) \cup E_2 \cup E_3$.*

Proof of Claim 3.2. If $\tilde{Q}(x;\rho) \geq 1/2$, and $x \notin (E_1 \cup E_2)$, then $q(x) > 1/2 - 3\alpha$. So either $q(x) \geq 1/2$, or $x \in E_3$. Similar reasoning applies if $\tilde{Q}(x;\rho) < 1/2$. So these three sets cover E. □

We bound the sum for E_1 just by using Chernoff bounds, the sum for E_2 by the statistical distinguishability of our distributional inverter A, and the sum for E_3 using the fact that having q close to $1/2$ means $p_0(x)$ and $p_1(x)$ are relatively close. For $E_1(\rho)$ and E_2, we will actually upperbound the summation of $p_0(x) + p_1(x)$, over x from the respective set.

Claim 3.3. $\mathbf{Exp}_\rho\left[\sum_{x\in E_1(\rho)}(p_0(x)+p_1(x))\right] \leq \alpha$.

Proof of Claim 3.3. By linearity of expectation, it suffices to upperbound

$$\mathbf{Exp}_\rho\left[\sum_{x\in E_1(\rho)} p_0(x)\right] + \mathbf{Exp}_\rho\left[\sum_{x\in E_1(\rho)} p_1(x)\right].$$

The first expectation can be thought of as the probability that, if we sample x from D_0, and then perform the empirical estimate (using randomness ρ), that we are off by at least α. The second expectation is the same but for D_1. By the Chernoff bounds (our choice of K), each probability is at most $\alpha/2$. □

Claim 3.4. $\sum_{x\in E_2}(p_0(x)+p_1(x)) \leq \alpha$.

Proof of Claim 3.4. We use the accuracy of the inverter A. The distinguishing probability between $(A(F(b,r)),\, F(b,r))$ and $((b,r),\, F(b,r))$ is at least that between any distributions computable from these. So in particular, the statistical distance between $(B(x),\, x)$ and $(b,\, x)$, for $x = F(b,r)$, is at most α^2. Using the fact that the statistical distance is the half of the ℓ_1-norm of the difference between the distributions, we get

$$\begin{aligned}\alpha^2 &\geq (1/2)\cdot\sum_x (1/2)\cdot(p_0(x)+p_1(x))\cdot(|q(x)-Q(x)| + |1-Q(x)-(1-q(x))|)\\ &= (1/2)\cdot\sum_x (p_0(x)+p_1(x))\cdot|q(x)-Q(x)|,\end{aligned}$$

Since for all x in E_2, $|q(x)-Q(x)| \geq 2\alpha$, and restricting to $x\in E_2$ only reduces the sum in the last line, we have

$$\alpha^2 \geq (1/2)\cdot\sum_{x\in E_2}(p_0(x)+p_1(x))(2\alpha),$$

or $\sum_{x\in E_2}(p_0(x)+p_1(x)) \leq \alpha$, as required. □

Claim 3.5. $\sum_{x\in E_3}|p_0(x)-p_1(x)| \leq 12\alpha$.

Proof of Claim 3.5. If $x\in E_3$ then

$$\left|\frac{p_0(x)}{p_0(x)+p_1(x)} - \frac{1}{2}\right| \leq 3\alpha.$$

Multiplying through by $2(p_0(x)+p_1(x))$,

$$|p_0(x)-p_1(x)| \leq 6\alpha(p_0(x)+p_1(x)).$$

Thus,

$$\begin{aligned}\sum_{x\in E_3}|p_0(x)-p_1(x)| &\leq \sum_{x\in E_3}6\alpha(p_0(x)+p_1(x))\\ &\leq 12\alpha,\end{aligned}$$

as required. □

Combining Claims 3.3–3.5, we get that the advantage of our probabilistic circuit C at distinguishing D_0 and D_1 is at least $\delta - 14\alpha$, as required. Given oracle access to A, our algorithm $\hat{\mathcal{D}}^A(C_0, C_1; 1^{1/\alpha})$ will construct such a circuit C in time polynomial in $1/\alpha$. □

Corollary 3.6. *There is an* $\mathsf{FBPP}^{\mathsf{MCSP}}$ *algorithm* $\mathcal{D}$ *satisfying the following. Given any pair of circuits* C_0 *and* C_1 *that are samplers for some distributions* D_0 *and* D_1*, and given a parameter* K *in unary, the algorithm* $\mathcal{D}(C_0, C_1; 1^K)$ *will distinguish* D_0 *and* D_1 *with advantage at least* $\delta - 1/K$*, where* δ *is the statistical distance between* D_0 *and* D_1*.*

Proof. Use Lemma 2.14 to get an $\mathsf{FBPP}^{\mathsf{MCSP}}$-computable universal distributional inverter that achieves statistical distance α^2 for $\alpha = 1/(14K)$. Define the algorithm $\mathcal{D}$ as follows. For given input circuits C_0 and C_1, run the oracle algorithm $\hat{\mathcal{D}}^A(C_0, C_1; 1^{1/\alpha})$ from Lemma 3.1, using our universal inverter to get a distributional inverter A needed by $\hat{\mathcal{D}}$. □

Next, we show that we can extend our universal distinguisher for distributions samplable by O-oracle circuits for languages O satisfying certain technical conditions.

Corollary 3.7. *Let* $O \in \mathsf{BPP}^{\mathsf{MCSP}} \cap \mathsf{P/poly}$ *be any language. Then there is an* $\mathsf{FBPP}^{\mathsf{MCSP}}$ *algorithm* $\mathcal{D}$ *that, given any pair of* O*-oracle circuits* C_0 *and* C_1 *that are samplers for some distributions* D_0 *and* D_1*, and given a parameter* K *in unary, the algorithm* $\mathcal{D}(C_0, C_1; 1^K)$ *will distinguish* D_0 *and* D_1 *with advantage at least* $\delta - 1/K$*, where* δ *is the statistical distance between* D_0 *and* D_1*.*

Proof. Use Lemma 2.14 to get an $\mathsf{FBPP}^{\mathsf{MCSP}}$-computable universal distributional inverter that achieves statistical distance α^2 for $\alpha = 1/(14K)$. As in the proof of Lemma 3.1, we use MCSP-oracle circuit samplers for distributions D_0 and D_1 to get a circuit for distributional one-way function candidate F. We then use our universal inverter to get a distributional inverter A needed in Lemma 3.1 for any given input circuits C_0 and C_1. Observe that we can invert F since F is computable by a small O-oracle circuit (given the O-oracle circuits for sampling D_0 and D_1), and hence F is also computable by a circuit of polynomial size with *no* oracle gates (since by assumption $O \in \mathsf{P/poly}$). The correctness proof of the inverting algorithm relies on the fact that a small circuit for F *exists*. Yet, the inverting algorithm for F does not need to know a small circuit for F; it just must be able to evaluate F efficiently, given a small description of F. Using the encoding of an O-oracle circuit for F works since the inverting algorithm can evaluate the circuit with probability close to 1, given access to the MCSP oracle (since $O \in \mathsf{BPP}^{\mathsf{MCSP}}$). □

Remark 3.8. *We note that a universal distinguisher as in Corollary 3.6 is fairly easy to construct in* $\mathsf{FBPP}^{\mathsf{SAT}}$ *(using the well-known fact that approximate counting can be done in* $\mathsf{FBPP}^{\mathsf{NP}}$ *[JVV86]); see Theorem A.2 in Section A of the appendix. Thus, Corollary 3.6 is another example of a computational task that can still be performed with the* MCSP *oracle instead of the* SAT *oracle.*

4 Another Proof that $\mathsf{SZK} \subseteq \mathsf{BPP}^{\mathsf{MCSP}}$

Corollary 3.6 can be used to give another proof of the following result by Allender and Das [AD17].

Theorem 4.1 ([AD17])**.** $\mathsf{SZK} \subseteq \mathsf{BPP}^{\mathsf{MCSP}}$.

Proof. Recall the standard SZK-complete promise problem Statistical Difference (SD) (see Definition 2.5): Given a pair of circuits (C_0, C_1) that are samplers for the distributions D_0 and D_1

such that either D_0 and D_1 have the statistical distance less than $1/3$, or they have the statistical distance greater than $2/3$, decide which is the case.

By Corollary 3.6, we get an $\mathsf{FBPP}^{\mathsf{MCSP}}$ universal distinguisher $\mathcal{D}$. Consider the distinguisher $B = \mathcal{D}(C_0, C_1; 1^{10})$. Let δ be the statistical distance between D_0 and D_1. Note that in case $\delta < 1/3$, the algorithm B (and, in fact, any algorithm) has distinguishing advantage less than $1/3$, whereas for $\delta > 2/3$, B has advantage at least $(2/3) - (1/10) = 17/30 > 1/3$. Using random sampling and the Chernoff bounds, we can estimate in $\mathsf{FBPP}^{\mathsf{MCSP}}$ the advantage of our algorithm B at distinguishing between D_0 and D_1, with high probability and sufficient accuracy. The theorem follows. □

Remark 4.2. *Note that the* $\mathsf{BPP}^{\mathsf{MCSP}}$ *algorithm for* SZK *in the proof of Theorem 4.1 works for any version of the Statistical Difference problem with a non-negligible gap between the yes- and no-instances, not just for the* $1/3$ *vs.* $2/3$ *gap.*

Next, we extend the result above to the relativized version of the problem SD^O (see Definition 2.5) for any $O \in \mathsf{BPP}^{\mathsf{MCSP}} \cap \mathsf{P/poly}$.

Theorem 4.3. *Let* $O \in \mathsf{BPP}^{\mathsf{MCSP}} \cap \mathsf{P/poly}$ *be any language. Then for any* $\alpha(n)$ *and* $\beta(n)$ *such that* $\alpha(n) \geq \beta(n) + n^{-\ell}$*, for some* $\ell > 0$*, we have that:*

$$\mathsf{SD}^{O\ (\alpha(n)\ ,\ \beta(n))} \in \mathsf{BPP}^{\mathsf{MCSP}}. \tag{4}$$

In particular, if $\mathsf{MCSP} \in \mathsf{P/poly}$*, then*

$$\mathsf{SZK}^{\mathsf{MCSP}} \subseteq \mathsf{BPP}^{\mathsf{MCSP}}. \tag{5}$$

Proof. To prove (4), we proceed exactly as in the proof of Theorem 4.1 above, except using Corollary 3.7 instead of Corollary 3.6, and using the observation in Remark 4.2. To prove (5), we use (4) for $O = \mathsf{MCSP}$ and the fact that $\mathsf{SD}^{\mathsf{MCSP}\ (2/3\ ,\ 1/3)}$ is $\mathsf{SZK}^{\mathsf{MCSP}}$-complete (by Lemma 2.6). □

5 Implications for Circuit Minimization from Obfuscation

The following lemma provides some consequences of the existence of an imperfect, computationally-secure IO, with an appropriate range of parameters. Among other things, the proof uses some ideas from [GR14] and [IKV18].

Lemma 5.1. *Let* $\Gamma \in \{\mathsf{FBPP}, \mathsf{P/poly}\}$*. Suppose there exist an* ε*-imperfect* $\mathsf{IO}(C; r)$ *that is* $(1-\delta)$*-secure against* Γ*, where* $(1-2\varepsilon)^2 - \delta \geq 2/n^{\ell}$ *for some constant* $\ell > 0$*. If* $\mathsf{MCSP} \in \Gamma$*, then:*

1. $\mathsf{NP} \subseteq \mathsf{SZK}$*,*
2. $\mathsf{PH} = \mathsf{MA} = \mathsf{ZPP}^{\mathsf{MCSP}}$*, and*
3. *There is a* $\mathsf{ZPP}^{\mathsf{MCSP}}$ *algorithm* A *and a constant* $k > 0$*, such that* $A(1^n)$ *outputs an* $O(n^k)$*-size circuit for* SAT *(and for* MCSP*) on* n*-bit inputs.*

Proof.

1. First, observe that since $\mathsf{MCSP} \in \Gamma$, $\mathsf{FBPP}^{\mathsf{MCSP}} \subseteq \Gamma$. Consequently, an IO that is secure against Γ is also secure against $\mathsf{FBPP}^{\mathsf{MCSP}}$. It follows from Corollary 3.6 that this IO is statistically $(1-\delta')$-secure, for $\delta' = \delta + 1/n^{\ell}$ (since we can make the distributional inverter's error α to be smaller than any inverse polynomial of our choice). We now use this IO to reduce SAT to the Statistical Difference problem (see Definition 2.5): Given a SAT instance ϕ, construct some unsatisfiable instance $\bot$ of the same size as ϕ and on the same set of input variables. Consider the distributions

   $$\mathsf{IO}(\phi; r) \text{ and } \mathsf{IO}(\bot; r') \tag{6}$$

   over all random strings r, r'.

   We have two cases:

   - If ϕ is unsatisfiable, then $\phi \equiv \bot$, and by the statistical $(1-\delta')$-security property of our IO, we get that these two distributions in (6) have statistical distance at most δ'.
   - If ϕ is satisfiable, then by Lemma 2.9, the statistical distance between the distributions in (6) is at least $1 - 2\varepsilon$.

   Since $(1-2\varepsilon)^2 \geq \delta + 2n^{-\ell} = \delta' + n^{-\ell}$, by Lemma 2.6, the resulting instance of the SD problem is SZK-complete, and so $\mathsf{NP} \subseteq \mathsf{SZK}$.[8]

2. By [For89; AH91], we have $\mathsf{SZK} \subseteq \mathsf{AM} \cap \mathsf{coAM}$. By [BHZ87], since $\mathsf{NP} \subseteq \mathsf{SZK} \subseteq \mathsf{coAM}$, it follows that

   $$\mathsf{PH} = \mathsf{AM}. \tag{7}$$

   Next, by Theorem 4.1, $\mathsf{SZK} \subseteq \mathsf{BPP}^{\mathsf{MCSP}}$. By Lemma 2.12, $\mathsf{BPP}^{\mathsf{MCSP}} = \mathsf{ZPP}^{\mathsf{MCSP}}$. Hence, we get that

   $$\mathsf{NP} \subseteq \mathsf{SZK} \subseteq \mathsf{ZPP}^{\mathsf{MCSP}}. \tag{8}$$

   As $\mathsf{MCSP} \in \mathsf{P/poly}$, we also get from (8) that

   $$\mathsf{NP} \subseteq \mathsf{P/poly}. \tag{9}$$

   By [Arv+95], (9) implies that $\mathsf{AM} = \mathsf{MA}$. So by (7), we conclude that

   $$\mathsf{PH} = \mathsf{MA}.$$

   Finally, (9) also implies $\mathsf{PH} = \mathsf{ZPP}^{\mathsf{NP}}$ by Lemma 2.11. Hence, by (8), we get that

   $$\mathsf{PH} = \mathsf{ZPP}^{\mathsf{NP}} \subseteq \mathsf{ZPP}^{\mathsf{ZPP}^{\mathsf{MCSP}}} = \mathsf{ZPP}^{\mathsf{MCSP}}.$$

3. By Lemma 2.11, if $\mathsf{SAT} \in \mathsf{P/poly}$, then polynomial-size circuits for SAT can be found by a $\mathsf{ZPP}^{\mathsf{NP}}$ algorithm. By (9), we get that polynomial-size circuits for SAT can be found by a $\mathsf{ZPP}^{\mathsf{ZPP}^{\mathsf{MCSP}}}$ algorithm, which can be simulated by a $\mathsf{ZPP}^{\mathsf{MCSP}}$ algorithm. As $\mathsf{MCSP} \in \mathsf{NP}$ and SAT is NP-complete, a polynomial-size circuits for SAT can be used to construct a polynomial-size circuits for MCSP as well.

[8] Note that this reduction to SZK actually allows one to solve not just SAT but an *equivalence* problem for any class of circuits that an IO can obfuscate. Thus, to conclude that $\mathsf{NP} \subseteq \mathsf{SZK}$, it suffices to pick any coNP-hard circuit equivalence problem for the class of circuits where SAT may be easy. For example, one can take the problem of testing equivalence of depth-3 *monotone* formulas, known to be coNP-complete [EG95] (see Section B in Appendix).

□

Items (2) and (3) in the lemma should be contrasted with the result of [Bsh+96; KW98] that $\mathsf{SAT} \in \mathsf{P/poly}$ implies both that polynomial-size circuits for SAT can be constructed by a $\mathsf{ZPP}^{\mathsf{SAT}}$ algorithm, and that $\mathsf{PH} = \mathsf{ZPP}^{\mathsf{SAT}}$. Under an additional assumption that a $\mathsf{P/poly}$-secure imperfect IO exists, we get similar implications for MCSP instead of SAT.

The following corollary strengthens a result of [IKV18] to the imperfect setting.

Corollary 5.2 (Theorem 5 re-stated)**.** *Let $\Gamma \in \{\mathsf{ZPP}, \mathsf{BPP}\}$. Suppose there is an ε-imperfect IO that is $(1-\delta)$-secure against FBPP, where $(1-2\varepsilon)^2 \geq \delta + 2/n^\ell$ for some constant $\ell > 0$. Then $\mathsf{MCSP} \in \Gamma$ iff $\mathsf{NP} \subseteq \Gamma$.*

For the case of perfect IO, we get the following.

Theorem 5.3 (Theorem 6 re-stated)**.** *Suppose there is a perfect IO that is $(1-\delta)$-secure against FBPP, where $\delta \leq 1 - 2/n^\ell$ for some constant $\ell > 0$. If $\mathsf{MCSP} \in \mathsf{BPP}$, then $\mathsf{NP} = \mathsf{ZPP}$.*

Proof. Since $\mathsf{MCSP} \in \mathsf{BPP}$, computational $(1-\delta)$-security implies $(1-\delta)$-security against $\mathsf{FBPP}^{\mathsf{MCSP}}$. It follows by Corollary 3.6 that this IO is statistically $(1-\delta')$-secure, for $\delta' = \delta + 1/n^\ell \leq 1 - 1/n^\ell$.

By Lemma 2.8, $\mathsf{PH} = \mathsf{NP} = \mathsf{coNP}$. By Lemma 2.7, $\mathsf{PH} = \mathsf{NP} = \mathsf{ZPP}^{\mathsf{MCSP}} \subseteq \mathsf{BPP}$. But $\mathsf{NP} \subseteq \mathsf{BPP}$ implies that $\mathsf{NP} = \mathsf{RP}$. Since $\mathsf{coNP} = \mathsf{NP}$, we get $\mathsf{NP} = \mathsf{ZPP}$. □

6 Implications for Obfuscation from Circuit Minimization

Theorem 6.1. *Suppose $\mathsf{MCSP} \in \mathsf{P/poly}$. There is an $\mathsf{FZPP}^{\mathsf{MCSP}}$-computable perfect IO that is statistically secure if and only if there is an $\mathsf{FBPP}^{\mathsf{MCSP}}$-computable ε-imperfect IO that is $(1-\delta)$ secure against $\mathsf{P/poly}$, for any $0 \leq \varepsilon, \delta \leq 1$ such that $1 - 2\varepsilon \geq \delta + 2/n^\ell$.*

Proof. The interesting direction is from the right to the left. Since $\mathsf{MCSP} \in \mathsf{P/poly}$, $(1-\delta)$-security against $\mathsf{P/poly}$ implies, by Corollary 3.6, statistical $(1-\delta')$-security, for $\delta' = \delta + n^{-\ell}$.

Claim 6.2. *If $\mathsf{MCSP} \in \mathsf{P/poly}$ and there is an $\mathsf{FBPP}^{\mathsf{MCSP}}$-computable ε-imperfect IO that is statistically $(1-\delta')$-secure for $1 - 2\varepsilon \geq \delta' + n^{-\ell}$, then $\mathsf{SAT} \in \mathsf{ZPP}^{\mathsf{MCSP}}$.*

Proof of Claim 6.2. Given an instance ϕ of SAT, let $\perp$ be an unsatisfiable formula of the same size as ϕ (over the same variables). Consider the two distributions $\mathsf{IO}(\phi; r)$ and $\mathsf{IO}(\perp; r')$ over random r, r'. If $\phi \equiv \perp$, the two distributions are at most statistical distance δ' apart; if ϕ is in SAT, then the two distributions have the statistical distance at least $1 - 2\varepsilon$.

Each distribution is samplable using a polynomial-size MCSP-oracle circuit, which we can obtain from our IO algorithm. Thus, we get an $\mathsf{FP}^{\mathsf{MCSP}}$-reduction from coSAT to $\mathsf{SD}^{\mathsf{MCSP}\,(1-2\varepsilon\,,\,\delta')}$. Since $\delta' + n^{-\ell} \leq 1 - 2\varepsilon$, we conclude by Theorem 4.3 that $\mathsf{SAT} \in \mathsf{BPP}^{\mathsf{MCSP}}$. By Lemma 2.12, $\mathsf{BPP}^{\mathsf{MCSP}} = \mathsf{ZPP}^{\mathsf{MCSP}}$, concluding the proof. □

Since $\mathsf{SAT} \in \mathsf{ZPP}^{\mathsf{MCSP}} \subseteq \mathsf{P/poly}$, we get by Lemma 2.11 that $\mathsf{PH} = \mathsf{ZPP}^{\mathsf{MCSP}}$. Given a circuit C, we can find the lexicographically smallest equivalent circuit D (of size at most that of C) in $\mathsf{FP}^{\mathsf{PH}} \subseteq \mathsf{FZPP}^{\mathsf{MCSP}}$. This gives us a perfect $\mathsf{IO}(C; r)$ that is statistically secure.[9] □

[9] Technically, this $\mathsf{IO}(C; r)$ outputs either a smallest equivalent circuit D, or, with a tiny probability, the "don't know" answer. We can modify it to output the input circuit C in the latter case, getting perfect correctness, and only slightly decreasing statistical security.

Along the same lines:

Corollary 6.3. *Suppose* $\mathsf{MCSP} \in \mathsf{BPP}$*. There is an* ε*-imperfect* IO *with statistical security if and only if there is an* ε*-imperfect* IO *with computational* $(1-\delta)$*-security where* $1-2\varepsilon \geq \delta + 2/n^{\ell}$*. (Assuming* $\mathsf{MCSP} \in \mathsf{ZPP}$*, you get a similar equivalence but for a perfect* IO *with statistical security.)*

Proof sketch. The interesting direction is from the right to the left. We first argue as in the proof of Theorem 6.1 to conclude that $\mathsf{SAT} \in \mathsf{ZPP}^{\mathsf{MCSP}}$. Since $\mathsf{MCSP} \in \mathsf{BPP}$, we get that $\mathsf{NP} \subseteq \mathsf{BPP}$, and hence, $\mathsf{PH} = \mathsf{BPP}$. So, given an input circuit C, we can find the lexicographically smallest equivalent circuit D (of size at most that of C), using an $\mathsf{FP}^{\mathsf{PH}} = \mathsf{FBPP}$ algorithm. This algorithm is a (negligibly) imperfect IO with statistical security. (In case of $\mathsf{MCSP} \in \mathsf{ZPP}$, we argue in a similar way, getting that $\mathsf{PH} = \mathsf{ZPP}$, and so a canonical circuit D for a given input circuit C can be found in FZPP.) □

7 Circuit Lower Bounds from Obfuscation

Here we prove Theorems 1 and 2, re-stated below.

Theorem 7.1 (Theorem 1 re-stated)**.** *Suppose there exist a perfect* IO $(1-\delta)$*-secure against* $\mathsf{P/poly}$*, where* $\delta \leq 1 - 2/n^{\ell}$ *for some* $\ell > 0$*. Then:*

1. $\mathsf{NEXP} \cap \mathsf{ZPEXP}^{\mathsf{MCSP}} \not\subseteq \mathsf{P/poly}$.
2. *For all* $k \in \mathbb{N}$*,* $\mathsf{NP} \cap \mathsf{ZPP}^{\mathsf{MCSP}} \not\subseteq \mathsf{SIZE}[n^k]$.

Proof. The proof of all items goes by a "win-win" argument. Suppose $\mathsf{MCSP} \notin \mathsf{P/poly}$. Then both claims follow immediately since $\mathsf{MCSP} \in \mathsf{NP}$.

Now suppose $\mathsf{MCSP} \in \mathsf{P/poly}$. Then randomized polynomial-time algorithms with MCSP oracle can be simulated by polynomial-size circuits. Consequently, IO is $(1-\delta)$-secure against these algorithms. By Corollary 3.6, this IO is statistically $(1-\delta')$-secure, for $\delta' = \delta + n^{-\ell} \leq 1 - n^{-\ell}$. By Lemmas 2.7 and 2.8, we get that

$$\mathsf{NP}^{\mathsf{NP}} \subseteq \mathsf{PH} = \mathsf{NP} \cap \mathsf{coNP} \subseteq \mathsf{ZPP}^{\mathsf{MCSP}} \subseteq \mathsf{NP}^{\mathsf{NP}}.$$

So, $\mathsf{NP}^{\mathsf{NP}} = \mathsf{NP} \cap \mathsf{coNP} = \mathsf{ZPP}^{\mathsf{MCSP}}$. By padding, $\mathsf{NEXP}^{\mathsf{NP}} = \mathsf{NEXP} \cap \mathsf{coNEXP} = \mathsf{ZPEXP}^{\mathsf{MCSP}}$, and so both claims follow from Lemma 2.10. □

Theorem 7.2 (Theorem 2 re-stated)**.** *Suppose there exist an* ε*-imperfect* IO $(1-\delta)$*-secure against* $\mathsf{P/poly}$*, where* $(1-2\varepsilon)^2 \geq \delta + 2/n^{\ell}$ *for some* $\ell > 0$*. Then for all* $k \in \mathbb{N}$*,* $\mathsf{MA} \cap \mathsf{ZPP}^{\mathsf{MCSP}} \not\subseteq \mathsf{SIZE}[n^k]$*.*

Proof. Again we use a "win-win" argument. If $\mathsf{MCSP} \notin \mathsf{P/poly}$, then the theorem follows. Otherwise, we get by Lemma 5.1 (Item 2) that $\mathsf{PH} = \mathsf{MA} = \mathsf{ZPP}^{\mathsf{MCSP}}$, which is not in $\mathsf{SIZE}[n^k]$ for any fixed $k > 0$ by Lemma 2.10. □

Theorem 7.3 (Theorem 3 re-stated)**.** *Suppose there is a perfect* IO *that is* $(1-\delta)$*-secure against* FBPP*, where* $\delta \leq 1 - 2/n^{\ell}$ *for some constant* $\ell > 0$*. Then* $\mathsf{ZPEXP} \neq \mathsf{BPP}$*.*

Proof of Theorem 3. Suppose for a contradiction that $\mathsf{ZPEXP} = \mathsf{BPP}$. Then, in particular, $\mathsf{MCSP} \in \mathsf{BPP}$. By Theorem 6, $\mathsf{NP} = \mathsf{ZPP}$ and hence

$$\mathsf{ZPEXP} = \mathsf{BPP} \subseteq \mathsf{ZPP}^{\mathsf{NP}} = \mathsf{ZPP}^{\mathsf{ZPP}} = \mathsf{ZPP}$$

which leads to a contradiction (see Appendix C). □

8 Excluding an Imperfect IO from **Heuristica**

Below we assume that the reader is familiar with the basic definitions of average-case complexity (in particular, the definitions of DistNP and AvgP); see, e.g., [BT06].

Theorem 8.1. *Suppose* $\mathsf{DistNP} \subseteq \mathsf{AvgP}$*. If an* ε*-imperfect computationally* $(1-\delta)$*-secure* IO *exists for* $1 - 2\varepsilon \geq \delta + 2n^{-\ell}$ *for some* $\ell > 0$*, then* $\mathsf{NP} = \mathsf{P}$*.*

Proof. If $\mathsf{DistNP} \subseteq \mathsf{AvgP}$, we get a language $L \in \mathsf{P}$ of polynomial density such that for every $x \in L$, the circuit complexity of x (when viewed as a truth table of a boolean function) is at least $|x|^{0.9}$. All results in this paper that use MCSP as an oracle continue to hold with any such L as an oracle instead. In particular, as in the proof of Theorem 6.1 (see Claim 6.2), we conclude that $\mathsf{NP} \subseteq \mathsf{BPP}^L = \mathsf{BPP}$. Finally, by [BFP05], if $\mathsf{DistNP} \subseteq \mathsf{AvgP}$ then $\mathsf{BPP} = \mathsf{P}$, and so $\mathsf{NP} = \mathsf{P}$. □

9 Open Questions

In this paper we showed that an (even imperfect) IO secure against non-uniform polynomial-size circuits implies non-trivial circuit lower bounds. Can one prove circuit lower bounds from the assumption that a (uniform) computationally-secure IO exists?

Can we leverage the connection between one-way functions and a close relative of MCSP (time-bounded Kolmogorov complexity) [LP20; All+21] to get better understanding of IO?

Acknowledgments

The authors would thank Moni Naor and Guy Rothblum for answering our questions regarding [NR06]. We also thank the anonymous referees for their useful comments.

References

[AD17] Eric Allender and Bireswar Das. "Zero knowledge and circuit minimization". In: *Inf. Comput.* 256 (2017), pp. 2–8. DOI: 10.1016/j.ic.2017.04.004.

[Adl78] L. M. Adleman. "Two Theorems on Random Polynomial Time". In: *Proceedings of the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 1978, pp. 75–83.

[AH91] William Aiello and Johan Håstad. "Statistical Zero-Knowledge Languages can be Recognized in Two Rounds". In: *J. Comput. Syst. Sci.* 42.3 (1991), pp. 327–345. DOI: 10.1016/0022-0000(91)90006-Q.

[All+06] Eric Allender, Harry Buhrman, Michal Koucký, Dieter van Melkebeek, and Detlef Ronneburger. "Power from Random Strings". In: *SIAM J. Comput.* 35.6 (2006), pp. 1467–1493. DOI: 10.1137/050628994.

[All+21] Eric Allender, Mahdi Cheraghchi, Dimitrios Myrisiotis, Harsha Tirumala, and Ilya Volkovich. “One-Way Functions and a Conditional Variant of MKTP”. In: *41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2021, December 15-17, 2021, Virtual Conference*. Ed. by Mikolaj Bojanczyk and Chandra Chekuri. Vol. 213. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 7:1–7:19. DOI: 10.4230/LIPIcs.FSTTCS.2021.7.

[Arv+95] Vikraman Arvind, Johannes Köbler, Uwe Schöning, and Rainer Schuler. “If NP has Polynomial-Size Circuits, then MA=AM”. In: *Theor. Comput. Sci.* 137.2 (1995), pp. 279–282. DOI: 10.1016/0304-3975(95)91133-B.

[Bar+12] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. “On the (im)possibility of obfuscating programs”. In: *J. ACM* 59.2 (2012), 6:1–6:48. DOI: 10.1145/2160158.2160159.

[Bar02] Boaz Barak. “A Probabilistic-Time Hierarchy Theorem for ”Slightly Non-uniform” Algorithms”. In: *Randomization and Approximation Techniques, 6th International Workshop, RANDOM 2002, Cambridge, MA, USA, September 13-15, 2002, Proceedings*. Ed. by José D. P. Rolim and Salil P. Vadhan. Vol. 2483. Lecture Notes in Computer Science. Springer, 2002, pp. 194–208. DOI: 10.1007/3-540-45726-7_16.

[BBF16] Zvika Brakerski, Christina Brzuska, and Nils Fleischhacker. “On Statistically Secure Obfuscation with Approximate Correctness”. In: *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9815. Lecture Notes in Computer Science. Springer, 2016, pp. 551–578. DOI: 10.1007/978-3-662-53008-5_19.

[BFP05] Harry Buhrman, Lance Fortnow, and Aduri Pavan. “Some Results on Derandomization”. In: *Theory Comput. Syst.* 38.2 (2005), pp. 211–227. DOI: 10.1007/s00224-004-1194-y.

[BHZ87] Ravi B. Boppana, Johan Håstad, and Stathis Zachos. “Does co-NP Have Short Interactive Proofs?” In: *Inf. Process. Lett.* 25.2 (1987), pp. 127–132. DOI: 10.1016/0020-0190(87)90232-8.

[Bsh+96] Nader H. Bshouty, Richard Cleve, Ricard Gavaldà, Sampath Kannan, and Christino Tamon. “Oracles and Queries That Are Sufficient for Exact Learning”. In: *J. Comput. Syst. Sci.* 52.3 (1996), pp. 421–433. DOI: 10.1006/jcss.1996.0032.

[BT00] Harry Buhrman and Leen Torenvliet. “Randomness is Hard”. In: *SIAM J. Comput.* 30.5 (2000), pp. 1485–1501. DOI: 10.1137/S0097539799360148.

[BT06] Andrej Bogdanov and Luca Trevisan. “Average-Case Complexity”. In: *Found. Trends Theor. Comput. Sci.* 2.1 (2006). DOI: 10.1561/0400000004.

[BZ17] Dan Boneh and Mark Zhandry. “Multiparty Key Exchange, Efficient Traitor Tracing, and More from Indistinguishability Obfuscation”. In: *Algorithmica* 79.4 (2017), pp. 1233–1285. DOI: 10.1007/s00453-016-0242-8.

[EG95] Thomas Eiter and Georg Gottlob. “Identifying the Minimal Transversals of a Hypergraph and Related Problems”. In: *SIAM J. Comput.* 24.6 (1995), pp. 1278–1304. DOI: 10.1137/S0097539793250299.

[FK96] Michael L. Fredman and Leonid Khachiyan. "On the Complexity of Dualization of Monotone Disjunctive Normal Forms". In: *J. Algorithms* 21.3 (1996), pp. 618–628. DOI: 10.1006/jagm.1996.0062.

[For89] Lance Fortnow. "The Complexity of Perfect Zero-Knowledge". In: *Adv. Comput. Res.* 5 (1989), pp. 327–343.

[FS04] Lance Fortnow and Rahul Santhanam. "Hierarchy Theorems for Probabilistic Polynomial Time". In: *45th Symposium on Foundations of Computer Science (FOCS 2004), 17-19 October 2004, Rome, Italy, Proceedings*. IEEE Computer Society, 2004, pp. 316–324. DOI: 10.1109/FOCS.2004.33.

[Gar+16] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. "Candidate Indistinguishability Obfuscation and Functional Encryption for All Circuits". In: *SIAM J. Comput.* 45.3 (2016), pp. 882–929. DOI: 10.1137/14095772X.

[GHM08] Judy Goldsmith, Matthias Hagen, and Martin Mundhenk. "Complexity of DNF minimization and isomorphism testing for monotone formulas". In: *Inf. Comput.* 206.6 (2008), pp. 760–775. DOI: 10.1016/j.ic.2008.03.002.

[Gol90] Oded Goldreich. "A Note on Computational Indistinguishability". In: *Inf. Process. Lett.* 34.6 (1990), pp. 277–281. DOI: 10.1016/0020-0190(90)90010-U.

[GP15] Sanjam Garg and Antigoni Polychroniadou. "Two-Round Adaptively Secure MPC from Indistinguishability Obfuscation". In: *Theory of Cryptography - 12th Theory of Cryptography Conference, TCC 2015, Warsaw, Poland, March 23-25, 2015, Proceedings, Part II*. Ed. by Yevgeniy Dodis and Jesper Buus Nielsen. Vol. 9015. Lecture Notes in Computer Science. Springer, 2015, pp. 614–637. DOI: 10.1007/978-3-662-46497-7_24.

[GR14] Shafi Goldwasser and Guy N. Rothblum. "On Best-Possible Obfuscation". In: *J. Cryptol.* 27.3 (2014), pp. 480–505. DOI: 10.1007/s00145-013-9151-z.

[IKV18] Russell Impagliazzo, Valentine Kabanets, and Ilya Volkovich. "The Power of Natural Properties as Oracles". In: *33rd Computational Complexity Conference, CCC 2018, June 22-24, 2018, San Diego, CA, USA*. Ed. by Rocco A. Servedio. Vol. 102. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, 7:1–7:20. DOI: 10.4230/LIPIcs.CCC.2018.7.

[IL89] Russell Impagliazzo and Michael Luby. "One-way Functions are Essential for Complexity Based Cryptography (Extended Abstract)". In: *30th Annual Symposium on Foundations of Computer Science, Research Triangle Park, North Carolina, USA, 30 October - 1 November 1989*. IEEE Computer Society, 1989, pp. 230–235. DOI: 10.1109/SFCS.1989.63483.

[Imp95] Russell Impagliazzo. "A Personal View of Average-Case Complexity". In: *Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995*. IEEE Computer Society, 1995, pp. 134–147. DOI: 10.1109/SCT.1995.514853.

[JLS21] Aayush Jain, Huijia Lin, and Amit Sahai. "Indistinguishability obfuscation from well-founded assumptions". In: *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*. Ed. by Samir Khuller and Virginia Vassilevska Williams. ACM, 2021, pp. 60–73. DOI: 10.1145/3406325.3451093.

[JVV86] Mark Jerrum, Leslie G. Valiant, and Vijay V. Vazirani. "Random Generation of Combinatorial Structures from a Uniform Distribution". In: *Theor. Comput. Sci.* 43 (1986), pp. 169–188. DOI: 10.1016/0304-3975(86)90174-X.

[Kan82] Ravi Kannan. "Circuit-Size Lower Bounds and Non-Reducibility to Sparse Sets". In: *Inf. Control.* 55.1-3 (1982), pp. 40–56. DOI: 10.1016/S0019-9958(82)90382-5.

[KC00] Valentine Kabanets and Jin-yi Cai. "Circuit minimization problem". In: *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA*. Ed. by F. Frances Yao and Eugene M. Luks. ACM, 2000, pp. 73–79. DOI: 10.1145/335305.335314.

[KNY17] Ilan Komargodski, Moni Naor, and Eylon Yogev. "Secret-Sharing for NP". In: *J. Cryptol.* 30.2 (2017), pp. 444–469. DOI: 10.1007/s00145-015-9226-0.

[Kom+14] Ilan Komargodski, Tal Moran, Moni Naor, Rafael Pass, Alon Rosen, and Eylon Yogev. "One-Way Functions and (Im)Perfect Obfuscation". In: *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*. IEEE Computer Society, 2014, pp. 374–383. DOI: 10.1109/FOCS.2014.47.

[KW98] Johannes Köbler and Osamu Watanabe. "New Collapse Consequences of NP Having Small Circuits". In: *SIAM J. Comput.* 28.1 (1998), pp. 311–324. DOI: 10.1137/S0097539795296206.

[LP20] Yanyi Liu and Rafael Pass. "On One-way Functions and Kolmogorov Complexity". In: *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*. Ed. by Sandy Irani. IEEE, 2020, pp. 1243–1254. DOI: 10.1109/FOCS46700.2020.00118.

[MP07] Dieter van Melkebeek and Konstantin Pervyshev. "A Generic Time Hierarchy with One Bit of Advice". In: *Comput. Complex.* 16.2 (2007), pp. 139–179. DOI: 10.1007/s00037-007-0227-8.

[NR06] Moni Naor and Guy N. Rothblum. "Learning to impersonate". In: *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006*. Ed. by William W. Cohen and Andrew W. Moore. Vol. 148. ACM International Conference Proceeding Series. ACM, 2006, pp. 649–656. DOI: 10.1145/1143844.1143926.

[RR97] Alexander A. Razborov and Steven Rudich. "Natural Proofs". In: *J. Comput. Syst. Sci.* 55.1 (1997), pp. 24–35. DOI: 10.1006/jcss.1997.1494.

[San09] Rahul Santhanam. "Circuit Lower Bounds for Merlin–Arthur Classes". In: *SIAM J. Comput.* 39.3 (2009), pp. 1038–1061. DOI: 10.1137/070702680.

[SV03] Amit Sahai and Salil P. Vadhan. "A complete problem for statistical zero knowledge". In: *J. ACM* 50.2 (2003), pp. 196–249. DOI: 10.1145/636865.636868.

[SW21] Amit Sahai and Brent Waters. "How to Use Indistinguishability Obfuscation: Deniable Encryption, and More". In: *SIAM J. Comput.* 50.3 (2021), pp. 857–908. DOI: 10.1137/15M1030108.

[Tra84] Boris A. Trakhtenbrot. "A Survey of Russian Approaches to Perebor (Brute-Force Searches) Algorithms". In: *IEEE Ann. Hist. Comput.* 6.4 (1984), pp. 384–400. DOI: 10.1109/MAHC.1984.10036.

[Vol14] Ilya Volkovich. "On Learning, Lower Bounds and (un)Keeping Promises". In: *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I.* Ed. by Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias. Vol. 8572. Lecture Notes in Computer Science. Springer, 2014, pp. 1027–1038. DOI: 10.1007/978-3-662-43948-7_85.

[Vol23] Ilya Volkovich. "The final nail in the coffin of statistically-secure obfuscator". In: *Information Processing Letters* 182 (2023), p. 106366. ISSN: 0020-0190. DOI: https://doi.org/10.1016/j.ipl.2023.106366.

[Wil13] Ryan Williams. "Towards NEXP versus BPP?" In: *Computer Science - Theory and Applications - 8th International Computer Science Symposium in Russia, CSR 2013, Ekaterinburg, Russia, June 25-29, 2013. Proceedings.* Ed. by Andrei A. Bulatov and Arseny M. Shur. Vol. 7913. Lecture Notes in Computer Science. Springer, 2013, pp. 174–182. DOI: 10.1007/978-3-642-38536-0_15.

[Wil14] Ryan Williams. "Nonuniform ACC Circuit Lower Bounds". In: *J. ACM* 61.1 (2014), 2:1–2:32. DOI: 10.1145/2559903.

A A Universal Distinguisher in $\mathsf{FBPP}^{\mathsf{NP}}$

Lemma A.1 ([JVV86])**.** *There exists a randomized algorithm that given oracle access to* NP *can approximate any function $f(x)$ in* $\#\mathsf{P}$ *to within the multiplicative factor $(1 \pm \varepsilon)$, with probability at least $1-\gamma$, in time polynomial in $|x|$, $1/\varepsilon$, and* $\log(1/\gamma)$*.*

Theorem A.2. *There is an* $\mathsf{FBPP}^{\mathsf{NP}}$ *algorithm $\mathcal{D}$, that given circuits C_0 and C_1 that are samplers for distributions D_0 and D_1, and $K \in \mathbb{N}$ in unary, will distinguish D_0 and D_1 with the distinguishing advantage at least $\delta - 1/K$, where δ is the statistical distance between D_0 and D_1.*

Proof. Given C_0 and C_1, let $p_0(x)$ be the probability of x according to D_0, and $p_1(x)$ that according to D_1. For $0 < \gamma = \varepsilon \leq 1/2$ to be determined, consider the following probabilistic circuit $A(x;r)$: Compute the estimates $\tilde{p}_0(x) = (1 \pm \varepsilon)p_0(x)$ and $\tilde{p}_1(x) = (1 \pm \varepsilon)p_1(x)$ with probability at least $1-\gamma$ (using the algorithm from Lemma A.1), and accept iff $\tilde{p}_0(x) > \tilde{p}_1(x)$.

We say that randomness r is good for x if both estimates $\tilde{p}_0(x)$ and $\tilde{p}_1(x)$ are correct within the multiplicative factor $(1 \pm \varepsilon)$. Note that by Lemma A.1, for every x, r is good for x with probability at least $1-2\gamma$. We have

$$\begin{aligned}
&\mathbf{Pr}_{x\sim D_0,r}[A(x;r)=1]-\mathbf{Pr}_{x\sim D_1,r}[A(x;r)=1]\\
&\geq \mathbf{Pr}_{x\sim D_0,r}[A(x;r)=1 \mid\ r \text{ is good for } x]-\mathbf{Pr}_{x\sim D_1,r}[A(x;r)=1 \mid\ r \text{ is good for } x]-2\gamma\\
&= \sum_{x\colon p_0(x)>p_1(x)} (p_0(x)-p_1(x))-2\gamma\\
&\quad - \sum_{x\colon p_0(x)>p_1(x)\,\wedge\,\tilde{p}_0(x)<\tilde{p}_1(x)} (p_0(x)-p_1(x))\\
&\quad + \sum_{x\colon p_0(x)\leq p_1(x)\,\wedge\,\tilde{p}_0(x)>\tilde{p}_1(x)} (p_0(x)-p_1(x)).
\end{aligned}$$

Note that

$$\begin{aligned}
&\sum_{x\colon p_0(x)>p_1(x)\,\wedge\,\tilde{p}_0(x)<\tilde{p}_1(x)} (p_0(x)-p_1(x))\\
&\leq \sum_{x\colon (p_0(x)>p_1(x))\,\wedge\,((1-\varepsilon)p_0(x)<(1+\varepsilon)p_1(x))} (p_0(x)-p_1(x))\\
&\leq \sum_x ((1+\varepsilon)/(1-\varepsilon)-1)\cdot p_1(x)\\
&= (2\varepsilon)/(1-\varepsilon).
\end{aligned}$$

Similarly,

$$\sum_{x\colon p_0(x)\leq p_1(x)\,\wedge\,\tilde{p}_0(x)>\tilde{p}_1(x)} (p_1(x)-p_0(x)) \leq (2\varepsilon)/(1-\varepsilon).$$

Putting everything together, we get

$$\mathbf{Pr}_{x\sim D_0,r}[A(x;r)=1]-\mathbf{Pr}_{x\sim D_1,r}[A(x;r)=1] \geq \delta-(2\gamma+(4\varepsilon)/(1-\varepsilon)),$$

which is at least $\delta-10\varepsilon$. Setting $\varepsilon=1/(10K)$ concludes the proof. □

B Testing Equivalence of Monotone Formulas

Theorem B.1 ([EG95]). *Deciding if two given monotone formulas are equivalent is* coNP*-complete.*

Proof. Membership in coNP is clear. For coNP-hardness, we reduce from the complement of 3SAT. Let $\varphi(x_1,\dots,x_n)$ be a given 3-CNF. For each $1\leq i\leq n$, replace all occurrences of $\neg x_i$ in φ by a new variable y_i. Let $\varphi^{\text{MON}}(x_1,\dots,x_n,y_1,\dots,y_n)$ denote the resulting monotone 3-CNF.

Claim B.2. *φ is unsatisfiable iff*

$$\varphi^{\text{MON}} \vee \bigvee_{i=1}^{n}(x_i\wedge y_i) \;\equiv\; \bigvee_{i=1}^{n}(x_i\wedge y_i). \tag{10}$$

Proof of Claim B.2. Suppose φ is satisfied by an assignment $a\in\{0,1\}^n$ to its variables $x_1,\dots,x_n$. Set $y_i=\neg x_i$ for all $1\leq i\leq n$. Then φ^{MON} evaluates to 1 on this assignment, making the left-hand side of (10) true. But the right-hand side of (10) is obviously false under this assignment.

For the other direction, suppose an assignment $(a, b) \in \{0,1\}^{2n}$ to $x_1, \ldots, x_n, y_1, \ldots, y_n$ violates the equivalence of the expressions in (10). By monotonicity, this can only happen if the right-hand side of (10) is 0 under this assignment, but φ^{MON} evaluates to 1. The former means that, for each $1 \leq i \leq n$, either $x_i = 0$ or $y_i = 0$, or both $x_i = y_i = 0$. If both $x_i = y_i = 0$, then modify the current assignment (a, b) by setting $x_i = 1$. By monotonicity, the value of φ^{MON} is still 1 after this update in the assignment. Perform such an update to the assignment for all $1 \leq j \leq n$ where $x_j = y_j = 0$, getting a new assignment $(a', b) \in \{0,1\}^{2n}$ which still satisfies φ^{MON}. Since under this new assignment (a', b), we get that $y_i = \neg x_i$ for all $1 \leq i \leq n$, we conclude that φ is satisfied by the assignment $a' \in \{0,1\}^n$. □

The theorem follows. □

Thus, to decide a coNP-complete problem, it suffices to be able to test equivalence of monotone depth-3 (OR-AND-OR) formulas: a fixed monotone 2-DNF $\bigvee_{i=1}^{n}(x_i \wedge y_i)$ and the disjunction of this 2-DNF with an arbitrary monotone 3-CNF.

Remark B.3. *This hardness result essentially tight as far as the formula depth is concerned. Deciding the equivalence of two monotone depth-2 formulas is either in* P*, or in quasipolynomial time. In particular, if both formulas are DNFs (or both are CNFs), then testing for equivalence is in* P.[10] *If one is a DNF and the other one is a CNF, then testing for equivalence can be done in quasipolynomial time* $n^{o(\log n)}$ *[FK96]; if one formula is a* k*-CNF for a constant* $k \in \mathbb{N}$ *and the other one is a DNF, then equivalence testing is in* P *[EG95] (even in* LOGSPACE *[GHM08]).*

Remark B.4. *For monotone DNFs, we have a trivial deterministic polynomial-time* IO *which outputs the unique minimal DNF (consisting of prime implicants). The challenge is to get an* IO *for monotone depth-3 formulas that are ORs of CNFs.*

C Separating ZPEXP from ZPP

Claim C.1. ZPEXP $\neq$ ZPP.

Proof. Suppose for a contradiction that ZPEXP = ZPP. Then

$$\mathsf{NP}^{\mathsf{NP}} \subseteq \mathsf{EXP} \subseteq \mathsf{ZPEXP} \subseteq \mathsf{ZPP}.$$

By translation, $\mathsf{NEXP}^{\mathsf{NP}} \subseteq \mathsf{ZPEXP}$ and hence by Lemma 2.10, $\mathsf{ZPEXP} \not\subseteq \mathsf{P/poly}$. Yet, by Adleman's Theorem ([Adl78]) $\mathsf{ZPP} \subseteq \mathsf{BPP} \subseteq \mathsf{P/poly}$. □

Remark C.2. *Similarly, one can show that* BPEXP $\neq$ BPP. *However, separating* ZPEXP *or even* NEXP *or* $\mathsf{EXP}^{\mathsf{NP}}$ *from* BPP *remains a longstanding open question. See e.g. [BT00; Wil13].*

[10] Every monotone DNF has a unique minimal DNF consisting of all its prime implicants, and such a minimal form can be found in polynomial time for any given monotone DNF.