Structure vs. Hardness through the Obfuscation Lens

by

Akshay Degwekar

B.Tech. (Honors), Indian Institute of Technology Madras (2014)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

Massachusetts Institute of Technology

June 2016

© 2016 Massachusetts Institute of Technology. All rights reserved.

Signature of Author: ____________________________

Department of EECS,
May 20, 2016

Certified by: ____________________________

Prof. Vinod Vaikuntanathan,
Associate Professor, EECS

Accepted by: ____________________________

Prof. Leslie A. Kolodziejski,
Chair of the Committee on Graduate Students, EECS

Structure vs. Hardness through the Obfuscation Lens

by

Akshay Degwekar

Submitted to the Department of Electrical Engineering and Computer Science
on May 20, 2016 in partial fulfillment of the requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Cryptography relies on the computational hardness of structured problems. While one-way functions, the most basic cryptographic object, does not seem to require much structure, as we advance up the ranks into public-key cryptography and beyond, we seem to require that certain *structured problems are hard*. For example, factoring, quadratic residuosity, discrete logarithms, and approximate shortest and closest vectors in lattices all have considerable algebraic structure. This structure, on the one hand, enables useful applications such as public-key and homomorphic encryption, but on the other, also puts their hardness in question. Their structure is exactly what puts them in low complexity classes such as SZK or $\mathsf{NP} \cap \mathsf{coNP}$, and is in fact the reason behind (sub-exponential or quantum) algorithms for these problems. The question is whether such structure is inherent in different cryptographic primitives, deeming them inherently *easier*.

We study the relationship between two structured complexity classes, statistical zero-knowledge (SZK) and $\mathsf{NP} \cap \mathsf{coNP}$, and cryptography. To frame the question in a meaningful way, we rely on the language of black-box constructions and separations.

Our results are the following:

- Cryptography vs. Structured Hardness: Our two main results show that there are no black-box constructions of hard problems in SZK or $\mathsf{NP} \cap \mathsf{coNP}$ starting from one of a wide variety of cryptographic primitives such as one-way and trapdoor functions, one-way and trapdoor permutations (in the case of SZK), public-key encryption, oblivious transfer, deniable encryption, functional encryption, and even indistinguishability obfuscation;
- Complexity-theoretic Implications: As a corollary of our result, we show a separation between SZK and $\mathsf{NP} \cap \mathsf{coNP}$ and the class PPAD that captures the complexity of computing Nash Equilibria; and
- Positive Results: We construct collision-resistant hashing from a strong form of SZK-hardness and indistinguishability obfuscation. It was previously known that indistinguishability obfuscation by itself does not imply collision-resistant hashing in a black-box way; we show that it does if one adds SZK-hardness as a "catalyst".

Our black-box separations are derived using indistinguishability obfuscation as a "gateway", by first showing a (separation) result for indistinguishability obfuscation and then leveraging on the fact that indistinguishability obfuscation can be used to construct the above variety of cryptographic primitives and hard PPAD instances in a black-box manner.

Thesis Supervisor: Vinod Vaikuntanathan
Title: Associate Professor

Acknowledgements

I would like to thank my adviser Vinod Vaikuntanathan for his advice and encouragement. Vinod is an amazing researcher and a very caring mentor. I would be elated if my work-ethic is ever half as good as his. Over the last two years, I got to explore numerous exciting research problems for which I am grateful. I thank Shafi Goldwasser for not only discovering so many of the notions we hold dear in cryptography but also making my decision to attend MIT so easy. I look forward to becoming a better researcher under their guidance.

This thesis is based on joint work with Nir Bitansky and Vinod Vaikuntanathan. I thank them for this very enjoyable and fruitful time. Choosing a good problem to work on is half the struggle and I thank Nir for rescuing me from this.

I have had the privilege of collaborating with some amazing people including Itay Berman, Ron Rothblum, Jayalal Sarma and Prashant Vasudevan. Special thanks to Jayalal Sarma, my undergraduate adviser, for introducing me to to theory through the many courses he taught and for being a very patient and encouraging mentor to me when I was a clueless undergrad taking his first steps in research. I would like to thank Prashant for making me realize how much fun it was to collaborate. I thank Itay for his forthright and wise counsel in research and everything else.

Being a part of the theory group is a great privilege. I would like to thank my graduate counselor Silvio Micali for his great advice and Madhu Sudan for great courses, the social outings and importantly free beer. I would also like to thank the administrative assistants especially Linda Lynch and Debbie Lehto for making things happen. Grad school would not have been so much fun without all the residents of Stata 5th and 6th floor. In particular, thanks to Sam Park, Govind Ramnarayanan, Nishanth Dikkala, Tianren Liu, Pritish Kamath and Adam Sealfon.

My time in IIT-M would not have been half as good without the amazing friends I had. I would like to thank Abhik Mondal, Jaideep Pathak, Karthik Abinav, Saikrishna Badrinarayanan, Sameer Wagh and Rohit Patki for making my stay at IIT-M memorable.

Finally I would like to thank my family: my parents Dhananjai and Vidya and my siblings Aditya and Anand. To my parents I owe eternal gratitude for my very existence, a superb upbringing and their unconditional love. Aditya for being a great older brother even in the face of very cliquish behavior by us. Lastly my twin Anand for a lifetime of shared experiences. Our boisterous play, deathmatch-esque fights and never-ending conversations helped define who I am.

Contents

Chapter 1

Introduction

The last four decades of research in the theory of cryptography has produced a host of fantastic objects, starting from one-way functions and permutations to public-key encryption [DH76, RSA78, GM82] and zero-knowledge proofs [GMR85] in the 1980s, all the way to fully homomorphic encryption [RAD78, Gen09, BV11] and indistinguishability obfuscation [BGI$^+$01, GGH$^+$13] in the modern day.

The existence of all these objects require at the very minimum that $\mathsf{NP} \not\subseteq \mathsf{BPP}$, but that is hardly ever enough. While one-way functions, the most basic cryptographic object, does not seem to require much structure, as we advance up the ranks, we seem to require that certain *structured problems are hard.* For example, the conjectured hard problems used in most of cryptography (especially the public-key kind), such as quadratic residuosity, discrete logarithms, and approximate shortest and closest vectors on lattices all have considerable algebraic structure. It is their structure that enables useful applications such as public-key and homomorphic encryption, but it is also their structure that enables surprising (sub-exponential) algorithms for these problems, and puts them in low complexity classes such as SZK and $\mathsf{NP} \cap \mathsf{coNP}$. Moreover, we know that some cryptographic primitives *necessarily* imply the hardness of structured problems: the existence of one-way permutations requires a hard problem in $\mathsf{NP} \cap \mathsf{coNP}$ [Bra79]; any fully homomorphic encryption scheme implies a hard problem in SZK [BL13]; and any indistinguishability obfuscation scheme implies a hard problem in $\mathsf{PPAD} \subseteq \mathsf{TFNP}$ [BPR15].

There is of course the fear that the structure that makes these problems *useful* might also make them *easy* (eventually, if not today). Or, as Barak says more eloquently [Bar13]:

> [...] *based on the currently well studied schemes, structure is strongly associated with (and perhaps even implied by) public key cryptography. This is troubling news, since it makes public key crypto somewhat of an "endangered species" that could be wiped out by a surprising algorithmic advance. Therefore the question of whether structure is inherently necessary for public key crypto is not only of mathematical interest but also of practical importance as well.*

Thus, it is natural to ask:

What type of structure is *necessary* for different cryptographic primitives?

Could there be one-way functions if $\mathsf{NP} \cap \mathsf{coNP}$ collapses? Is there hope of constructing one-way permutations if SZK collapses? How about more advanced cryptographic primitives like public-key encryption and indistinguishability obfuscation? More broadly: to what extent does cryptography rely on structured hardness? These are the types of questions we seek to answer in this work.

Black-box Separations. In order to frame these problems in a meaningful way, we study these questions in the language of black-box constructions, reductions and separations, whose formal study in the context of cryptography was initiated by Impagliazzo and Rudich [IR89, Rud84] more than two decades ago. At a very high level, we aim to show theorems which state that "the existence of crypto primitive $\mathcal{P}$ does not imply hardness in a complexity class $\mathcal{C}$". It is easy to formalize a statement of the form "the existence of crypto primitive $\mathcal{P}$ *implies* hardness in a complexity class $\mathcal{C}$": one just needs to show a reduction from breaking $\mathcal{P}$ to solving problems in $\mathcal{C}$. On the other hand, a naïve formulation of our desired statement (for example, where the primitive $\mathcal{P}$ is one-way functions and the complexity class $\mathcal{C}$ is SZK), will involve showing that there are problems in $\mathsf{NP} \setminus \mathsf{SZK}$, a task that is beyond the reach of complexity theory at this point.

Impagliazzo and Rudich, followed by several others, proposed to circumvent this difficulty by instead asking to put limits on "natural classes of constructions" of a hard problem in complexity class $\mathcal{C}$ starting from the existence of crypto primitive $\mathcal{P}$. Slightly more elaborately, we call a construction of problem $\mathcal{P}'$ from primitive $\mathcal{P}$ black-box when the construction does not exploit the internal structure of an implementation of primitive $\mathcal{P}$, but rather just the input-output interface. A security reduction is called black-box if the adversary that "breaks" $\mathcal{P}$ uses the one that "solves" $\mathcal{P}'$ (as well as the primitive $\mathcal{P}$) as a black box. Being able to rule out such *fully black-box* constructions [RTV04] can be seen as evidence for a separation, or a guide as to the barriers that need to be overcome and the techniques that need to be involved in an eventual construction of a hard problem in $\mathcal{P}'$ starting from crypto primitive $\mathcal{P}$.

There have been many subsequent black-box separations (see, e.g., [Rud91, Sim98, KST99, GKM+00, GT00, GMR01, BT03, GGKT05, AGGM06, Pas06, GMM07, BM09, HH09, KSS11, BKSY11, DLMM11, GKLM12, DHT12, Pas13, BL13, BB15, HHRS15a] and many others), and many works that explore the taxonomies and nuances of black-box reductions [RTV04, HR04, Fis12, BBF13]. In particular, the most relevant to us are the recent works of Asharov and Segev [AS15, AS16] that study black-box separations in the context of indistinguishability obfuscation that seems as an inherently non-black-box primitive.

1.1 Our Results

We study the relationship between two structured complexity classes, namely statistical zero-knowledge (SZK) and $\mathsf{NP} \cap \mathsf{coNP}$, and cryptography. In broad strokes, our results show that there are no fully black-box constructions of hard problems in these classes starting from one of a wide variety of cryptographic primitives such as one-way and trapdoor functions, one-way and trapdoor permutations (in the case of SZK), public-key encryption, oblivious transfer, deniable encryption, functional encryption, and even indistinguishability obfuscation. Our results are derived using indistinguishability obfuscation as a "gateway", by first showing the (separation) result for indistinguishability obfuscation (in the framework of Asharov and Segev [AS15]), and then leveraging on the fact that indistinguishability obfuscation can be used to construct a wide variety of cryptographic primitives. In addition, we show:

- Complexity-theoretic Implications: One of the corollaries of our result is a separation between SZK and $\mathsf{NP} \cap \mathsf{coNP}$ from the class PPAD that captures the complexity of computing Nash Equilibria [MP91]; and
- Positive Results: We construct collision-resistant hashing from a strong form of SZK-hardness and indistinguishability obfuscation. It was previously known [AS15] that indistinguishability obfuscation by itself does not imply collision-resistant hashing in a black-box way; we show that it does if one adds SZK-hardness as a "catalyst".

We elaborate below.

1.1.1 Statistical Zero-Knowledge and Cryptography.

The notion of statistical zero-knowledge proofs was introduced in the seminal work of Goldwasser, Micali and Rackoff [GMR85]. The class of promise problems with statistical zero-knowledge proofs (SZK) can be characterized by several complete problems, such as *statistical distance* and *entropy difference* (see [Vad99] and references within). SZK hardness is known to follow from various number-theoretic problems that are commonly used in cryptography, such as Discrete Logarithms [GK93], Quadratic Residuosity [GMR85], and Lattice Problems [MV03]. We also know that a small handful of cryptographic primitives such as homomorphic encryption [BL13], private information retrieval [LV16] and re-randomizable encryption imply SZK-hardness. (On the other hand, $\mathsf{SZK} \subseteq \mathsf{AM} \cap \mathsf{coAM}$, and thus, SZK cannot contain NP-hard problems, unless the polynomial hierarchy collapses.)

We ask more generally which cryptographic primitives can be shown to imply such hardness, with the intuition that such primitives are *structured* in a certain way. In particular, whereas one may not expect a completely unstructured object like one-way functions to imply such hardness, what can we say for instance about public-key encryption, trapdoor permutations, or even indistinguishability obfuscation (which has proven to be structured enough to yield almost any known cryptographic goal)?

Our first main result is that none of these primitives imply such hardness through the natural class of black-box constructions and security reductions.

Theorem 1.1 (Informal)**.** *There is no fully black-box construction of a hard problem in* SZK *from the following cryptographic primitives: one-way permutations, public-key encryption, oblivious transfer, trapdoor permutations, deniable encryption, functional encryption, indistinguishability* *obfuscation and thus, any other cryptographic primitive that can be constructed from these in a fully black-box way.*

We would like to elaborate a bit more on what a black-box construction of a hard statistical distance problem means. We shall focus on the characterization of SZK by the statistical distance problem. Here an instance is a pair of circuit samplers $C_0, C_1 : \{0,1\}^n \to \{0,1\}^m$ which induce distributions $\boldsymbol{C}_0$ and $\boldsymbol{C}_1$ that evaluate C_0 and C_1 on a uniformly random input. The promise is that the statistical distance $s = \Delta(\boldsymbol{C}_0, \boldsymbol{C}_1)$ of the corresponding distributions is either large (say, $s \geq 2/3$) or small (say, $s \leq 1/3$). The problem, named $\mathbf{SD}^{1/3,2/3}$ (or just $\mathbf{SD}$), is to decide which is the case.

Let us look at a specific example of the construction of such a problem from *rerandomizable encryption*. In a (say, symmetric-key) rerandomizable encryption scheme, on top of the usual encryption and decryption algorithms $(\mathsf{Enc}, \mathsf{Dec})$ there is a ciphertext rerandomization algorithm ReRand that can statistically refresh ciphertexts. Namely, for any ciphertext C encrypting a bit b, $\mathsf{ReRand}(\mathsf{C})$ produces a ciphertext that is statistically close to a fresh encryption $\mathsf{Enc}_{\mathsf{sk}}(b)$. Note that this immediately gives rise to a hard statistical distance problem: given a pair of ciphertexts $(\mathsf{C}, \mathsf{C}')$, decide whether the corresponding rerandomized distributions given by the circuits $(C_0(\cdot), C_1(\cdot)) := (\mathsf{ReRand}(\mathsf{C}; \cdot), \mathsf{ReRand}(\mathsf{C}'; \cdot))$ are statistically far or close. Indeed, this corresponds to whether they encrypt the same bit or not, which is hard to decide by the security of the encryption scheme.

A feature of this construction of hard statistical distance instances is that, similarly to most constructions in cryptography, it is *fully black-box* [RTV04] in the sense that the circuits C_0, C_1 only make black-box use of the encryption scheme's algorithms, and can in fact be represented as oracle-aided circuits $(C_0^{\mathsf{ReRand}(\cdot)}, C_1^{\mathsf{ReRand}(\cdot)})$. Furthermore, "hardness" can be shown by a black-box reduction that can use any decider for the problem in a black-box way to break the underlying encryption scheme. More generally, one can consider the statistical distance problem relative to different oracles implementing different cryptographic primitives and ask when can hardness be shown based on a black-box reduction. Theorem 1.1 rules out such reductions relative to IO and OWPs (and everything that follows from these in a fully black-box way). For more details, see Section 1.2 and Section 3.

1.1.2 $\mathsf{NP} \cap \mathsf{coNP}$ and Cryptography.

Hard (on average) problems in $\mathsf{NP} \cap \mathsf{coNP}$ are known to follow based on certain number-theoretic problems in cryptography, such as Discrete Logs, Factoring and Lattice Problems [AR04]. As in the previous section for SZK, we are interested in understanding which cryptographic primitives would imply such hardness, again with the intuition that this implies structure. For instance, it is well-known that any one-way permutation $f : \{0,1\}^n \to \{0,1\}^n$ implies a hard problem in $\mathsf{NP} \cap \mathsf{coNP}$, e.g. given an index $i \in [n]$ and an image $f(x)$ find the ith preimage bit x_i. In contrast, in his seminal work, Rudich [Rud84] proved that completely unstructured objects like one-way functions do not imply hardness in $\mathsf{NP} \cap \mathsf{coNP}$ by fully black-box constructions. In this context, a fully black-box construction essentially means that the non-deterministic verifiers only make black-box use of the OWF (or OWP in the previous example) and the reduction establishing the hardness is also black-box (in both the adversary and the OWF).

But what about more structured primitives such as public-key encryption, oblivious transfer, or even indistinguishability obfuscation. Indeed, IO (plus OWFs) has-been shown to imply hardness in PPAD and more generally in the class TFNP of total search problems, which is often viewed as the search analog of $\mathsf{NP} \cap \mathsf{coNP}$ [MP91]. We will show, however, that fully black-box constructions do not give rise to hard problems in $\mathsf{NP} \cap \mathsf{coNP}$ from OWFs (or even *injective OWFs*) and IO (and everything that follows from these in a fully black-box way).

Theorem 1.2 (Informal)**.** *There is no fully black-box construction of a hard problem in* $\mathsf{NP} \cap \mathsf{coNP}$ *from the following cryptographic primitives: injective one-way functions, public-key encryption, oblivious transfer, deniable encryption, functional encryption, indistinguishability obfuscation and thus, any other cryptographic primitive that can be constructed from these in a fully black-box way.*

For more details, see Section 1.2 and Section 4.

1.1.3 Indistinguishability Obfuscation as a "Gateway".

Our results are derived using indistinguishability obfuscation as a "gateway", in two steps.

1. First, in Theorems 3.3 and 4.3, we show the (separation) result for indistinguishability obfuscation, namely that there are no fully black-box constructions of SZK-hardness (resp. $\mathsf{NP} \cap \mathsf{coNP}$-hardness) from one-way functions and indistinguishability obfuscation.

2. Second, we leverage on the long sequence of works in cryptography, starting from Sahai and Waters [SW14a], which use indistinguishability obfuscation can be used to construct a wide variety of cryptographic primitives, such as public-key encryption and oblivious transfer [SW14a], deniable encryption [SW14a] and functional encryption [Wat15]. Together with the first step, we show in one shot that there are no fully black-box constructions of SZK-hardness (resp. $\mathsf{NP} \cap \mathsf{coNP}$-hardness) from any of these primitives.

Showing a black-box separation from indistinguishability obfuscation (IO) needs a great deal of care, given that the typical use of IO makes non-black-box use of the circuits it obfuscates and thus any associated cryptographic primitive such as a one-way function. We follow the framework of Asharov and Segev [AS15] who consider obfuscators that take as input circuits with OWF- or OWP-gates. Since most known IO-based constructions fall into this category, such a separation is strong enough to rule out SZK (resp. $\mathsf{NP} \cap \mathsf{coNP}$) hardness from a wide variety of cryptographic primitives. See Section 1.2 for more details.

Theorems for Complexity Theorists. One of the corollaries of our result is a separation between SZK and $\mathsf{NP} \cap \mathsf{coNP}$ from the complexity class PPAD. PPAD, a subclass of total NP search problems called TFNP [MP91], was defined by Papadimitriou [Pap94] and has been shown to capture the complexity of computing Nash equilibria [DGP06, CDT09]. Bitansky, Paneth and Rosen [BPR15] have recently shown that indistinguishability obfuscation and injective one-way functions can be used (in a black-box way) to construct hard problems in PPAD. Put together with our separation, we get that there is no black-box construction of an SZK (resp. $\mathsf{NP} \cap \mathsf{coNP}$) hard problem from PPAD-hardness.

Theorem 1.3 (Informal)**.** *There is no fully black-box construction of a hard problem in* SZK *or* $\mathsf{NP} \cap \mathsf{coNP}$ *from hard problems in* PPAD*.*

Given that TFNP, which contains PPAD, is commonly thought of as a search version of $\mathsf{NP} \cap \mathsf{coNP}$, it is interesting to note that the result shows that hardness in $\mathsf{NP} \cap \mathsf{coNP}$ (aka, hardness of decisional problems) does not follow from hardness in TFNP (aka, hardness of search problems) in a black-box way. Namely, there is no black-box "search-to-decision reduction" between these classes.

1.1.4 A Positive Result: Collision-Resistant Hashing from Strong SZK-Hardness.

We end our paper with a positive result. While most of our focus has been on showing that hardness in SZK and $\mathsf{NP} \cap \mathsf{coNP}$ does *not* follow from cryptography, here we ask the "inverse question", namely whether certain cryptographic primitives can be built from other cryptographic primitives together with hardness in certain structured complexity classes? Ostrovsky and Wigderson [OW93a] showed that average-case SZK-hardness gives us one-way functions. Applebaum and Raykov [AR16] showed that average-case hardness in the sub-class $\mathsf{PRE} \subseteq \mathsf{SRE} \subseteq \mathsf{SZK}$ of languages with a perfect randomized encoding gives us collision-resistant hashing.

We construct collision-resistant hashing from a strong form of SZK-hardness and indistinguishability obfuscation. It was previously known [AS15] that indistinguishability obfuscation by itself does not imply collision-resistant hashing in a black-box way; we show that it does if one adds SZK-hardness as a "catalyst". Slightly more precisely, in the SZK-complete problem $\mathbf{SD}^{1/3,2/3}$ is is required to distinguish between distributions that are 1/3-close from ones that are 2/3-far. We show that indistinguishability obfuscation together with average-case hardness of $\mathbf{SD}^{0,1}$ (a stronger assumption) implies collision-resistant hashing.

Theorem 1.4 (Informal)**.** *Assuming average-case hardness of* $\mathbf{SD}^{0,1}$ *and the existence of indistinguishability obfuscation, there is a collision-resistant hashing scheme.*

1.1.5 Related Work

Oracle separations have been studied in both Computational Complexity theory and Cryptography. In computational complexity, starting with the seminal work of Baker, Gill and Solovay [BGS75], oracle separations have been used to show limitations on relativizing proof techniques. Most of this work has focused on constructing oracles relative to which relationships between two complexity classes can be inferred. In Cryptography, starting with the work of Impagliazzo and Rudich [IR89], oracle separations have been used to show limitations on constructions of cryptographic primitives from other cryptographic primitives in a black-box fashion. Most of the works here focus on the relationship between two cryptographic primitives.

In this work, we try to understand the relationship between the hardness of complexity classes and the existence of cryptographic primitives. There are two related questions here:

1. What cryptography can we get assuming complexity theoretic separations?

2. Does the existence of cryptographic primitives imply complexity theoretic separations?

Cryptography from Complexity. While most of cryptography is based on specific hardness assumptions, it is desirable to base cryptography on minimal complexity theoretic hardness assumptions. Ostrovsky and Wigderson [OW93a] showed that *a*verage-case SZK hardness implies one-way functions. Ong and Vadhan [OV08] construct stronger forms of commitment schemes from average-case SZK hardness.

On the other hand in terms of impossibility results, a line of work starting with Bogdanov-Trevisan [BT03, AGGM06, BB15, LV16] shows limitations on constructions of cryptographic primitives from worst-case NP-hardness.

Cryptography implies Complexity The existence of most cryptographic primitives implies $\mathsf{NP} \not\subset \mathsf{BPP}$. Moreover, we know that some cryptographic primitives *necessarily* imply the hardness of structured problems: the existence of one-way permutations requires a hard problem in $\mathsf{NP} \cap \mathsf{coNP}$ [Bra79]; any fully homomorphic encryption scheme implies a hard problem in SZK [BL13]; and any indistinguishability obfuscation scheme implies a hard problem in $\mathsf{PPAD} \subseteq \mathsf{TFNP}$ [BPR15].

In terms of impossibility results, the work of Rudich [Rud91] showed that one-way functions do not separate $\mathsf{NP} \cap \mathsf{coNP}$ from P in a black-box manner. The work of Haitner et.al. [HHRS15a] along with Ong and Vadhan [OV08] implies that one way permutations do not imply average-case SZK hardness in a black-box way. Our work extends these results and shows that a broad array of cryptographic primitives do not separate SZK or $\mathsf{NP} \cap \mathsf{coNP}$ from P in a black-box manner.

Impossibility results from IO Starting with the work of Sahai and Waters [SW14b], indistinguishability obfuscation has been used to construct a plethora of cryptographic primitives. The work of Asharov and Segev [AS15] showed that IO along with one-way permutations cannot construct collision resistant hash function families in a black-box manner. Another work by the same authors [AS16] shows that IO and OWFs cannot construct domain-invariant one-way permutations. This work extends this line of work by showing further limitations on black-box use of IO. The results of this work are incomparable to [AS15].

1.1.6 Organization.

We give an overview of the methodology and techniques used in the following Section 1.2. Chapter 2 provides required preliminaries. The black-box separation between SZK and IO (plus OWPs) is given in Chapter 3. The separation between $\mathsf{NP} \cap \mathsf{coNP}$ and IO (plus injective OWFs) is given in Chapter 4. Our construction of collision-resistant hashing functions from IO and SZK hardness is given in Chapter 5.

1.2 Overview of Techniques

We now give an overview of our approach and main ideas. We start by discussing how to capture black-box constructions in the context of indistinguishability obfuscation following [AS15]. We then recall the common methodology for ruling out black-box constructions [IR89, RTV04, BBF13], and explain the main ideas behind our impossibility results for SZK and $\mathsf{NP} \cap \mathsf{coNP}$. In the last part of this section, we outline the construction of collision-resistant hashing from indistinguishability obfuscation and SZK-hardness and the main ideas behind it.

1.2.1 Indistinguishability obfuscation and black-box constructions.

Traditionally, when thinking about a *black-box construction* of one cryptographic primitive $\mathcal{P}'$ (e.g., a pseudo-random generator) from a primitive $\mathcal{P}$ (e.g., a one-way function), we mean that all algorithms in the construction of $\mathcal{P}'$ invoke $\mathcal{P}$ as a black-box, obliviously of its actual implementation. This is hardly the case in constructions based on indistinguishability obfuscation where circuits that explicitly invoke the primitive $\mathcal{P}$ may be obfuscated.

Nonetheless, as observed by Asharov and Segev [AS15], in almost all existing constructions, the code implementing $\mathcal{P}$ is used in a very restricted manner. Typically, obfuscated circuits can be implemented as oracle aided circuits $C^{\mathcal{P}}$ that are completely black-box in $\mathcal{P}$, and $\mathcal{P}$ is some low-level primitive, such as a one-way function. Indeed, in most cases the circuits obfuscated invoke symmetric-key primitives, such as puncturable pseudo-random functions [SW14a], which can be constructed in a black-box way from one-way functions (in some constructions more structured low-level primitives may be used, like injective one-way functions, or one-way permutations). Furthermore, in these constructions, the obfuscator $i\mathcal{O}$ itself is also treated as a black-box.

Accordingly, almost all existing constructions based on indistinguishability obfuscation can be casted into a model in which indistinguishability obfuscation exists for oracle-aided circuits $C^{\mathcal{P}}$, where $\mathcal{P}$ is say a one-way functions, and both $\mathcal{P}$ and the obfuscator $i\mathcal{O}$ itself can only be accessed as black-boxes. On top of that, they can be proven secure in this model by a *black-box reduction* that makes black-box use of $(\mathcal{P}, i\mathcal{O})$ and any attacker against the constructed primitive $\mathcal{P}'$. Such constructions where both the construction itself and the reduction are black-box are called *fully black-box constructions* [RTV04]. Following Asharov and Segev [AS15, AS16], we shall prove our results in this model, ruling out black-box constructions of hard problems in SZK and $\mathsf{NP} \cap \mathsf{coNP}$ based on indistinguishability obfuscation for oracle-aided circuits. Further details follow.

1.2.2 Ruling out black-box constructions.

We prove our results in the model described above following the methodology of oracle separations (see e.g. [IR89, Sim98, RTV04, HR04]). Concretely, to prove that there is no fully black-box construction of a primitive $\mathcal{P}'$ from primitive $\mathcal{P}$, we demonstrate oracles $(\Psi, \mathcal{A})$ such that:

- relative to Ψ, there exists a construction $C^{\Psi}_{\mathcal{P}}$ realizing $\mathcal{P}$ that is secure in the presence of $\mathcal{A}$,
- but *any* construction $C^{\Psi}_{\mathcal{P}'}$ realizing $\mathcal{P}'$ can be broken in the presence of $\mathcal{A}$.

Indeed, if such oracles $(\Psi, \mathcal{A})$ exist, then no efficient reduction will be able to use (as a black-box) the attacker $\mathcal{A}$ against $\mathcal{P}'$ to break $\mathcal{P}$ (as the construction of $\mathcal{P}$ is secure in the presence of $\mathcal{A}$). In our case, we would like to apply this paradigm rule out black-box constructions of hard instances in either SZK or $\mathsf{NP} \cap \mathsf{coNP}$ from a low-level primitive (e.g. a one-way function) indistinguishability obfuscation for oracle-aided circuits. We next outline the main ideas behind the construction and analysis of the oracles $(\Psi, \mathcal{A})$ in each of the two cases.

1.2.3 Ruling our black-box constructions of hard SZK problems.

As explained in the previous section, we focus on the characterization of SZK by its complete statistical distance problem **SD** [SV03]. We demonstrate oracles $(\Psi, \mathcal{A})$ such that relative to Ψ there exist constructions of one-way permutations (OWPs) and IO for circuits with OWP gates, and these constructions are secure in the presence of $\mathcal{A}$. At the same time, $\mathcal{A}$ will decide (in the worst-case) $\mathbf{SD}^{\Psi}$. That is, $\mathcal{A}$ will decide instances (C^{Ψ}_0, C^{Ψ}_1) of circuit samplers that only use the IO and OWPs realized by Ψ in a black-box manner. We next explain how each of the two is constructed.

The construction of Ψ follows a general recipe suggested in [AS15, AS15]. The oracle consists of three parts $(f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$ where:

1. f is a random permutation, realizing the one-way permutation primitive.
2. $\mathcal{O}$ is a random injective function, realizing the obfuscation algorithm. It takes as input an oracle-aided circuit $C^{(\cdot)}$ along with randomness r and outputs an obfuscation $\widehat{C} = \mathcal{O}(C, r)$.
3. $\mathsf{Eval}^{\mathcal{O},f}$ realizes evaluation of obfuscated circuits. On input $(\widehat{C}, x)$, it inverts $\mathcal{O}$ to find (C, r), and outputs $C^f(x)$. If $\widehat{C}$ is not in the image of $\mathcal{O}$, it returns $\perp$.

The above construction readily satisfies the syntactic (or "functionality") requirements of one-way permutations and indistinguishability obfuscation. Furthermore, using standard techniques, it is not hard to show that relative to Ψ, the function f is one-way and $\mathcal{O}$ satisfies IO indistinguishability requirement. The challenge is to now come up with an oracle $\mathcal{A}$ that, on one hand, will decide $\mathbf{SD}^{\Psi}$, but on the other, will not compromise the security of the latter primitives.

Recall that deciding $\mathbf{SD}^{\Psi}$ means that given two oracle-aided circuit samplers (C_0, C_1) such that the statistical distance of the corresponding distributions $(\boldsymbol{C}^{\Psi}_0, \boldsymbol{C}^{\Psi}_1)$ is $s = \Delta(\boldsymbol{C}^{\Psi}_0, \boldsymbol{C}^{\Psi}_1) \in$

$[0, \frac{1}{3}] \cup [\frac{2}{3}, 1]$, the oracle $\mathcal{A}$ must decide in which of the two intervals s lies, whereas if the promise is not satisfied and $s \in (\frac{1}{3}, \frac{2}{3})$, there is no requirement whatsoever. With this in mind, a first naive attempt would be the following. $\mathcal{A}$ will have unbounded access to Ψ, give a query (C_0, C_1), it would compute compute $s = \Delta(\boldsymbol{C}_0, \boldsymbol{C}_1)$, and simply say whether $s < \frac{1}{2}$ or $s \geq \frac{1}{2}$. While such an oracle would definitely decide $\mathbf{SD}^\Psi$, it is not too hard to show that it is simply too powerful, and would not only break IO and OWPs, but would, in fact, allow solving any problem in NP^Ψ (or even in PP^Ψ). Other naive attempts such as refusing to answer outside the promise intervals, encounter a similar problem.

At high-level, the problem with such oracles is that solutions to hard problems can be easily correlated with "tiny" differences in the statistical distance of the two input circuits, whereas the above oracle may reflect tiny changes when the statistical distance is close to some threshold (1/2 in the above example) on which the oracle changes its behaviour. This motivates our actual definition of $\mathcal{A}$ as a *noisy oracle* that produces its answer, not according to some fixed threshold, but according to a random threshold, chosen afresh for each and every query. Concretely, the oracle, which we call StaDif^Ψ, for any query (C_0, C_1), chooses a uniformly random threshold $t \leftarrow (\frac{1}{3}, \frac{1}{3})$, and answers accordingly:

$$\mathsf{StaDif}^\Psi(C_0, C_1) = \begin{cases} Y & \text{if } s \geq t \text{ (far distributions)} \\ N & \text{if } s < t \text{ (similar distributions)} \end{cases} .$$

The main challenge in proving that the security of the IO and OWPs realized by $\mathcal{A}$ is not compromised by this oracle is that StaDif^Ψ has the power to query Ψ on exponentially many points in order to compute s. For instance, it may query Ψ on the preimage of a OWP challenge $f(x)$ or of a given obfuscation $\mathcal{O}(C, r)$. The key observation behind the proof is that the oracle's final answer still does not reflect how Ψ behaves locally on random points.

Intuitively, choosing the threshold t at random, for each query (C_0, C_1), guarantees that with high probability t is "far" from the corresponding statistical distance $s = \Delta(C_0^\Psi, C_1^\Psi)$. Thus, changing the oracle Ψ on, say, a single input x, such as the preimage of a OWP challenge $f(x)$, should not significantly change s and will not affect the oracle's answer; that is, unless the circuits query Ψ on x with high probability to begin with. We give a reduction showing that we can always assume that (C_0, C_1) are "smooth", in the sense that they do not make any specific query to Ψ with too high probability.

Following this intuition, we are able to show that through such local changes that go undetected by StaDif^Ψ, we can move to an ideal world where inverting the OWP or breaking IO can be easily shown to be impossible. We refer the reader to Section 3 for further details.

1.2.4 Ruling our black-box constructions of hard $\mathsf{NP} \cap \mathsf{coNP}$ problems.

As mentioned earlier, a fully black-box construction of hard problems in $\mathsf{NP} \cap \mathsf{coNP}$ is actually known assuming one-way permutations (OWPs), and cannot be ruled out as in the case of SZK. Instead, we rule out constructions from (non-surjective) injective one-way functions (IOWFs) and IO for circuits with IOWF gates. This generalizes several previous results by Rudich [Rud84], showing that OWFs do not give hardness in $\mathsf{NP} \cap \mathsf{coNP}$, by Matsuda and Matsuura [MM11], showing that IOWFs do not give OWPs (which are a special case of hardness $\mathsf{NP} \cap \mathsf{coNP}$), and by Asharov and Segev [AS16], showing that OWFs and IO for circuits with OWF gates do not give OWPs. In fact, our approach yields a new (and rather different) proof for each one of these results.

We follow a similar methodology to one we used for the case of SZK. That is, we would like to come up with oracles $(\Psi, \mathcal{A})$ such that Ψ realizes IOWFs and IO for circuits with IOWFs gates, which are both secure in the presence of $\mathcal{A}$, whereas black-box constructions of problems in $\mathsf{NP} \cap \mathsf{coNP}$ from these primitives can be easily solved by $\mathcal{A}$. Recall that by black-box constructions here we mean a pair of efficient oracle-aided non-deterministic verifiers V_0, V_1 that define co-languages $\overline{L}^{\Psi}, L^{\Psi}$ in $\mathsf{NP}^{\Psi} \cap \mathsf{coNP}^{\Psi}$ relative to the oracle Ψ implementing IOWFs and IO.

Similarly to previous works, we shall crucially rely on the fact that the construction, namely (V_0, V_1), must be "correct" (i.e., respect the $\mathsf{NP} \cap \mathsf{coNP}$ structure) for *any* oracle Ψ realizing the latter primitives. In particular, we observe that a construction where correctness is only guaranteed for particular (even if natural) oracles may definitely exist. This is for example the case if we only consider implementations of IO similar to those presented above in the context of SZK. Indeed, in that construction the implementation of IO has an additional property — it allows identifying invalid obfuscations (the Eval oracle would simply return $\perp$ on such obfuscations). This "verifiability" property coupled with the injectivity of obfuscators actually imply a hard problem in $\mathsf{NP} \cap \mathsf{coNP}$ in a black-box way.[1] Our separation thus leverages the fact that IO need not necessarily be verifiable, and rules out constructions that are required to be correct for any implementation of IO, even a non-verifiable one.

Accordingly, the oracles $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$ that we consider are a tweaked version of the oracles considered in the SZK case. Now f is a random injective function that is expanding, rather than a permutation, the oracle $\mathcal{O}$ is defined as before, and the oracle $\mathsf{Eval}^{f,\mathcal{O}}$ is defined as before for valid obfuscations $\widehat{C} \in \mathrm{Image}(\mathcal{O})$ *but is allowed to act arbitrarily for invalid obfuscations.* As for $\mathcal{A}$, this time it is trivially implemented by an oracle Decide^{Ψ} that, given input x, simply returns the unique bit b such that $V_b(x) = 1$, namely it just decides the corresponding language L^{Ψ}.[2]

In the results mentioned above [Rud84, MM11, AS16], it is actually shown that such an oracle can be completely simulated with a small number of queries to Ψ.[3] In our (more general) case, however, we were unable to show such a simulation process. Instead, we take a different approach inspired by our proof for the SZK setting described above. Roughly speaking, we show that somewhat similarly to our statistical distance oracle StaDif^{Ψ}, the oracle Decide^{Ψ} is also rather robust to random local changes. The main observation here is that for any fixed yes-instance $x \in L^{\Psi}$, tweaking Ψ at a random input into a new oracle Ψ', it is likely that x will still be a yes-instance in

---

[1] E.g. the language of all valid obfuscations and indices i, such that the ith bit of the obfuscated circuit is 1

[2] In the body, we further allow it to answer relative to other languages L' provided that they are indeed in $\mathsf{NP} \cap \mathsf{coNP}$. This allows us later to prove a more general oracle separation. See details in Section 4.6.

[3] More accurately, this is the case for Rudich's result for $\mathsf{NP} \cap \mathsf{coNP}$, whereas for the other results that rule out constructions of one-way permutations, once can simulate an analog of Decide that inverts the permutation.

$L^{\Psi'}$, as long as Ψ' is in our allowed family of oracles and $L^{\Psi'}$ is indeed in $\mathsf{NP}^{\Psi} \cap \mathsf{coNP}^{\Psi}$ (and the same is true for no-instances).

In slightly more detail, fixing a witness w such that $V_1^{\Psi}(x, w) = 1$, we can show that since V_1 makes a small number of oracle calls, with high probability tweaking the oracle Ψ at a random place will not affect these oracle calls and thus $V_1^{\Psi'}(x, w) = V_1^{\Psi}(x, w) = 1$. Then, assuming $L^{\Psi'}$ is guaranteed to be in $\mathsf{NP} \cap \mathsf{coNP}$, we can deduce that x must still a yes-instance (other witnesses for this fact may be added or disappear, but this does not change the oracle's answer). In the body, we argue that indeed $L^{\Psi'} \in \mathsf{NP}^{\Psi'} \cap \mathsf{coNP}^{\Psi'}$, where we strongly rely on the fact that arbitrary behavior of Eval is permitted on invalid obfuscations.

Once again, we show that through local changes that go undetected by Decide^{Ψ}, we can move to an ideal world where inverting the IOWF or breaking IO can be easily shown to be impossible. We refer the reader to Section 4 for further details.

1.2.5 Implied separations.

As a result of the two separations discussed above, we can rule out black-box constructions of hard problems in SZK or $\mathsf{NP} \cap \mathsf{coNP}$ from various cryptographic primitives or complexity classes. This essentially includes all primitives that have fully black-box constructions from OWPs (or IOWFs) and IO for circuits with OWP (or IWOF) gates. This includes public-key encryption, oblivious transfer, deniable encryption [SW14a], functional encryption [Wat15], delegation, [BGL+15, CHJV15, KLW15], hard (on-average) PPAD instances [BPR15], and more.

We note that there a few applications of IO that do not fall under this characterization. For instance, the construction of IO for Turing machines from IO-based succinct randomized encodings [BGL+15, CHJV15, KLW15] involves obfuscating a circuit that itself outputs (smaller) obfuscated circuits. To capture this, we would need to extend the above model to IO for circuits that can also make IO oracle calls (on smaller circuits). Another example is the construction of non-interactive witness indistinguishable proofs from IO [BP15]. There an obfuscated circuit may get as input another obfuscated circuit and would have to internally run it; furthermore, in this application, the code of the obfuscator is used in a (non-black-box) ZAP. Extending the above model to account for this type of IO applications is an interesting question that we leave for future exploration.

1.2.6 Full oracle separations.

As explained, the methodology we rely on rules out fully black-box constructions by exhibiting two oracles $(\Psi, \mathcal{A})$, the first which may be used by the construction of a primitive $\mathcal{P}'$ from $\mathcal{P}$, and the second which breaks $\mathcal{P}'$. In the literature (e.g., in [IR89, Sim98]), a stronger type of separation is often shown where a single oracle Γ is exhibited and can be fully accessed, not only by the adversary, but also by the construction (whereas above the construction can only access Ψ, but not $\mathcal{A}$). This rules out an even weaker type of reductions called *relativizing reductions* [RTV04], which guarantee that $\mathcal{P}'$ can be securely realized in any oracle world where $\mathcal{P}$ can. In the body, we show how to extend our result for $\mathsf{NP} \cap \mathsf{coNP}$ to also imply this stronger type of separation.

1.2.7 The positive result: collision-resistance from IO and SZK hardness.

We now described the main ideas behind our construction of collision-resistant hash functions. The starting for the construction is the work of Ishai, Kushilevitz, and Ostrovsky [IKO05] that shows how to construct collision-resistant hash functions from commitments that are additively homomorphic (for simplicity, say over $\mathbb{F}_2$). The idea is simple: we can hash ℓ bits to m bits, where m is the size of a single bit commitment and ℓ can be arbitrarily longer, as follows. The hash key is a commitment $\gamma := (\mathsf{com}(\beta_1), \ldots, \mathsf{com}(\beta_\ell))$ to a random vector $\beta \in \mathbb{F}_2^\ell$, and hashing $x \in \mathbb{F}_2^\ell$, is done by homomorphically computing a commitment to the inner product $\mathsf{CRH}_\gamma(x) = \mathsf{com}(\langle \beta, x \rangle)$. Intuitively, the reason this works is that any collision in CRH_γ reveals a vector that is orthogonal to β and thus leaks information about it and violating the hiding of the commitment.

At a high-level, we aim to mimic the above construction based on obfuscation. As a key for the collision-resistant hash we can obfuscate a program Π_β associated with a random vector β that given x outputs a commitment $\mathsf{com}(\langle \beta, x \rangle)$, where the commitment is derandomized using a PRF.[4] The obfuscation $i\mathcal{O}(\Pi_\beta)$ can be thought of as the commitment to β, and evaluating this program at x, corresponds to homomorphic evaluation. Despite the clear intuition behind this construction, it is not clear how to prove its security based on IO. In fact, by the work of Asharov and Segev [AS15], it cannot be proven based on a black-box reduction as long as plain statistically-binding commitments are used, as these can be constructed from OWPs in a fully black-box manner, and [AS15] rule out black-box constructions of collision-resistant hashing from OWPs and IO for circuits with OWP gates.

We show, however, that relying on a relaxed notion of perfectly-hiding commitments, as well as subexponential hardness of IO and puncturable PRFs, the construction can be proven secure. The perfect hiding of the commitment is leveraged in a probabilistic IO argument [CLTV15] that involves a number of hybrids larger than the overall number of commitments. We then observe that these relaxed commitments follow from average-case hardness of the polar statistical distance problem $\mathbf{SD}^{0,1}$.[5]

---

[4] In the body, we describe a slightly more abstract construction where inner product is replaced by an arbitrary 2-universal hash function.

[5] Similar SZK-hardness is known to imply statistically-hiding commitments against malicious receivers, but with a larger (constant) number of rounds [OV08].

Chapter 2

Preliminaries

In this section, we introduce the basic definitions and notation used throughout the paper.

2.1 Conventions

For a distribution D, we denote the process of sampling from D by $x \leftarrow D$. A function $\mathsf{negl} : \mathbb{N} \rightarrow \mathbb{R}^+$ is negligible if for every constant c, there exists a constant n_c such that for all $n > n_c$ $\mathsf{negl}(n) < n^{-c}$. We refer to uniform probabilistic polynomial-time algorithms as PPT algorithms.

Randomized Algorithms. As usual, for a random algorithm A, we denote by $A(x)$ the corresponding output distribution. When we want to be explicit about the algorithm using randomness r, we shall denote the corresponding output by $A(x; r)$.

Oracles. We consider *oracle-aided algorithms (or circuits)* that make repeated calls to an oracle Γ. Throughout, we will consider deterministic oracles Γ that are a-priori sampled from a distribution Γ on oracles. More generally, we consider infinite oracle ensembles $\Gamma = \{\Gamma_n\}_{n \in \mathbb{N}}$, one distribution Γ_n for each security parameter $n \in \mathbb{N}$ (each defined over a finite support). For example, we may consider an ensemble $f = \{f_n\}$ where each $f_n : \{0,1\}^n \rightarrow \{0,1\}^n$ is a random function. For such an ensemble Γ and an oracle aided algorithm (or circuit) A with finite running time, we will often abuse notation and denote by $A^\Gamma(x)$ and execution of A on input x where each of (finite number of) oracle calls that A makes is associated with a security parameter n and is answered by the corresponding oracle Γ_n. When we write $A_1^\Gamma, \ldots, A_k^\Gamma$ for k algorithms, we mean that they all access the same realization of Γ.

2.2 Indistinguishability Obfuscation for Oracle-Aided Circuits

The notion of *indistinguishability obfuscation* (IO) was introduced by Barak et al. [BGI+01] and the first candidate construction was demonstrated in the work of Garg et al. [GGH+13]. Since then, IO has given rise to a plethora of applications in cryptography and beyond. Nevertheless, Asharov and

Segev [AS15, AS16] demonstrated that IO is insufficient to achieve some cryptographic tasks, most notably (domain-invariant) one-way permutations, collision-resistant hashing, and as a corollary, private information retrieval and (even additively) homomorphic encryption. To formally show such a statement, they introduced the framework of oracle-aided indistinguishability obfuscation for oracle-aided circuits. We follow their framework.

We begin by recalling the notion of two oracle-aided circuits being equivalent, and move on to defining IO relative to oracles.

Definition 2.1. Let C_0 and C_1 be two oracle-aided circuits and let f be a function. C_0 and C_1 are said to be *functionally equivalent relative to* f, denoted as $C_0^f \equiv C_1^f$, if for every input x, $C_0^f(x) = C_1^f(x)$.

Definition 2.2. Let $\mathcal{C} = \{\mathcal{C}_n\}_{n\in\mathbb{N}}$ be a class of oracle aided circuits, where each $C \in \mathcal{C}_n$ is of size n.[1] A PPT algorithm $i\mathcal{O}$ is an *indistinguishability obfuscator* for $\mathcal{C}$ relative to an oracle distribution ensemble $\Gamma = \{\Gamma_n\}_{n\in\mathbb{N}}$ if the following conditions are met:

1. **Functionality.** For all $n \in \mathbb{N}$ and for all $C \in \mathcal{C}_n$ it holds that

$$\Pr_{\Gamma, i\mathcal{O}}\left[C^\Gamma \equiv \widehat{C}^\Gamma \mid \widehat{C} \leftarrow i\mathcal{O}^\Gamma(1^n, C)\right] = 1 \ .$$

2. **Indistinguishability.** For any non-uniform PPT distinguisher $D = (D_1, D_2)$ there exists a negligible function negl such that for all $n \in \mathbb{N}$

$$\mathsf{Adv}^{\mathsf{iO}}_{\Gamma, i\mathcal{O}, \mathcal{C}, D}(n) = \left|\Pr\left[\mathsf{Exp}^{\mathsf{iO}}_{\Gamma, i\mathcal{O}, \mathcal{C}, D}(n) = 1\right] - \frac{1}{2}\right| \leq \mathsf{negl}(n)$$

where the random variable $\mathsf{Exp}^{\mathsf{iO}}_{\Gamma, i\mathcal{O}, \mathcal{C}, D}(n)$ is defined via the following experiment:

(a) $b \leftarrow \{0, 1\}$.

(b) $(C_0, C_1, \mathsf{state}) \leftarrow D_1^\Gamma(1^n)$ where $C_0, C_1 \in \mathcal{C}_n$ and $C_0^\Gamma \equiv C_1^\Gamma$.

(c) $\widehat{C} \leftarrow i\mathcal{O}^\Gamma(1^n, C_b)$.

(d) $b' = D_2^\Gamma(\mathsf{state}, \widehat{C})$.

(e) If $b = b'$ output 1 else output 0.

We further say that $i\mathcal{O}$ satisfies δ-indistinguishability if the above negligible advantage is at most δ.

[1] As in [AS15], we assume throughout that the size of the obfuscated circuits equals the security parameter. This is only for simplicity of notation, and is without loss of generality as the circuits can be padded up if they are too small, and the security parameter can be polynomially increased if the circuits are too large.

2.3 Statistical Zero Knowledge

The notion of Statistical Zero Knowledge was introduced by Goldwasser, Micali and Rackoff [GMR85]. It consists of languages that can be proved using statistical zero knowledge proofs. Vadhan's thesis [Vad99] is an excellent exposition on the topic. We define them below.

Definition 2.3 (Statistical Zero Knowledge)**.** A language L is in SZK if there exists a tuple of functions (P, V, S) where the *verifier* V, and simulator S are computable by probabilistic polynomial time Turing machines satisfying the following:

- (P, V) is an interactive proof for L with negligible completeness and soundness errors.
- Let $(P, V)(x)$ denote the distribution of transcripts of the interaction between P and V on input x. For any $x \in L$,
$$\Delta(S(x), (P, V)(x)) < \mathsf{negl}(|x|)$$

This is actually the definition of Honest-verifier Statistical Zero Knowledge. From the work of Goldreich, Sahai and Vadhan [GSV98] we know that these two classes are equivalent. Sahai and Vadhan [SV03] showed that SZK has complete problems. For the purpose of this paper, it suffices to think of SZK as the class defined by its complete problem: Statistical Difference.

In the following definition, for a circuit C with n-bit input, we denote by $\boldsymbol{C}$ the output distribution $C(r)$ where $r \leftarrow \{0,1\}^n$. For two distributions $\mathbf{X}$ and $\mathbf{Y}$ we denote their statistical distance by $\Delta(\mathbf{X}, \mathbf{Y})$.

Definition 2.4 (Statistical difference)**.** Statistical difference is the promise problem $\mathbf{SD}^{\Psi} = (\mathbf{SD}^{\Psi}_{Y}, \mathbf{SD}^{\Psi}_{N})$, where

$$\mathbf{SD}_Y = \left\{ (C_0, C_1) \;\middle|\; \Delta(\boldsymbol{C}_0, \boldsymbol{C}_1) \geq \frac{2}{3} \right\} \;,$$
$$\mathbf{SD}_N = \left\{ (C_0, C_1) \;\middle|\; \Delta(\boldsymbol{C}_0, \boldsymbol{C}_1) \leq \frac{1}{3} \right\} \;.$$

Chapter 3

One-Way Permutations, Indistinguishability Obfuscation, and Hardness in Statistical Zero Knowledge

In this section, we ask which cryptographic primitives imply hardness in the class statistical zero-knowledge (SZK). Roughly speaking, we show that one-way permutations (OWPs) and indistinguishability obfuscation (IO), for circuits with OWP-gates, do not give rise to a black-box construction of hard problems in SZK. This, in turn implies that many cryptographic primitives (e.g., public-key encryption, functional encryption, and delegation), and hardness in certain low-level complexity classes (e.g. PPAD), also do not yield black-box constructions of hard problems in SZK.

We first motivate and define a framework of SZK relative to oracles, define fully black-box constructions of hard SZK problems, and then move on to the actual separation.

3.1 SZK and Statistical Distance

The notion of statistical zero-knowledge proofs was introduced in the seminal work of Goldwasser, Micali and Rackoff [GMR85]. The class of promise problems with statistical zero-knowledge proofs (SZK) can be characterized by several complete problems, such as *statistical distance* and *entropy difference* (see [Vad99] and references within). We shall focus on the characterization of SZK by the statistical distance problem. Here an instance is a pair of circuit samplers $C_0, C_1 : \{0,1\}^n \to \{0,1\}^m$ with the promise that the statistical distance $s = \Delta(\boldsymbol{C}_0, \boldsymbol{C}_1)$ of the corresponding distributions is either large (say, $s \geq 2/3$) or small (say, $s \leq 1/3$). The problem is to decide which is the case.

Hard statistical distance problems from cryptography: Motivation. SZK hardness, and in particular hard statistical distance problems, are known to follow from various number-theoretic and lattice problems that are commonly used in cryptography, such as Decision Diffie-Hellman, Quadratic Residuosity, and Learning with Errors. We ask more generally which cryptographic primitives can be shown to imply such hardness, with the intuition that such primitives are *structured* in a certain way. In particular, whereas one would not expect a completely unstructured object like one-way functions to imply such hardness, what can we say for instance about public-key encryption, or even indistinguishability obfuscation (which has proven to be structured enough to yield almost any known cryptographic goal).

We prove that none of these primitives imply such hardness through the natural class of black-box constructions and security reductions. To understand what a black-box construction of a hard statistical distance problem means, let us look at a specific example of the construction of such a problem from *rerandomizable encryption*. In a (say, symmetric-key) rerandomizable encryption scheme, on top of the usual encryption and decryption algorithms $(\mathsf{Enc}, \mathsf{Dec})$ there is a ciphertext rerandomization algorithm ReRand that can statistically refresh ciphertexts. Namely, for any ciphertext C encrypting a bit b, $\mathsf{ReRand}(\mathsf{C})$ produces a ciphertext that is statistically close to a fresh encryption $\mathsf{Enc}(b)$. Note that this immediately gives rise to a hard statistical distance problem: given a pair of ciphertexts $(\mathsf{C}, \mathsf{C}')$, decide whether the corresponding rerandomized distributions given by the circuits $(C_0(\cdot), C_1(\cdot)) := (\mathsf{ReRand}(\mathsf{C}; \cdot), \mathsf{ReRand}(\mathsf{C}'; \cdot))$ are statistically far or close. Indeed, this corresponds to whether they encrypt the same bit or not, which is hard to decide by the security of the encryption scheme.

A feature of this construction of hard statistical distance instances is that, similarly to most constructions in cryptography, it is *fully black-box* [RTV04] in the sense that the circuits C_0, C_1 only make black-box use of the encryption scheme's algorithms, and can in fact be represented as oracle-aided circuits $(C_0^{\mathsf{ReRand}(\cdot)}, C_1^{\mathsf{ReRand}(\cdot)})$. Furthermore, "hardness" can be shown by a black-box reduction that can use any decider for the problem in a black-box way to break the underlying encryption scheme. More generally, one can consider the statistical distance problem relative to different oracles implementing different cryptographic primitives and ask when can hardness be shown based on a black-box reduction. We will rule out such reductions relative to IO and OWPs (and everything that follows from these in a fully black-box way).

3.2 Fully Black-Box Constructions of Hard SD Problems from IO and OWPs

We start by defining statistical distance relative to oracles. In the following definition, for an oracle-aided (sampler) circuit $C^{(\cdot)}$ with n-bit input and an oracle Ψ, we denote by $\boldsymbol{C}^{\Psi}$ the output distribution $C^{\Psi}(r)$ where $r \leftarrow \{0,1\}^n$. For two distributions $\mathbf{X}$ and $\mathbf{Y}$ we denote their statistical distance by $\Delta(\mathbf{X}, \mathbf{Y})$.

Definition 3.1 (Statistical distance relative to oracles)**.** For an oracle Ψ, the statistical distance promise problem relative to Ψ, denoted as $\mathbf{SD}^{\Psi} = (\mathbf{SD}^{\Psi}_{Y}, \mathbf{SD}^{\Psi}_{N})$, is given by

$$\mathbf{SD}^{\Psi}_{Y} = \left\{(C_0, C_1) \;\middle|\; \Delta(\boldsymbol{C}^{\Psi}_0, \boldsymbol{C}^{\Psi}_1) \geq \frac{2}{3}\right\} \ ,$$
$$\mathbf{SD}^{\Psi}_{N} = \left\{(C_0, C_1) \;\middle|\; \Delta(\boldsymbol{C}^{\Psi}_0, \boldsymbol{C}^{\Psi}_1) \leq \frac{1}{3}\right\} \ .$$

We now formally define the class of constructions and reductions ruled out. That is, *fully black-box* constructions of hard statistical distance problems from OWPs and IO for OWP-aided circuits. The definition is similar in spirit to those in [AS15, AS16], adapted to our context of SZK-hardness.

Definition 3.2. A fully black-box construction of a hard statistical distance problem from OWPs and IO for the class $\mathcal{C}$ of circuits with OWP-gates consists of a collection of oracle-aided circuit pairs $\Pi = \left\{\Pi_n = \left\{(C_0, C_1) \in \{0,1\}^{n\times 2}\right\}\right\}_{n\in\mathbb{N}}$ and a probabilistic oracle-aided reduction $\mathcal{R}$ that satisfy:

1. **Non-Triviality:** Let f be any permutation and let $i\mathcal{O}$ be any function such that whenever $\widehat{C}^{(\cdot)} := i\mathcal{O}(C^{(\cdot)}, r)$, we have $\widehat{C}^f \equiv C^f$ for any $C^{(\cdot)}$ and r. Then

$$\Pi \cap \mathbf{SD}^{f,i\mathcal{O}}_{Y} \neq \emptyset \text{ and } \Pi \cap \mathbf{SD}^{f,i\mathcal{O}}_{N} \neq \emptyset \text{ (as per Definition 3.1)} \ .$$

2. **Black-box security proof:** There exist functions $q_{\mathcal{R}}(\cdot), \varepsilon_{\mathcal{R}}(\cdot)$ such that the following holds. Let f be any distribution on permutations and let $i\mathcal{O}$ be any distribution on functions such that $\widehat{C}^f \equiv C^f$ for any $C^{(\cdot)}$ and r, where $\widehat{C}^{(\cdot)} := i\mathcal{O}(C^{(\cdot)}, r)$. Then for any probabilistic oracle-aided $\mathcal{A}$ that *decides* Π *in the worst-case*, namely, for all $n \in \mathbb{N}$

$$\Pr_{f,i\mathcal{O},\mathcal{A}}\left[\mathcal{A}^{f,i\mathcal{O}}(C_0, C_1) = B \quad \text{for all} \quad \begin{array}{l}(C_0, C_1) \in \Pi_n, B \in \{Y, N\} \\ \text{such that } (C_0, C_1) \in \mathbf{SD}^{f,i\mathcal{O}}_{B}\end{array}\right] = 1$$

the reduction breaks either f *or* $i\mathcal{O}$, namely, for infinitely many $n \in \mathbb{N}$ either

$$\Pr_{\substack{x\leftarrow\{0,1\}^n \\ f,i\mathcal{O},\mathcal{A}}}\left[\mathcal{R}^{\mathcal{A},f,i\mathcal{O}}(f(x)) = x\right] \geq \varepsilon_{\mathcal{R}}(n) \ ,$$

or

$$\left|\Pr\left[\mathsf{Exp}^{\mathsf{iO}}_{(f,i\mathcal{O}),i\mathcal{O},\mathcal{C},\mathcal{R}^{\mathcal{A}}}(n) = 1\right] - \frac{1}{2}\right| \geq \varepsilon_{\mathcal{R}}(n) \ ,$$

where in both $\mathcal{R}$ makes at most $q_{\mathcal{R}}(n)$ queries to any of its oracles $(\mathcal{A}, f, i\mathcal{O})$, and any query $(C^{(\cdot)}_0, C^{(\cdot)}_1)$ it makes to $\mathcal{A}$ consists of circuits that also make at most $q_{\mathcal{R}}(n)$ queries to their oracles $(f, i\mathcal{O})$. The random variable $\mathsf{Exp}^{\mathsf{iO}}_{(f,i\mathcal{O}),i\mathcal{O},\mathcal{C},\mathcal{R}^{\mathcal{A}}}(n)$ represents the reductions winning probability in the IO security game (Definition 2.2) relative to $(f, i\mathcal{O})$.

We make several remarks about the definition:

- **Worst-case vs. average-case hardness.** In the above, we address *worst-case hardness*, in the sense that the reduction $\mathcal{R}$ breaks the underlying primitives only given a decider $\mathcal{A}$ that is always correct. One could further ask whether IO and OWPs even imply average-case hardness in SZK (as do many of the algebraic hardness assumptions in cryptography). Ruling out worst-case hardness (as we will do shortly) in particular rules out such average-case hardness.

- **IO for oracle-aided circuits.** Following [AS15, AS16], we consider indistinguishability obfuscation for oracle-aided circuits C^f that can make calls to the one-way permutation oracle. This model captures constructions where IO is applied to circuits that use pseudo-random generators, puncturable pseudo-random functions, or injective one-way functions as all of those have fully black-box constructions from one-way permutations (see further discussion in [AS15]). This includes almost all known constructions from IO, including public-key encryption, deniable encryption [SW14a], functional encryption [Wat15], delegation [BGL$^+$15, CHJV15, KLW15], and hard (on-average) PPAD instances [BPR15]. Accordingly, separating SZK from IO and OWPs in this model, results in a similar separation between SZK and any one of these primitives.

  We note that there a few applications though that do not fall under this model. The first is in applications where the obfuscated circuit might itself output (smaller) obfuscated circuit, for instance in the construction of IO for Turing machines from IO-based succinct randomized encodings [BGL$^+$15, CHJV15, KLW15]. To capture such applications, one would have to extend the model to also account for circuits with IO gates (and not only OWP gates). A second example is the construction of non-interactive witness indistinguishable proofs from IO [BP15]. There an obfuscated circuit may get as input another obfuscated circuit and would have to internally run it; furthermore, in this application, the code of the obfuscator is used in a (non-black-box) ZAP. Extending our results (and those of [AS15, AS16]) to these models is an interesting question, left for future work.

- **Security loss.** In the above definition the functions $q_{\mathcal{R}}$ and $\varepsilon_{\mathcal{R}}$ capture the *security loss* of the reduction. Most commonly in cryptography, the query complexity is polynomial $q_{\mathcal{R}}(n) = n^{O(1)}$ and the probability of breaking the underlying primitive is inverse polynomial $\varepsilon_{\mathcal{R}}(n) = n^{-O(1)}$. Our lower-bounds will in-fact apply for *exponential* $q_{\mathcal{R}}, \varepsilon_{\mathcal{R}}^{-1}$. This allows capturing also constructions that rely on sub-exponentially secure primitives (e.g., [BGL$^+$15, CHJV15, KLW15, BPR15, BPW16]).

Ruling out fully black-box constructions: a road map. Our main result in this section is that fully black-box constructions of a hard statistical distance problem from IO and OWPs do not exist. Furthermore, this holds even if the latter primitives are exponentially secure.

Theorem 3.3. *Any fully black-box construction of a statistical distance problem* Π *from OWPs and IO for circuits with OWP gates has an exponential security loss:* $\max(q_{\mathcal{R}}(n), \varepsilon_{\mathcal{R}}^{-1}(n)) \geq \Omega(2^{n/12})$.

The proof of the theorem follows a common methodology (applied for instance in [HR04, HHRS15b, AS15]). We exhibit two (distributions on) oracles $(\Psi, \mathsf{StaDif}^{\Psi})$, where Ψ realizes OWPs and IO for circuits with OWP gates, and StaDif^{Ψ} that decides $\mathbf{SD}^{\Psi}$, the statistical distance problem relative to Ψ, in the worst case. We then show that the primitives realized by Ψ are (exponentially) secure even in the presence of StaDif^{Ψ}. Then viewing StaDif as a worst-case decider $\mathcal{A}$ (as per Definition 3.2) directly implies Theorem 3.3 establishing ruling out fully black-box constructions with a subexponential security loss.

The rest of this section is organized according to the above plan. First, in Section 3.3, we describe the oracle StaDif^{Ψ} (which is independent of the specific way that Ψ realizes IO and OWPs). Then, in Sections 3.4 and 3.5, we describe the oracle Ψ realizing OWPs and IO and prove its (exponential) security in the presence of StaDif^{Ψ}.

3.3 A Noisy Statistical-Distance Oracle

We now define the oracle StaDif^{Ψ} that will solve the statistical distance problem $\mathbf{SD}^{\Psi}$ in all the separations proved in this section. Our goal is to design StaDif^{Ψ} in a way that will not break the security of the cryptographic primitives realized by Ψ (OWPs in the warmups, and then OWPs and IO for circuits with OWP-gates). For this purpose, in our definition of the oracle StaDif^{Ψ}, we will try to exploit the fact that statistical distance is insensitive to *local changes* in the input distributions. Then, we will show that breaking the relevant cryptographic primitives, captured by Ψ, is impossible without detecting such local changes.

The concrete way of capturing the spoken insensitivity will be to define a "noisy oracle" that would be correct on distribution pairs whose distance is within the promise range $\left[0, \frac{1}{3}\right] \cup \left[\frac{2}{3}, 1\right]$, but would behave randomly within $\left(\frac{1}{3}, \frac{2}{3}\right)$.

Definition 3.4 (Oracle StaDif^{Ψ})**.** The oracle consists of $\mathbf{t} = \{\mathbf{t}_n\}_{n \in \mathbb{N}}$ where $\mathbf{t}_n : \{0,1\}^{2n} \to \left(\frac{1}{3}, \frac{2}{3}\right)$ is a uniformly random function. Given oracle-aided circuits $(C_0, C_1) \in \{0,1\}^n$, let $t = \mathbf{t}_n(C_0, C_1)$, and let $s = \Delta(\boldsymbol{C}_0^{\Psi}, \boldsymbol{C}_1^{\Psi})$, return

$$\mathsf{StaDif}^{\Psi}(C_1, C_2; t) := \begin{cases} N & \text{If } s < t \\ Y & \text{If } s \geq t \end{cases}$$

It is immediate to see that StaDif^{Ψ} decides $\mathbf{SD}^{\Psi}$ in the worst-case. The main challenge is in showing that Ψ can implement OWPs and IO (for OWP-aided circuits) that will be secure in the presence of StaDif^{Ψ}. We next develop the terminology and establish several useful properties of StaDif that will allow us to carry out the above plan.

Capturing insensitivity to local changes. We introduce two general notions of *farness* and *smoothness* that aim to capture the sense in which the statistical difference oracle StaDif^{Ψ} defined above is insensitive to local changes.

Roughly speaking *farness* says that the random threshold t used for a query (C_0, C_1) to StaDif^Ψ is "far" from the actual statistical distance. We will show that with high probability over the choice of random threshold $\mathbf{t}$, farness holds for all queries (C_0, C_1) made to StaDif^Ψ by any (relatively) efficient adversary. This intuitively means that changing the distributions $(\boldsymbol{C}_0^\Psi, \boldsymbol{C}_1^\Psi)$, on sets of small density, will not change the oracle's answer.

Definition 3.5 (Farness)**.** The oracles $(\Psi, \mathsf{StaDif}^\Psi)$ satisfy δ-*farness* with respect to oracle-aided circuits $(C_0, C_1) \in \{0,1\}^n$ if the statistical difference $s = \Delta(\boldsymbol{C}_0^\Psi, \boldsymbol{C}_1^\Psi)$ and the threshold $t = \mathbf{t}_n(C_0, C_1)$ sampled by StaDif are δ-far:

$$|s - t| \geq \delta \ .$$

For an adversary $\mathcal{A}$, we denote by $\mathbf{Far}(\mathcal{A}, \Psi, \delta)$ the event that $\Gamma = (\Psi, \mathsf{StaDif}^\Psi)$ satisfies δ-farness for all queries (C_0, C_1) made by $\mathcal{A}$ to StaDif^Ψ.

Claim 3.6. *Fix any Ψ and any oracle-aided adversary $\mathcal{A}$ such that $\mathcal{A}^{\Psi, \mathsf{StaDif}^\Psi}$ makes at most q queries to StaDif^Ψ. Then*

$$\Pr_{\mathbf{t}}\left[\mathbf{Far}(\mathcal{A}, \Psi, \delta)\right] \geq 1 - 6\delta q \ ,$$

where the probability is over the choice $\mathbf{t}$ of random thresholds by StaDif.

Proof. This follows from the fact that, for any query (C_0, C_1) to StaDif^Ψ with $s = \Delta(\boldsymbol{C}_0^\Psi, \boldsymbol{C}_1^\Psi)$, δ-farness does not hold only if the threshold $t = \mathbf{t}(C_0, C_1)$, chosen at random for this query, happens to be in the interval $(s - \delta, s + \delta)$, which occurs with probability at most $|(s-\delta, s+\delta)| / \left|\left(\frac{1}{3}, \frac{2}{3}\right)\right| = 6\delta$. The lemma then follows by a union bound over at most q queries. ∎

We now turn to define the notion of *smoothness*. Roughly speaking we will say that an oracle-aided circuit C is smooth with respect to some oracle Ψ if any specific oracle query is only made with small probability. In particular, for a pair of smooth circuits (C_0, C_1), local changes to the oracle Ψ should not change significantly the statistical distance $s = \Delta(\boldsymbol{C}_0^\Psi, \boldsymbol{C}_1^\Psi)$.

Definition 3.7 ((Ψ, δ)-Smoothness)**.** An oracle-aided circuit $C^{(\cdot)} : \{0,1\}^n \to \{0,1\}^m$ is said to be (Ψ, δ)-smooth if for all $x \in \{0,1\}^\star$,

$$\Pr_{r \leftarrow \{0,1\}^n}\left[C^\Psi(r) \text{ queries } \Psi \text{ at } x\right] \leq \delta \ .$$

For an adversary $\mathcal{A}$, we denote by $\mathbf{Smo}(\mathcal{A}, \Psi, \delta)$ the event that all queries (C_0, C_1) made by $\mathcal{A}$ to StaDif^Ψ are (Ψ, δ)-smooth.

Claim 3.8. *Let Ψ, Ψ' be oracles that differ on at most c values in the domain. Let (C_0, C_1) be (Ψ, δ)-smooth. Let $s = \Delta(C_0^\Psi, C_1^\Psi)$ and $s' = \Delta(C_0^{\Psi'}, C_1^{\Psi'})$ then $|s - s'| \leq 2c\delta$.*

Proof. For either $b \in \{0,1\}$,

$$\begin{aligned}
&\Delta(C_b^{\Psi}, C_b^{\Psi'}) \leq \\
&\Pr_r\left[C_b^{\Psi}(r) \neq C_b^{\Psi'}(r)\right] \leq \\
&\Pr_r\left[C_b^{\Psi}(r) \text{ queries } \Psi \text{ at } x \text{ where } \Psi(x) \neq \Psi'(x)\right] \leq \\
\sum_{x:\Psi(x)\neq\Psi'(x)} &\Pr_r\left[C_b^{\Psi}(r) \text{ queries } \Psi \text{ at } x\right] \leq c \cdot \delta \ .
\end{aligned}$$

The claim then follows by the fact that

$$|s - s'| := \left|\Delta(C_0^{\Psi}, C_1^{\Psi}) - \Delta(C_0^{\Psi'}, C_1^{\Psi'})\right| \leq \Delta(C_0^{\Psi}, C_0^{\Psi'}) + \Delta(C_1^{\Psi}, C_1^{\Psi'}) \leq 2c\delta \ .$$

□

The above roughly means that (under the likely event that farness holds) making smooth queries should not help the adversary detect local changes in the oracle Ψ. We will next show that, in fact, we can always "smoothen" the adversary's circuit at the expense of making (a few) more queries to Ψ, which intuitively deems the statistical distance oracle StaDif^{Ψ} useless altogether for detecting local changes in Ψ. Looking ahead, we will later show that breaking certain cryptographic primitives (OWPs and IO) is impossible without detecting such local changes, and then deduce that they do not break in the presence of StaDif^{Ψ}.

In what follows, we say that an adversary $\mathcal{A}$ is q-query if $\mathcal{A}^{\Psi,\mathsf{StaDif}^{\Psi}}$ makes at most q queries to Ψ and q queries to StaDif^{Ψ}, and any query made to StaDif^{Ψ} consist of oracle-aided circuits (C_0, C_1) that make at most q queries to Ψ, on any specific input. (We do not restrict the size of these circuits, but only the number of queries they make.)

Lemma 3.9 (Smoothing Lemma)**.** *For any q-query algorithm $\mathcal{A}$ and $\beta \in \mathbb{N}$, there exists a $(q+2\beta q)$-query algorithm $\mathcal{S}$ such that for any input $z \in \{0,1\}^\star$ and oracles $\Psi, \mathsf{StaDif}^{\Psi}$:*

1. *$\mathcal{S}^{\Psi,\mathsf{StaDif}^{\Psi}}(z)$ perfectly simulates the view of $\mathcal{A}^{\Psi,\mathsf{StaDif}^{\Psi}}(z)$,*
2. *$\mathcal{S}^{\Psi,\mathsf{StaDif}^{\Psi}}(z)$ only makes (Ψ,δ)-smooth queries to StaDif^{Ψ} with probability:*

$$\Pr_{\mathcal{S}}\left[\mathbf{Smo}(\mathcal{S},\Psi,\delta)\right] \geq 1 - 2^{-\delta\beta + \log(2q^2/\delta)} \ ,$$

over its own random coin tosses.

Proof. The simulator $\mathcal{S}$ emulates $\mathcal{A}$ and whenever $\mathcal{A}$ makes a query (C_0, C_1) to Ψ, $\mathcal{A}$ first evaluates each of the two circuits C_0^{Ψ}, C_1^{Ψ} on β random inputs and stores all the queries they make to Ψ along with their answers in a table T. It then generates a new query consisting of circuits (C_0', C_1') that have the table T hardwired in them. Each C_b' emulates C_b, but whenever the emulated C_b makes an oracle query to Ψ, C_b' first tries to answer using the table T, and only if the answer is not there turns to the oracle Ψ.

By construction, $\mathcal{S}$ perfectly emulates the view of $\mathcal{A}$. We now bound the probability that $\mathcal{S}$ generates a circuit that is not (Ψ, δ)-smooth. Fix any query (C_0, C_1) and let x be a *heavy query* in the sense that it is queried with probability larger than δ by one of the two circuits. Then the query x will be put in the table T except with probability $(1-\delta)^\beta \leq 2^{-\delta\beta}$. Furthermore, each one of the two circuits makes at most q oracle queries and thus each has at most q/δ inputs x as above. The claim now follows by a union bound over at most q queries (C_0, C_1) and at most q/δ heavy inputs that each of the two has. □

3.4 Warmup: One-Way Permutations in the Presence of StaDif

In this section, we show that a random permutation f is hard to invert even given access to the noisy statistical distance oracle StaDif^f. We start by defining the oracle. In what follows, $\mathbf{P}_n$ denotes the set of permutations of $\{0,1\}^n$.

Definition 3.10 (The Oracle f)**.** $f = \{f_n\}_{n\in\mathbb{N}}$ on input $x \in \{0,1\}^n$ answers with $f_n(x)$ where f_n is a random permutation $f_n \leftarrow \mathbf{P}_n$.

Our main theorem states that f cannot be inverted, except with exponentially small probability, even given an exponential number of oracle queries to f and StaDif^f. Here, consistently with the previous subsection, we say that an adversary $\mathcal{A}$ is q-query if $\mathcal{A}^{\Psi,\mathsf{StaDif}^\Psi}$ makes at most q queries to f and q queries to StaDif^f, and any query made to StaDif^f consists of oracle-aided circuits (C_0, C_1) that make at most q queries to f, on any specific input.

Theorem 3.11. *Let $q \leq O(2^{n/6})$. Then for any q-query adversary $\mathcal{A}$*

$$\Pr_{f,\mathsf{StaDif},x}\left[\mathcal{A}^{f,\mathsf{StaDif}^f}(f(x)) = x\right] \leq O(2^{-n/6}) \ ,$$

where the probability is over the random choices of f, StaDif and $x \leftarrow \{0,1\}^n$.

At a very high level, the proof of the theorem follows the plan outlined above, showing that in order to invert a random permutation the adversary must be able to detect certain local changes to the permutation, which the noisy statistical distance oracle is insensitive to.

Proof. We, in fact, prove a stronger statement: the above holds when fixing the oracles $f_{-n} := \{f_k\}_{k\neq n}$. For simplicity, we will suppress oracle access to the fixed f_{-n} in our notation and only denote the oracle f_n. Fix a q-query adversary $\mathcal{A}$ and let $\mathcal{S}$ be its smooth $(q+2\beta q)$-query simulator given by Lemma 3.9, where β will be specified later on. Since $\mathcal{S}$ perfectly emulates $\mathcal{A}$, it is enough to bound the probability that $\mathcal{S}$ successfully inverts. To bound $\mathcal{S}$'s inversion probability, we consider four hybrid experiments $\{\mathbf{H_i}\}_{i\in[4]}$ given in Table 3.1. Throughout, for a permutation $f \in \mathbf{P}_n$ and $x, y \in \{0,1\}^n$, we denote by $f_{x\mapsto y}$ the function that maps x to y and is identical to f on all other inputs (in particular, $f_{x\mapsto y}$ is no longer a permutation when $x \neq f^{-1}(y)$).

Hybrid	$\mathbf{H_1}$ (Real)	$\mathbf{H_2}$	$\mathbf{H_3}$	$\mathbf{H_4}$ (Ideal)
Permutation	$f \leftarrow \mathbf{P}_n$			
Preimage	$x \leftarrow \{0,1\}^n$			
2nd Preimage	$z \leftarrow \{0,1\}^n$			
Planted Image	$y \leftarrow \{0,1\}^n$			
Challenge	$f(x)$	$f(x)$	y	y
Oracle	f, StaDif^f	$f_{z\mapsto f(x)}, \mathsf{StaDif}^{f_{z\mapsto f(x)}}$	$f_{x\mapsto y}, \mathsf{StaDif}^{f_{x\mapsto y}}$	f, StaDif^f
Winning Condition	Find x			

Table 3.1: The hybrid experiments.

Hybrid $\mathbf{H_1}$ is identical to the real world where $\mathcal{S}$ wins if it successfully inverts the permutation at a random output. We show that the probability that the simulator wins in any of the experiments is roughly the same, and that in hybrid $\mathbf{H_4}$ the probability that $\mathcal{S}$ wins is tiny.

Claim 3.12. $|\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_1}] - \Pr[\mathcal{S} \textit{ wins in } \mathbf{H_2}]| \le O(2^{-n/6})$

Proof. The difference between the two hybrids is in the oracle that $\mathcal{S}$ is given: simply f in the first, and its slightly tweaked version $f_{z\mapsto f(x)}$ in the second. We can bound the difference between the winning probabilities in $\mathbf{H_1}$ and $\mathbf{H_2}$ as follows:

$$|\Pr[\mathcal{S} \text{ wins in } \mathbf{H_1}] - \Pr[\mathcal{S} \text{ wins in } \mathbf{H_2}]| \le \Pr_{\substack{\mathcal{S},x,z\\ f,\mathsf{StaDif}}}\left[\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x)) \neq \mathcal{S}^{f_{z\mapsto f(x)},\mathsf{StaDif}^{f_{z\mapsto f(x)}}}(f(x))\right] ,$$

where the probability is over the coins of $\mathcal{S}$ and StaDif and the choice of $x, z \leftarrow \{0,1\}^n, f \leftarrow \mathbf{P}_n$.

In what follows, we denote by $\mathbf{Hit} = \mathbf{Hit}(\mathcal{S}, f, x, z)$ the event that $\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x))$ queries f on z. Also, let $\mathbf{Far} = \mathbf{Far}(\mathcal{S}(f(x)), f, 2\delta)$ be the event that 2δ-farness holds for all StaDif-queries made by $\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x))$ (Definition 3.7), and $\mathbf{Smo} = \mathbf{Smo}(\mathcal{S}(f(x)), f, \delta)$ is the event that all StaDif-queries made by $\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x))$ are (f, δ)-smooth (Definition 3.7).

We now claim

Claim 3.13. *For any* $\delta < 1$,

$$\Pr_{\substack{\mathcal{S},x,z\\ f,\mathsf{StaDif}}}\left[\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x)) \neq \mathcal{S}^{f_{z\mapsto f(x)},\mathsf{StaDif}^{f_{z\mapsto f(x)}}}(f(x))\right] \le \Pr_{\substack{\mathcal{S},x,z\\ f,\mathsf{StaDif}}}\left[\mathbf{Hit} \vee \overline{\mathbf{Far}} \vee \overline{\mathbf{Smo}}\right] .$$

Proof. We argue that whenever the complement $\overline{\mathbf{Hit}} \wedge \mathbf{Far} \wedge \mathbf{Smo}$ occurs then

$$\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x)) = \mathcal{S}^{f_{z\mapsto f(x)},\mathsf{StaDif}^{f_{z\mapsto f(x)}}}(f(x)) .$$

Indeed, for any StaDif-query (C_0, C_1) made by $\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x))$, we know by (f, δ)-smoothness that changing f at one point does not affect the statistical distance by much. Concretely, by Claim 3.8:

$$\left|\Delta(\boldsymbol{C}_0^f, \boldsymbol{C}_1^f) - \Delta(\boldsymbol{C}_0^{f_{z\mapsto f(x)}}, \boldsymbol{C}_1^{f_{z\mapsto f(x)}})\right| \leq 2\delta \ .$$

Furthermore, if 2δ-farness also holds for any such query (for some threshold $\mathbf{t}$ sampled by StaDif), then

$$\mathsf{StaDif}^f(C_0, C_1; \mathbf{t}) = \mathsf{StaDif}^{f_{z\mapsto f(x)}}(C_0, C_1; \mathbf{t}) \ .$$

If in addition **Hit** does not occur, then for any f-query w made by $\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x))$,

$$f(w) = f_{z\mapsto f(x)}(w) \ .$$

It follows that the views of $\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x))$ and $\mathcal{S}^{f_{z\mapsto f(x)},\mathsf{StaDif}^{f_{z\mapsto f(x)}}}(f(x))$ are identical. $\square$

It is left to bound the probability of each of the events $\mathbf{Hit}, \overline{\mathbf{Far}}, \overline{\mathbf{Smo}}$. First, noting that the view of $\mathcal{S}^{f,\mathsf{StaDif}^f}(f(x))$ is independent of the random z, we can bound

$$\Pr[\mathbf{Hit}] \leq 2^{-n} \cdot \#\{f\text{-queries made by } \mathcal{S}\} \leq 2^{-n} \cdot (q + 2\beta q) \ .$$

Furthermore, by the farness Claim 3.6 and smoothing Lemma 3.9

$$\Pr\left[\overline{\mathbf{Far}}\right] \leq 12q\delta \ .$$
$$\Pr\left[\overline{\mathbf{Smo}}\right] \leq 2^{-\delta\beta + \log(2q^2/\delta)} \ ,$$

Overall we can bound the difference between $\mathbf{H_1}$ and $\mathbf{H_2}$ by

$$2^{-\delta\beta+\log(2q^2/\delta)} + 2^{-n} \cdot (q + 2\beta q) + 12q\delta \leq O(2^{-n/6}) \ ,$$

when setting $\delta = 2^{-n/3}, \beta = 2^{n/3} \cdot n$, and recalling that $q \leq O(2^{n/6})$. $\square$

Claim 3.14. $\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_2}] = \Pr[\mathcal{S} \textit{ wins in } \mathbf{H_3}]$.

Proof. The difference between $\mathbf{H_2}$ and $\mathbf{H_3}$ is in the input of $\mathcal{S}$, $f(x)$ in the first and a random y in the second, and in the oracle $\mathcal{S}$ is given, $f_{z\mapsto f(x)}$ in the first and $f_{x\mapsto y}$ in the second. We argue, however, that the distribution $\left\{(f(x), f_{z\mapsto f(x)}, x) \;\middle|\; f \leftarrow \mathbf{P}_n, x, z \leftarrow \{0,1\}^n\right\}$ in $\mathbf{H_1}$ is identical to that of $\{(y, f_{x\mapsto y}, x) \mid f \leftarrow \mathbf{P}_n, x, z \leftarrow \{0,1\}^n\}$ are in $\mathbf{H_2}$. Indeed, in $\mathbf{H_1}$, $(f(x), x)$ are distributed uniformly and independently just as (y, x) in $\mathbf{H_2}$. Then, conditioned on any (y, x), the oracle in both distribution can be sampled as a random permutation f conditioned on $y = f(x)$ and diverting a random z from $f(z)$ to y. $\square$

Claim 3.15. $|\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_3}] - \Pr[\mathcal{S} \textit{ wins in } \mathbf{H_4}]| \leq O(2^{-n/6})$.

The difference between the two hybrids is in the oracle that $\mathcal{S}$ is given: simply f in the second and its slightly tweaked version $f_{x \mapsto y}$ in the first. The proof of their indistinguishability is essentially identical to that of Claim 3.12 and is omitted.

To conclude the proof of Theorem 3.11, we observe that

Claim 3.16. $\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_4}] \leq 2^{-n}$.

Proof. The view of $\mathcal{S}$ in this hybrid is completely independent of the random choice of x. □

□

3.5 Indistinguishability Obfuscation (and OWPs) in the Presence of StaDif

In this section, we consider an oracle Ψ that realizes both indistinguishability obfuscation (IO) and one-way permutations (OWPs) and show that neither break in the presence of the noisy statistical distance oracle StaDif^{Ψ}. We start by defining the oracle Ψ. In a nutshell, the oracle realizes OWPs through a random permutation oracle. IO for circuits with OWP-gates is captured in a similar way to [AS15] by a random injective mapping coupled with a corresponding evaluation algorithm.

In what follows, $\mathbf{P}_n$ denotes the set of permutations of $\{0,1\}^n$, $\mathbf{F}_n^m$ denotes the set of functions mapping $\{0,1\}^n$ to $\{0,1\}^m$, and $\mathbf{I}_n^m$ denotes the set of injective functions mapping $\{0,1\}^n$ to $\{0,1\}^m$.

Definition 3.17 (The Oracle Ψ)**.** The oracle $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$ consists of three parts:

- $f = \{f_n\}_{n \in \mathbb{N}}$ on input $x \in \{0,1\}^n$ answers with $f_n(x)$, where f_n is a random permutation $f_n \leftarrow \mathbf{P}_n$.
- $\mathcal{O} = \{\mathcal{O}_n\}_{n \in \mathbb{N}}$ on input $(C, r) \in \{0,1\}^n \times \{0,1\}^n$ answers with $\widehat{C} := \mathcal{O}_n(C, r)$ where $\mathcal{O}_n$ is a random injective function $\mathcal{O}_n \leftarrow \mathbf{I}_{2n}^{5n}$ into $\{0,1\}^{5n}$.
- $\mathsf{Eval}^{f,\mathcal{O}}$ given $\widehat{C} \in \{0,1\}^{5n \times 2}, x \in \{0,1\}^{\star}$ computes $(C, r) = \mathcal{O}_n^{-1}(\widehat{C})$, interprets C as an oracle-aided circuit, and returns $C^f(x)$. If $\widehat{C}$ does not have a unique preimage, or the input size of C is inconsistent with $|x|$, the oracle returns $\perp$.

In the next two subsections, we show that the oracle Ψ securely realizes OWPs and IO in the presence of the noisy statistical distance oracle StaDif^{Ψ}. Throughout, we address adversaries with oracles $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{\mathcal{O},f})$ and StaDif^{Ψ}. We will say that such an adversary is q-query if they

1. make only q queries to f,
2. make only q queries to either $\mathcal{O}$ or Eval, and for any query (C, r) made to $\mathcal{O}$, C is an f-aided circuit that makes at most q queries to f,
3. make only q queries to StaDif^{Ψ}, and for any query (C_0, C_1) made to StaDif^{Ψ}, (C_0, C_1) are Ψ-aided and each of them is q-query (according to the two conditions above).

3.5.1 One-Way Permutations

We show that f cannot be inverted, except with exponentially small probability even given an exponential number of oracle queries to $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{\mathcal{O},f})$ and StaDif^{Ψ}.

Theorem 3.18. *Let $q(n) \leq O(2^{n/12})$. Then for any q-query adversary $\mathcal{A}$*

$$\Pr_{\substack{\Psi=(f,\mathcal{O},\mathsf{Eval})\\ \mathsf{StaDif},x}}\left[\mathcal{A}^{\Psi,\mathsf{StaDif}^{\Psi}}(f(x)) = x\right] \leq O(2^{-n/6}) \ ,$$

where the probability is over the random choice of Ψ, StaDif and $x \leftarrow \{0,1\}^n$.

Proof. We will, in fact, prove a stronger statement: the above holds when fixing the oracles $f_{-n} := \{f_k\}_{k\neq n}$, $\mathcal{O} = \{\mathcal{O}_n\}_{n\in\mathbb{N}}$. We prove the theorem by a reduction to the case that Ψ only consists of the permutation f (and does not include $\mathcal{O}, \mathsf{Eval}$). Concretely, fix any q-query adversary $\mathcal{A}$ that inverts the random permutation f_n given access to $\Psi = (f, \mathcal{O}, \mathsf{Eval})$ and StaDif^{Ψ}, we show how to reduce it to a q^2-query adversary $\mathcal{B}^f(f_n(x))$ that inverts f_n for a random $x \leftarrow \{0,1\}^n$ with the same probability as $\mathcal{A}$. The proof then follows from Theorem 3.11.

The new adversary $\mathcal{B}^{f,\mathsf{StaDif}^f}(f_n(x))$ emulates $\mathcal{A}^{\Psi,\mathsf{StaDif}^{\Psi}}(f_n(x))$ answering Ψ-queries as follows:

- f **queries:** answered according to $\mathcal{B}$'s oracle f. This translates to at most q queries to f.
- $\mathcal{O}$ **queries:** answered according to the fixed oracle $\mathcal{O}$. This does not add any calls to f.
- $\mathsf{Eval}^{f,\mathcal{O}}$ **queries:** given query $\mathsf{Eval}(\widehat{C}, x)$, invert the fixed oracle $\mathcal{O}$ to find $(C, r) = \mathcal{O}^{-1}(\widehat{C})$. If no such preimage exists, return $\perp$. If a preimage does exist, using the f-oracle, compute $C^f(x)$ and return the result. This translates to at most q^2 queries to f: q queries by C, for each of the q queries $\widehat{C}$ to Eval.
- StaDif^{Ψ} **queries:** given query (C_0, C_1), where C_b makes Ψ-queries translate to D_0, D_1 that only make f-queries, where each query to $\Psi = (f, \mathcal{O}, \mathsf{Eval})$ is translated to a query to f according to the previous three items. The resulting oracle-aided (D_0, D_1) may thus make up to $q + q^2$ queries f: q corresponding to the first item, and q^2 corresponding to the third.[1]

Overall $\mathcal{B}^f$ is $O(q^2)$-query and perfectly emulates the view of $\mathcal{A}^{\Psi}$. The theorem now follows from Theorem 3.11.

□

3.5.2 Indistinguishability Obfuscation

We now turn to show that Ψ also realizes an indistinguishability obfuscator that does not break in the presence of StaDif^{Ψ}. We start by describing the construction, which is similar to the one in [AS15].

[1] We note that while there is a bound on the number of queries that they make, we do not put any restrictions on their size, which allows to hardwire the fixed $\mathcal{O}$ and f_{-n} as required in the previous three items. Indeed, Theorem 3.11 does not put any restriction on the size of these circuits.

Construction 3.19 (The Obfuscator $i\mathcal{O}^{\Psi}$). *Let $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$. Given an oracle-aided circuit $C \in \{0,1\}^n$, $i\mathcal{O}^{\Psi}(1^n, C)$ samples a random $r \leftarrow \{0,1\}^n$, computes $\widehat{C} = \mathcal{O}(C, r)$, and returns an oracle aided circuit $E_{\widehat{C}}$ that given input x, computes $\mathsf{Eval}^{f,\mathcal{O}}(\widehat{C}, x)$.*

It is easy to see that $i\mathcal{O}^{f,\mathcal{O},\mathsf{Eval}}$ satisfies the functionality requirement of Definition 2.2 for the class $\mathcal{C}$ of f-aided circuits; indeed, this follows by the fact that $\mathcal{O}$ is injective, and by the definition of $i\mathcal{O}$ and the oracles $\mathcal{O}, \mathsf{Eval}$. We now show that it also satisfies indistinguishability, with an exponentially small distinguishing gap, even given an exponential number of oracle queries to $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{\mathcal{O},f})$ and the statistical distance oracle StaDif^{Ψ}.

Theorem 3.20. *Let $q(n) \leq O(2^{n/6})$. Then for any q-query adversary $\mathcal{A}$*

$$\left|\Pr\left[\mathsf{Exp}^{\mathsf{iO}}_{\Psi,\mathsf{StaDif},i\mathcal{O},\mathcal{C},\mathcal{A}}(n) = 1\right] - \frac{1}{2}\right| \leq O(2^{-n/6})$$

where the random variable $\mathsf{Exp}^{\mathsf{iO}}_{\Gamma,i\mathcal{O},\mathcal{C},\mathcal{A}}(n)$ denotes the adversary's winning probability in the IO security game (Definition 2.2) relative to $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$ and StaDif^{Ψ}.

At a very high-level, the proof of the theorem follows a similar rationale to the proof of Theorem 3.11 showing that one-way permutations do not break in the presence of the noisy statistical distance oracle. Roughly speaking, we show that in order to break the above construction of IO, the adversary must be able to detect local changes in the oracles realizing it, whereas the noisy statistical distance oracle is insensitive of these changes. At a technical level, the case of IO requires somewhat more care than the case of one-way permutations. For once, it has a more elaborate interface consisting not only of a hard to invert mapping $\mathcal{O}$, but also of the evaluation oracle $\mathsf{Eval}^{\mathcal{O}}$. In particular, a single change to $\mathcal{O}$ may introduce many changes to $\mathsf{Eval}^{\mathcal{O}}$, which could potentially be detected by the statistical distance oracle. Another aspect that complicates the proof is that the IO game is more interactive in its nature. In particular, we need to deal with the fact that the actual circuits of the IO challenge are chosen adaptively, after the adversary had already interacted with all the oracles. We now turn to the actual proof.

Proof. We prove a stronger statement: the above holds when fixing the oracles f and $\mathcal{O}_{-n} = \{\mathcal{O}_k\}_{k \neq n}$. For simplicity, we often suppress oracle access to the fixed $\mathcal{O}_{-n}, f$ in our notation and only denote the oracle $\mathcal{O}_n$. Fix a q-query adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ and let $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ be its smooth $(q + 2\beta q)$-query simulator given by Lemma 3.9, where β will be specified later on. Since $\mathcal{S}$ perfectly emulates $\mathcal{A}$, it suffices to prove the theorem for $\mathcal{S}$. To bound $\mathcal{S}$'s advantage in breaking $i\mathcal{O}$, we consider six hybrid experiments $\{\mathbf{H_i}\}_{i\in[6]}$ given in Table 3.2.

We introduce some notation that will be useful to describe the hybrids:

- For a function $\mathcal{O} : \{0,1\}^{2n} \rightarrow \{0,1\}^{5n}$, a pair $(C, r) \in \{0,1\}^{n\times 2}$, and $\widehat{C} \in \{0,1\}^{5n}$, we denote by $\mathcal{O}_{(C,r)\mapsto\widehat{C}}$ the function that maps (C, r) to $\widehat{C}$ and is otherwise identical to $\mathcal{O}$.

- For a function $\mathcal{O} : \{0,1\}^{2n} \to \{0,1\}^{5n}$, we denote by $\Gamma(\mathcal{O})$ the oracle

$$\Gamma(\mathcal{O}) := \mathcal{O}, \mathsf{Eval}^{\mathcal{O}}, \mathsf{StaDif}^{\mathcal{O},\mathsf{Eval}^{\mathcal{O}}} .$$

- For a function $\mathcal{O} : \{0,1\}^{2n} \to \{0,1\}^{5n}$, a string $\widehat{C} \in \{0,1\}^{5n}$, and a circuit C, we denote by $\Gamma(\mathcal{O}, \widehat{C}, C)$ the oracle

$$\Gamma(\mathcal{O}, \widehat{C}, C) := \mathcal{O}, \mathsf{Eval}^{\mathcal{O}}_{\widehat{C},C}, \mathsf{StaDif}^{\mathcal{O},\mathsf{Eval}^{\mathcal{O}}_{\widehat{C},C}} ,$$

  where $\mathsf{Eval}^{\mathcal{O}}_{\widehat{C},C}$ is an oracle that

  - Given $(\widehat{D}, x)$ where $\widehat{D} \neq \widehat{C}$, acts like $\mathsf{Eval}^{\mathcal{O}}(\widehat{D}, x)$. Namely, it computes $(D, r) = \mathcal{O}^{-1}(\widehat{D})$, and returns $D(x)$, or $\perp$ in case there is no unique preimage or the size of x does not match the input size of D.
  - Given $(\widehat{C}, x)$ returns $C(x)$, or $\perp$ in case $C = \perp$, or the size of x does not match the input size of C.

  (In both oracles $\Gamma(\mathcal{O})$ and $\Gamma(\mathcal{O}, \widehat{C}, C)$, the fixed $\mathcal{O}_{-n}, f$ are also a part of Γ and the oracles of Eval and StaDif. These are omitted from our notation for the sake of simplicity.)

Hybrid	$\mathbf{H_1}$ **(Real)**	$\mathbf{H_2}$	$\mathbf{H_3}$	$\mathbf{H_4}$	$\mathbf{H_5}$	$\mathbf{H_6}$ **(Ideal)**
Obfuscator Functions	$\mathcal{O} \leftarrow \mathbf{I}^{5n}_{2n}$	$\mathcal{O} \leftarrow \mathbf{I}^{5n}_{2n}$	$\mathcal{O} \leftarrow \mathbf{F}^{5n}_{2n}$	$\mathcal{O} \leftarrow \mathbf{F}^{5n}_{2n}$	$\mathcal{O} \leftarrow \mathbf{I}^{5n}_{2n}$	$\mathcal{O} \leftarrow \mathbf{I}^{5n}_{2n}$
Challenger Randomness	$b \leftarrow \{0,1\}$, $r \leftarrow \{0,1\}^n$					
Chosen Circuits	$(C_0, C_1) \leftarrow \mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n)$ where $C_0 \equiv C_1$ (relative to the fixed f)					
Planted Obfuscation	$\widehat{C} \leftarrow \{0,1\}^{5n}$					
Prechallenge Oracle	$\Gamma(\mathcal{O})$	$\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$	$\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$			$\Gamma(\mathcal{O})$
Challenge Obfuscation	$\mathcal{O}(C_b, r)$	$\mathcal{O}(C_b, r)$	$\mathcal{O}(C_b, r)$	$\widehat{C}$	$\widehat{C}$	$\widehat{C}$
Postchallenge Oracle	$\Gamma(\mathcal{O})$	$\Gamma(\mathcal{O})$	$\Gamma(\mathcal{O})$	$\Gamma(\mathcal{O}_{(C_b,r)\mapsto\widehat{C}})$	$\Gamma(\mathcal{O}_{(C_b,r)\mapsto\widehat{C}})$	$\Gamma(\mathcal{O}, \widehat{C}, C_0)$
Winning Condition	Guess b					

Table 3.2: The hybrid experiments.

Hybrid $\mathbf{H_1}$ is identical to the real world where $\mathcal{S}$ wins if it produces functionally equivalent C_0, C_1, and it successfully guesses the bit b. We show that the probability that the simulator wins in any of the experiments is roughly the same, and that in hybrid $\mathbf{H_6}$ the probability that $\mathcal{S}$ wins is 1/2.

Claim 3.21. $|\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_1}] - \Pr[\mathcal{S} \textit{ wins in } \mathbf{H_2}]| \leq O(2^{-n/6})$

Proof. The difference between the two hybrids is in the oracle that $\mathcal{S}_1$ is given before the challenge phase: $\Gamma(\mathcal{O})$ in the first, and its tweaked version $\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$ in the second. We stress that in $\mathbf{H_2}$, the circuit C_b is defined according to the circuits (C_0, C_1) that $\mathcal{S}_1$ would have chosen given the non-tweaked oracle $\Gamma(\mathcal{O})$ (so there is no circularity).[2]

We can bound the difference between the winning probabilities in $\mathbf{H_1}$ and $\mathbf{H_2}$ as follows:

$$\begin{aligned}&|\Pr[\mathcal{S} \text{ wins in } \mathbf{H_1}] - \Pr[\mathcal{S} \text{ wins in } \mathbf{H_2}]| \leq \\ &\Pr_{\substack{\mathcal{S}_1,\mathcal{O}\\ r,\Gamma}}\left[\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n) \neq \mathcal{S}_1^{\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}},0^{5n},\perp)}(1^n)\right] \ ,\end{aligned}$$

where $\mathcal{S}_1$ is the part of $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ that participates in the post challenge phase, and the probability is over the coins of $\mathcal{S}_1$ and Γ (specifically, StaDif) and the choice of $r \leftarrow \{0,1\}^n$, and $\mathcal{O} \leftarrow \mathbf{I}_{2n}^{5n}$, and $b \in \{0,1\}$ is arbitrary.

In what follows, we denote by $\mathbf{ZHit} = \mathbf{ZHit}(\mathcal{O})$ the event that 0^{5n} is in the image of $\mathcal{O}$, and by $\mathbf{Hit} = \mathbf{Hit}(\mathcal{S}_1, \mathcal{O}, r)$ the event that $\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n)$ queries $\mathcal{O}$ on (C_b, r). Also, let $\mathbf{Far} = \mathbf{Far}(\mathcal{S}_1, \mathcal{O}, 2\delta)$ be the event that 2δ-farness holds for all StaDif-queries made by $\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n)$ (Definition 3.7), and $\mathbf{Smo} = \mathbf{Smo}(\mathcal{S}_1, \Gamma(\mathcal{O}), \delta)$ be the event that all StaDif-queries made by $\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n)$ are (Ψ, δ)-smooth (Definition 3.7), where $\Psi = (\mathcal{O}, \mathcal{O}_{-n}, f, \mathsf{Eval}^{\mathcal{O},\mathcal{O}_{-n},f})$.

We now claim

Claim 3.22. *For any $\delta < 1$,*

$$\Pr_{\substack{\mathcal{S}_1,\mathcal{O}\\ r,\Gamma}}\left[\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n) \neq \mathcal{S}_1^{\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}},0^{5n},\perp)}(1^n)\right] \leq \Pr_{\substack{\mathcal{S}_1,\mathcal{O}\\ r,\Gamma}}\left[\mathbf{ZHit} \vee \mathbf{Hit} \vee \overline{\mathbf{Far}} \vee \overline{\mathbf{Smo}}\right] \ .$$

Proof. We argue that whenever the complement $\overline{\mathbf{ZHit}} \wedge \overline{\mathbf{Hit}} \wedge \mathbf{Far} \wedge \mathbf{Smo}$ occurs then

$$\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n) \neq \mathcal{S}_1^{\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}},0^{5n},\perp)}(1^n) \ .$$

We first note that when $\mathbf{ZHit}$ does not occur, the tweaked evaluation function $\mathsf{Eval}_{0^{5n},\perp}^{\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}}$ in $\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$ behaves exactly as the non-tweaked function $\mathsf{Eval}^{\mathcal{O}}$. Indeed, the only potential change in Eval is on inputs of the form $(0^{5n}, x)$, however, since 0^{5n} is not in the image of $\mathcal{O}$, $\mathsf{Eval}^{\mathcal{O}}$ returns $\perp$ on such inputs just like its tweaked version. Accordingly, the function $\Psi = (\mathcal{O}, \mathcal{O}_{-n}, f, \mathsf{Eval}^{\mathcal{O},\mathcal{O}_{-n},f})$ changes on the single input (C_b, r) for $\mathcal{O}$.

Also, for any StaDif-query (C_0, C_1) made by $\mathcal{S}^{\Gamma(\mathcal{O})}(1^n)$, we know by (Ψ, δ)-smoothness that changing Ψ at one point does not affect the statistical distance by much. Concretely, by Claim 3.8:

$$\left|\Delta(\boldsymbol{C}_0^{\Psi}, \boldsymbol{C}_1^{\Psi}) - \Delta(\boldsymbol{C}_0^{\Psi'}, \boldsymbol{C}_1^{\Psi'})\right| \leq 2\delta \ ,$$

where Ψ' is the tweaked version of Ψ in $\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$.

[2] In more detail, we first look at an execution of $\mathcal{S}_1$ with $\Gamma(\mathcal{O})$, as in $\mathbf{H_1}$, with respect to the sampled $\mathcal{O}, b, r$ (and coins of $\mathcal{S}_1$). This defines circuits (C_0, C_1), one of which is the challenge circuit C_b. Then we consider an execution with exactly the same samples $\mathcal{O}, b, r$, but with a pre-challenge oracle $\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$.

Furthermore, if 2δ-farness also holds for any such query (for some threshold $\mathbf{t}$ sampled by StaDif), then

$$\mathsf{StaDif}^{\Psi}(C_0, C_1; \mathbf{t}) = \mathsf{StaDif}^{\Psi'}(C_0, C_1; \mathbf{t}) \ .$$

If in addition **Hit** does not occur, then for any $\mathcal{O}$-query (C, s) made by $\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n)$,

$$\mathcal{O}(C, s) = \mathcal{O}_{(C_b, r)\mapsto 0^{5n}}(C, s) \ .$$

It follows that the views of $\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n)$ and $\mathcal{S}_1^{\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}})}(1^n)$ are identical. □

It is left to bound the probability of each of the events $\mathbf{ZHit}, \mathbf{Hit}, \overline{\mathbf{Far}}, \overline{\mathbf{Smo}}$.

First, by counting

$$\Pr[\mathbf{ZHit}] = 2^{2n}/2^{5n} = 2^{-3n} \ .$$

Second, noting that the view of $\mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n))$ is independent of the random r, we can bound

$$\Pr[\mathbf{Hit}] \leq 2^{-n} \cdot \#\{\mathcal{O}\text{-queries made by } \mathcal{S}_1\} \leq 2^{-n} \cdot (q + 2\beta q) \ .$$

Further more by the farness Claim 3.6 and smoothing Lemma 3.9

$$\Pr\left[\overline{\mathbf{Far}}\right] \leq 12q\delta \ .$$
$$\Pr\left[\overline{\mathbf{Smo}}\right] \leq 2^{-\delta\beta + \log(2q^2/\delta)} \ ,$$

Overall we can bound the difference between $\mathbf{H_1}$ and $\mathbf{H_2}$ by

$$2^{-3n} + 2^{-\delta\beta + \log(2q^2/\delta)} + 2^{-n} \cdot (q + 2\beta q) + 12q\delta \leq O(2^{-n/6}) \ ,$$

when setting $\delta = 2^{-n/3}, \beta = 2^{n/3} \cdot n$, and recalling that $q \leq O(2^{n/6})$. □

Claim 3.23. $|\Pr[\mathcal{S} \text{ wins in } \mathbf{H_2}] - \Pr[\mathcal{S} \text{ wins in } \mathbf{H_3}]| \leq 2^{-n}$

Proof. The difference between the two hybrids is in the choice of the oracle $\mathcal{O}$: a random injective function in the first, and a random function in the second. Thus,

$$|\Pr[\mathcal{S} \text{ wins in } \mathbf{H_2}] - \Pr[\mathcal{S} \text{ wins in } \mathbf{H_3}]| \leq$$
$$\Pr_{\mathcal{O}\leftarrow \mathbf{F}_{2n}^{5n}}[\mathcal{O} \text{ is not injective}] \leq 2^{-n} \ .$$

□

Claim 3.24. $\Pr[\mathcal{S} \text{ wins in } \mathbf{H_3}] = \Pr[\mathcal{S} \text{ wins in } \mathbf{H_4}]$.

Proof. The difference between $\mathbf{H_3}$ and $\mathbf{H_4}$ is that in $\mathbf{H_4}$, in the challenge and post challenge phases, the value $\mathcal{O}(C_b, r)$ is re-sampled uniformly at random, i.e. it is replaced everywhere by $\widehat{C} \leftarrow \{0,1\}^{5n}$. We claim that this induces exactly the same distribution on $\mathcal{S}$'s view as in $\mathbf{H_3}$. Indeed, in $\mathbf{H_3}$, the view of $\mathcal{S}$ in prechallenge phase is completely independent of $\mathcal{O}(C_b, r)$ because $\mathcal{O}$ is a random function and thus $\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}$ contains no information regarding $\mathcal{O}(C_b, r)$. □

Claim 3.25. $|\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_4}] - \Pr[\mathcal{S} \textit{ wins in } \mathbf{H_5}]| \leq 2^{-n}$

Proof. The difference between the two hybrids is in the choice of the oracle $\mathcal{O}$: a random injective function in the first, and a random function in the second. The proof is thus identical to the proof Claim 3.23. □

Claim 3.26. $|\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_5}] - \Pr[\mathcal{S} \textit{ wins in } \mathbf{H_6}]| \leq O(2^{-n/6})$.

Proof. There are two differences between the hybrids. The first is in the oracle that $\mathcal{S}_1$ is given before the challenge phase: $\Gamma(\mathcal{O})$ in $\mathbf{H_6}$, and its tweaked version $\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$ in $\mathbf{H_5}$. The second is in the oracle that $\mathcal{S}_1$ is given after the challenge phase: $\Gamma(\mathcal{O}, \widehat{C}, C_0)$ in $\mathbf{H_6}$, and $\Gamma(\mathcal{O}_{(C_b,r)\mapsto \widehat{C}}, \widehat{C}, C)$ in $\mathbf{H_5}$. We can thus bound the difference between the winning probabilities in $\mathbf{H_5}$ and $\mathbf{H_6}$ as follows:

$$
\begin{aligned}
&|\Pr[\mathcal{S} \text{ wins in } \mathbf{H_5}] - \Pr[\mathcal{S} \text{ wins in } \mathbf{H_6}]| \leq \\
&\Pr_{\substack{\mathcal{S}_1,\mathcal{O}\\ r,\Gamma}}\left[\mathsf{state} := \mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n) \neq \mathcal{S}_1^{\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)}(1^n)\right] + \\
&\Pr_{\substack{\mathcal{S},\mathcal{O}\\ r,\Gamma}}\left[\mathcal{S}_2^{\Gamma(\mathcal{O},\widehat{C},C_0)}(\mathsf{state}) \neq \mathcal{S}_2^{\Gamma(\mathcal{O}_{(C_b,r)\mapsto \widehat{C}}, \widehat{C}, C)}(\mathsf{state}) \;\middle|\; \mathsf{state} = \mathcal{S}_1^{\Gamma(\mathcal{O})}(1^n)\right] \quad ,
\end{aligned}
$$

where the probabilities are over the coins of $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ and Γ (specifically, StaDif) and the choice of $r \leftarrow \{0,1\}^n$, and $\mathcal{O} \leftarrow \mathbf{I}_{2n}^{5n}$, and $b \in \{0,1\}$ is arbitrary.

As proved in Claim 3.12, the first summand is bounded by $O(2^{-n/6})$. We argue that a similar bound holds for the second summand as well. The proof is essentially identical to that of Claim 3.12 with one exception: in Claim 3.12, we argued that $\mathsf{Eval}_{0^{5n},\perp}^{\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}}$ in $\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}, 0^{5n}, \perp)$ behaves exactly as $\mathsf{Eval}^{\mathcal{O}}$. Here, we need to argue that $\mathsf{Eval}_{\widehat{C},C}^{\mathcal{O}_{(C_b,r)\mapsto \widehat{C}}}$ behaves exactly as $\mathsf{Eval}_{\widehat{C},C_0}^{\mathcal{O}}$. Indeed, the two can only differ on inputs of the form $(\widehat{C}, x)$, where the first would return $C_b(x)$ and the second $C_0(x)$. However, by the functional equivalence of (C_0, C_1), the two are identical. □

To conclude the proof of Theorem 3.11, we observe that

Claim 3.27. $\Pr[\mathcal{S} \textit{ wins in } \mathbf{H_6}] = \frac{1}{2}$.

Proof. The view of $\mathcal{S}$ in this hybrid is completely independent of the random choice of b. □

□

Chapter 4

One-Way Functions, Indistinguishability Obfuscation, and Hardness in $\mathsf{NP} \cap \mathsf{coNP}$

In this section, we show that injective one-way functions (IOWFs) and indistinguishability obfuscation (IO), for circuits with IOWF-gates, do not give rise to a black-box construction of hard problems in $\mathsf{NP} \cap \mathsf{coNP}$. This can be seen as a generalization of previous separations by Rudich [Rud84], showing that OWFs do not give hardness in $\mathsf{NP} \cap \mathsf{coNP}$, by Matsuda and Matsuura [MM11], showing that IOWFs do not give one-way permutations (which are a special case of hardness $\mathsf{NP} \cap \mathsf{coNP}$), and by Asharov and Segev [AS16], showing that OWFs and IO do not give one-way permutations. As in the previous section, the result implies that many cryptographic primitives and hardness in PPAD, also do not yield black-box constructions of hard problems in $\mathsf{NP} \cap \mathsf{coNP}$.

We first define the framework of $\mathsf{NP} \cap \mathsf{coNP}$ relative to oracles, define fully black-box constructions of hard $\mathsf{NP} \cap \mathsf{coNP}$ problems, and then move on to the actual separation.

4.1 $\mathsf{NP} \cap \mathsf{coNP}$

Throughout, we shall canonically represent languages $L \in \mathsf{NP} \cap \mathsf{coNP}$ by their corresponding non-deterministic poly-time verifiers V_1, V_0, where

$$L = \{x \in \{0,1\}^\star \mid \exists w : V_1(x,w) = 1\} \enspace ,$$
$$\overline{L} = \{x \in \{0,1\}^\star \mid \exists w : V_0(x,w) = 1\} = \{0,1\}^\star \setminus L \enspace .$$

Hardness in $\mathsf{NP} \cap \mathsf{coNP}$ from cryptography - motivation. Hard (on average) problems in $\mathsf{NP} \cap \mathsf{coNP}$ are known to follow based on certain number-theoretic problems in cryptography, such as Discrete Log and Factoring. As in the previous section for SZK, we are interested in understanding which cryptographic primitives would imply such hardness, again with the intuition that these should be appropriately structured. For instance, it is well-known that any one-way permutation $f : \{0,1\}^n \to \{0,1\}^n$ implies a hard problem in $\mathsf{NP} \cap \mathsf{coNP}$, e.g. given an index $i \in [n]$ and an image $f(x)$ find the ith pre-image bit x_i. In contrast, in his seminal work, Rudich [Rud84] proved that completely unstructured objects like one-way functions cannot construct even worst-case hard instances by fully black-box constructions. Here a fully black-box construction essentially means that the non-deterministic verifiers only make black-box use of the OWF (or OWP in the previous example) and the reduction establishing the hardness is also black-box (in both the adversary and the OWF).

But what about more structured primitives such as public-key encryption, oblivious transfer, or even indistinguishability obfuscation. Indeed, IO (plus OWFs) has-been shown to imply hardness in PPAD and more generally in the class TFNP of total search problem, which is often viewed as the search analog of $\mathsf{NP} \cap \mathsf{coNP}$ [MP91]. We will show, however, that fully black-box constructions do not give rise to a hard problem in $\mathsf{NP} \cap \mathsf{coNP}$ from OWFs (or even injective OWFs) and IO for circuits with OWF gates.

4.2 Fully Black-Box Constructions of Hardness in $\mathsf{NP} \cap \mathsf{coNP}$ from IO and IOWFs

We start by defining $\mathsf{NP} \cap \mathsf{coNP}$ relative to oracles [Rud84]. This, in particular, captures black-box constructions of such languages from cryptographic primitives, such as one-way functions in [Rud84] or indistinguishability obfuscation, which we will consider in this work.

Definition 4.1 ($\mathsf{NP} \cap \mathsf{coNP}$ relative to oracles)**.** For a family of oracles $\mathfrak{S}$, we say that a collection of languages $L^{\mathfrak{S}} = \left\{ L^\Gamma \;\middle|\; \Gamma \in \mathfrak{S} \right\}$ is in $\mathsf{NP} \cap \mathsf{coNP}$ if there exist oracle-aided non-deterministic polynomial-time verifiers $V_1^{(\cdot)}, V_0^{(\cdot)}$ such that for any $\Gamma \in \mathfrak{S}$, the machines V_1^Γ, V_0^Γ define a language $L^\Gamma \in \mathsf{NP}^\Gamma \cap \mathsf{coNP}^\Gamma$. That is

$$L^\Gamma = \left\{ x \in \{0,1\}^\star \;\middle|\; \exists w : V_1^\Gamma(x,w) = 1 \right\} \;\; ,$$
$$\overline{L}^\Gamma = \left\{ x \in \{0,1\}^\star \;\middle|\; \exists w : V_0^\Gamma(x,w) = 1 \right\} = \{0,1\}^\star \setminus L \;\; .$$

We now formally define the class of constructions and reductions ruled out. That is, *fully black-box* constructions of hard problems in $\mathsf{NP} \cap \mathsf{coNP}$ from injective one-way functions (IOWFs) and IO for IOWF-aided circuits. The definition is similar in spirit to those in [AS15, AS16] and in the Section 3, adapted to the context of $\mathsf{NP} \cap \mathsf{coNP}$ hardness.

Definition 4.2. A fully black-box construction of a hard $\mathsf{NP} \cap \mathsf{coNP}$ problem from IOWFs and IO for the class $\mathcal{C}$ of circuits with IOWF-gates is given by two oracle aided poly-time machines (V_0, V_1) and a probabilistic oracle-aided reduction $\mathcal{R}$ that satisfy:

1. **Structure:** Let f be any permutation and let $i\mathcal{O}$ be any function such that $\widehat{C}^f \equiv C^f$ for any $C^{(\cdot)} \in \mathcal{C}$, r, and $\widehat{C}^{(\cdot)} := i\mathcal{O}(C, r)$. Then $(V_0^{f,\mathcal{O}}, V_1^{f,\mathcal{O}})$ define a language $L^{f,\mathcal{O}} \in \mathsf{NP}^{f,\mathcal{O}} \cap \mathsf{coNP}^{f,\mathcal{O}}$

(as per Definition 4.1).

2. **Black-box security proof:** There exist functions $q_{\mathcal{R}}(\cdot), \varepsilon_{\mathcal{R}}(\cdot)$ such that the following holds. Let f be any distribution on injective functions and let $i\mathcal{O}$ be any distribution on functions such that $\widehat{C}^f \equiv C^f$ for any $C^{(\cdot)} \in \mathcal{C}$, r, and $\widehat{C}^{(\cdot)} := i\mathcal{O}(C, r)$. Then for any probabilistic oracle-aided $\mathcal{A}$ that *decides L in the worst-case*, namely, for all $n \in \mathbb{N}$

$$\Pr_{f,i\mathcal{O},\mathcal{A}}\left[\mathcal{A}^{f,i\mathcal{O}}(x) = b \quad \text{for all} \quad \begin{array}{c} x \in \{0,1\}^n, b \in \{0,1\} \\ \text{such that } V_b(x) = 1 \end{array}\right] = 1$$

the reduction breaks either f or $i\mathcal{O}$, namely, for infinitely many $n \in \mathbb{N}$ either

$$\Pr_{\substack{x \leftarrow \{0,1\}^n \\ f,i\mathcal{O},\mathcal{A}}}\left[\mathcal{R}^{\mathcal{A},f,i\mathcal{O}}(f(x)) = x\right] \geq \varepsilon_{\mathcal{R}}(n) \ ,$$

or

$$\left|\Pr\left[\mathsf{Exp}^{\mathsf{iO}}_{(f,i\mathcal{O}),i\mathcal{O},\mathcal{C},\mathcal{R}^{\mathcal{A}}}(n) = 1\right] - \frac{1}{2}\right| \geq \varepsilon_{\mathcal{R}}(n) \ ,$$

where in both $\mathcal{R}$ makes at most $q_{\mathcal{R}}(n)$ queries to any of its oracles $(\mathcal{A}, f, i\mathcal{O})$. The random variable $\mathsf{Exp}^{\mathsf{iO}}_{(f,i\mathcal{O}),i\mathcal{O},\mathcal{C},\mathcal{R}^{\mathcal{A}}}(n)$ represents the reductions winning probability in the IO security game (Definition 2.2) relative to $(f, i\mathcal{O})$.

We note that as in Section 3, our definition addresses *worst-case hardness*, which makes our impossibility result stronger. See further discussion after Definition 3.2 in Section 3.

Ruling out fully black-box constructions: a road map. Our main result in this section is that fully black-box constructions of a hard $\mathsf{NP} \cap \mathsf{coNP}$ problem from IO and IOWFs do not exist. Furthermore, this holds even if the latter primitives are exponentially secure.

Theorem 4.3. *Any fully black-box construction of an* $\mathsf{NP} \cap \mathsf{coNP}$ *problem L from IOWFs and IO for circuits with IOWF gates has an exponential security loss:* $\max(q_{\mathcal{R}}(n), \varepsilon_{\mathcal{R}}^{-1}(n)) \geq \Omega(2^{n/6})$.

The proof of the theorem follows a similar methodology to that in Section 3. We exhibit two (distributions on) oracles $(\Psi, \mathsf{Decide}^{\Psi})$, where Ψ realizes IOWFs and IO for circuits with IOWF gates, and Decide^{Ψ} that decides $L^{\Psi} \in \mathsf{NP}^{\Psi} \cap \mathsf{coNP}^{\Psi}$ in the worst case. We then show that the primitives realized by Ψ are (exponentially) secure even in the presence of Decide^{Ψ}. Then viewing Decide as a worst-case decider $\mathcal{A}$ (as per Definition 4.2) directly implies Theorem 4.3 establishing ruling out fully black-box constructions with a subexponential security loss.

The rest of this section is organized according to the above plan. First, in Section 4.3, we describe the oracle Decide^{Ψ}. Then, in Sections 4.4,4.5, we describe the oracle Ψ realizing IOWFs and IO and prove its (exponential) security in the presence of Decide^{Ψ}. These results rule out *fully black-box* separations of $\mathsf{NP} \cap \mathsf{coNP}$ from P using IOWFs, IO. In 4.6, we show a generic transformation from our fully black-box separation to a *relativizing separation* — that is an oracle world where $\mathsf{NP}^\Gamma \cap \mathsf{coNP}^\Gamma = \mathsf{PPT}^\Gamma$ and secure IOWFs, IO exist.

4.3 The Decision Oracle

In this section, we construct an oracle that given access to $\Psi \in \mathfrak{S}$ breaks all $\mathsf{NP}^\Psi \cap \mathsf{coNP}^\Psi$ languages. The oracle we will eventually consider would be the pair $(\Psi, \mathsf{Decide}^\Psi)$.

Definition 4.4 (Decide oracle)**.** Given a family of oracles $\mathfrak{S}$, we define the Decide oracle as follows:

- Decide has oracle access to Ψ where $\Psi \in \mathfrak{S}$.
- Decide takes as input a pair of circuits (V_0, V_1) along with an input z where the circuits V_0, V_1 define an $\mathsf{NP} \cap \mathsf{coNP}$ language for $\mathfrak{S}$.
- $\mathsf{Decide}^\Psi(V_0, V_1, z)$ does the following:
  1. Checks that $V_0^{\Psi'}, V_1^{\Psi'} \in \mathsf{NP}^{\Psi'} \cap \mathsf{coNP}^{\Psi'}$ for all $\Psi' \in \mathfrak{S}$. If not, output $\perp$.
  2. For the input z, it finds the lexicographically first w such that $V_b^\Psi(z, w) = 1$ for some b. Then outputs b.

A few remarks about the Decide oracle.

1. We will use the Decide oracle similar to the StaDif oracle. We will consider an oracle family $\mathfrak{S}$ that implements the required primitive $\mathcal{P}$ and consider the oracle to be $\Gamma = (\Psi, \mathsf{Decide}^\Psi)$ where $\Psi \leftarrow \mathfrak{S}$. This decide oracle will break the $\mathsf{NP}^\mathcal{P} \cap \mathsf{coNP}^\mathcal{P}$ language constructed using the oracle Ψ.

2. Since $\mathsf{NP} \cap \mathsf{coNP}$ does not have a complete language, we allow the oracle accept descriptions of the verifiers. The verifiers we accept are bounded only in their query complexity to Ψ.

3. Which witness the Decide oracle uses is inconsequential because of two reasons — first one being that the Decide oracle does not reveal it and secondly because it is an $\mathsf{NP} \cap \mathsf{coNP}$ language, all the witnesses either certify that x is in the language or all of them certify that z is not in the language. So, finding any one of them is sufficient to output the decision.

4. Decide does a check in the beginning to verify that V_0, V_1 are valid verifiers. This check does not give the adversary any power because the adversary could have done this check independently because the adversary is bounded in the number of queries he makes to the oracle and not in computational power.

5. The behavior of the Decide oracle on oracles outside the oracle family $\mathfrak{S}$ is unspecified. So, in our proofs, we need to be careful and ensure that in all hybrids, we always stick to oracles within the family.

We want to subsequently show that while Decide^{Ψ} is sufficient to break any $\mathsf{NP}^{\Psi} \cap \mathsf{coNP}^{\Psi}$ construction, it is not helpful in breaking one-way functions and indistinguishability obfuscation. Like Theorem 3.18, our strategy here is to change the oracles locally such that the required answer changes. We would like the Decide oracle to be oblivious to such changes. We now show that it is the case. More specifically, we show that for every Decide query, there is a small *critical* set of positions on the oracle, such that as long as they are invariant, the answer of the Decide query does not change.

We call a verifier V q-query if for any $z \in \{0,1\}^n$, and any potential witness w, the computation $V^{\Psi}(z,w)$ makes at most $q(n)$ queries before accepting or rejecting. Similarly, we call a $\mathsf{Decide}(V_0, V_1, z)$ query q-query if both the verifiers V_0 and V_1 are q-query verifiers.

Lemma 4.5. *Consider an oracle Ψ from the family $\mathfrak{S}$. Consider any q-query $\mathsf{Decide}^{\Psi}(V_0, V_1, z)$ query. Then there exists a* critical *set of queries $\mathbf{C}$ such that*

1. *The set is small: $|\mathbf{C}| < q$.*

2. *The answer is invariant if $\mathbf{C}$ is unchanged. That is for every $\Psi' \in \mathfrak{S}$ such that $\Psi|_{\mathbf{C}} = \Psi'|_{\mathbf{C}}$,*

$$\mathsf{Decide}^{\Psi}(V_0, V_1, z) = \mathsf{Decide}^{\Psi'}(V_0, V_1, z)$$

Proof. This Lemma exploits the fact that in any $\mathsf{NP} \cap \mathsf{coNP}$ language L, and any input z, if $z \in L$, then all the accepting witnesses w certify that $V_1(z,w) = 1$ and no witness exists that certifies $V_0(z,w) = 1$. So, on an input z, we consider any witness w such that $V_b^{\Psi}(z,w) = 1$. We call the queries made in this computation critical. In any oracle consistent with Ψ on the critical queries will still have w as a witness to the statement $V_b^{\Psi'}(z,w) = 1$. And due to this, Decide has to answer the same way in the new oracle.

To prove the Lemma, let $b = \mathsf{Decide}^{\Psi}(V_0, V_1, z)$. Consider any witness w which certifies this. That is $V_b^{\Psi}(z,w) = 1$. We define set $\mathbf{C}$ to be the queries $V_b^{\Psi}(z,w)$ makes to Ψ to verify that $V_b^{\Psi}(z,w) = 1$. Clearly this set is not unique. Different witnesses could make different queries. We can pick any one arbitrarily.

Now, we consider any $\Psi' \in \mathfrak{S}$ such that it is consistent on $\mathbf{C}$ that is $\Psi|_{\mathbf{C}} = \Psi'|_{\mathbf{C}}$. Then we see that

$$V_b^{\Psi'}(z,w) = 1$$

This follows from the fact that $1 = V_b^{\Psi}(z,w) = V_b^{\Psi'}(z,w)$ because both these oracles are consistent on $\mathbf{C}$ and $V_b^{\Psi}(z,w)$ makes only those queries to Ψ. So, even with Ψ', the same witness w is accepted.

So, $\mathsf{Decide}^{\Psi'}(V_0, V_1, z) = b$ because $V_b^{\Psi'}(z,w) = 1$. This completes the proof. □

4.4 Warmup: Injective One-Way Functions in the Presence of Decide

As a warmup, we give a generalization of [MM11] which shows that injective one-way functions cannot be used to construct one-way permutations in a black-box manner. We extend this result to show that injective one-way functions that expand by one bit cannot be used to separate $\mathsf{NP} \cap \mathsf{coNP}$ from P even in the worst case.[1]

We consider an oracle family $\mathfrak{S}$ to be the set of all injective one-bit expanding functions that is $\mathfrak{S} = \{f\}$ where $f = \left\{f_n : \{0,1\}^n \to \{0,1\}^{n+1}\right\}_{n \in \mathbb{N}}$ is one-bit expanding injective function. The oracle Γ we consider is $\Gamma = (f, \mathsf{Decide}^f)$ where $f \leftarrow \mathfrak{S}$. We will show that $\mathsf{NP}^f \cap \mathsf{coNP}^f \subseteq \mathsf{P}^\Gamma$ and furthermore, f is a one-way function. We actually prove a stronger statement. Even if V_0, V_1 made an exponential number of queries to f, they still cannot construct a hard language. Formally define our oracle below:

Definition 4.6. Let $\mathbf{I}_n^m$ denote the distribution on all injective functions from $\{0,1\}^n$ to $\{0,1\}^m$. The IOWF is defined as $f = \{f_n\}_{n \in \mathbb{N}}$ where $f_n \leftarrow \mathbf{I}_n^{n+1}$ for all $n \in \mathbb{N}$. The oracle Γ consists of (f, Decide^f).

The only restriction we impose on V_0 and V_1 is that they make at most $2^{n/4}$-query verifiers. That is for any input $z \in \{0,1\}^n$, and any witness w, the computation $V_b(z, w)$ makes at most $2^{n/4}$ queries to f before accepting or rejecting. This not only covers $\mathsf{NP} \cap \mathsf{coNP}$ but also larger similar classes.

Claim 4.7. *Let L be the language defined by V_0^f and V_1^f. Then $L \in \mathsf{P}^\Gamma$.*

This is immediate from the definition of Γ. We will show the stronger statement that $\mathsf{NP}^{\Psi, \mathsf{Decide}^\Psi} \cap \mathsf{coNP}^{\Psi, \mathsf{Decide}^\Psi} \subseteq \mathsf{P}^\Psi$ in Theorem 4.21. We now show that even in the presence of the Decide oracle, f is a secure IOWF.

Theorem 4.8. *Any q-query adversary $\mathcal{A}$ where $q = 2^{n/4}$ cannot invert f_n with non-negligible probability.*

$$\Pr_{f, x \leftarrow \{0,1\}^n} \left[\mathcal{A}^\Gamma(f_n(x)) = x\right] < 2^{-n/3}$$

Proof. We need to show that even given access to the Decide oracle, an adversary cannot invert f. We show this is via a coupling argument. We want to look at the adversary's view in two worlds — the real world where the adversary gets a challenge $f(x)$ for a random x and the ideal world where the adversary gets $y \leftarrow \{0,1\}^{n+1} \setminus \mathrm{Image}(f)$ as the challenge. We will show that with very high probability, the adversary's view in both the worlds is identical. Since the adversary cannot invert in the ideal world, f is secure in the real world. To that end, we consider the following three hybrids in Table 4.1. A brief description of the hybrids is given below:

[1] [MM11] show a slightly different statement — they show that the injective function is adaptively one-way. That is even given the ability to invert the function at all values except the challenge, it is still hard to invert. Our proof works unchanged for this stronger definition. We omit it for expository reasons.

$\mathbf{H_1}$ This is the OWF security game. We pick a random injective function and a random input x. The adversary gets $f(x)$ as the challenge to invert.

$\mathbf{H_2}$ This is also the OWF security game, albeit with some syntactic changes. We sample the OWF by first sampling f along with a random pair x, y where $y \notin \text{Image}(f)$ and the oracle is $f_{x \mapsto y}$. This is identically distributed to picking $f \leftarrow \mathbf{I}_n^{n+1}$. The adversary gets the challenge y which is the image of x. These two distributions are identical.

$$(f, x, f(x)) \cong (f_{x \mapsto y}, x, y = f_{x \mapsto y}(x))$$

$\mathbf{H_3}$ This is the ideal world. Here we sample f and a random y not in the image of f and give that as the challenge. The adversary cannot invert in this world.

Eventually we will show that the oracles and the challenge in the first two hybrids are identically distributed. Then we will switch the oracle between $\mathbf{H_2}$ and $\mathbf{H_3}$ while keeping the challenge identical. For this this argument to go through, it is crucial that all the hybrid oracles we consider are *legitimate* oracles in $\mathfrak{S}$ — due to the fact that the verifiers are guaranteed to define an $\mathsf{NP} \cap \mathsf{coNP}$ language only for valid oracles.

Hybrid	**$\mathbf{H_1}$ (Real)**	**$\mathbf{H_2}$**	**$\mathbf{H_3}$ (Ideal)**
Injective OWF	$f = \left\{ f_t \leftarrow \mathbf{I}_t^{t+1} \right\}_{t \in \mathbb{N}}$		
Preimage	$x \leftarrow \{0,1\}^n$		
Planted Image	$y \leftarrow \{0,1\}^{n+1} \setminus \text{Image(f)}$		
Challenge	$f_n(x)$	y	y
Oracle	f, Decide^f	$f_{x \mapsto y}, \mathsf{Decide}^{f_{x \mapsto y}}$	f, Decide^f
Winning Condition	Find x		

Table 4.1: The hybrid experiments.

We will now show that the adversary cannot distinguish between the hybrids and hence cannot invert.

Claim 4.9. $\Pr_{f,x,y}[\mathcal{A} \text{ wins in } \mathbf{H_1}] = \Pr_{f,x,y}[\mathcal{A} \text{ wins in } \mathbf{H_2}]$

Proof. This follows from the fact that the two hybrids are identical. We are picking a random f and a random y outside the range and planting it at a random $x \in \{0,1\}^n$. This new oracle $f_{x \mapsto y}$ is also uniformly distributed in $\mathbf{I}_n^{n+1}$. Also the adversary's view in the two hybrids — in $\mathbf{H_1}$, $(f, f(x))$ and in $\mathbf{H_2}$, $(f_{x \mapsto y}, f_{x \mapsto y}(x))$ are identically distributed. □

Claim 4.10. *For every fixed f,*

$$\Pr_{x,y}[\mathcal{A} \text{ wins in } \mathbf{H_3}] = 2^{-n}$$

Proof. In the third hybrid $\mathbf{H_3}$, the challenge y is independent of the answer x and x is chosen uniformly at random. So, with probability 2^{-n}, the adversary's response will be x. □

Now for the more interesting part, we want to show that the hybrids $\mathbf{H_2}$ and $\mathbf{H_3}$ are indistinguishable. The strategy here is we observe that $f_{x \mapsto y}$ and f differ in exactly one location — x. Furthermore, we know that in the ideal world ($\mathbf{H_3}$), the probability of x being queried is exponentially low. So, the adversary with overwhelming probability will get the same response to any f query he makes. To get a different answer in hybrids $\mathbf{H_2}$ and $\mathbf{H_3}$, has to get a different response from some $\mathsf{Decide}(V_0, V_1, z)$ query that the adversary makes. We show that the adversary would get the same answer for most x's.

Claim 4.11. $|\Pr_{f,x,y}[\mathcal{A} \textit{ wins in } \mathbf{H_2}] - \Pr_{f,x,y}[\mathcal{A} \textit{ wins in } \mathbf{H_3}]| \leq 2^{-n/2}$

Proof. The adversary would make queries to f and Decide oracles. We want to show that the adversary sees the same view in both $\mathbf{H_2}$ and $\mathbf{H_3}$. We look at the queries adversary makes to the Decide oracle. We want to show that the adversary gets the same answer in both $\mathbf{H_3}$ and $\mathbf{H_2}$ with very high probability. We first start by showing that for every Decide query the adversary $\mathcal{A}$ makes to Γ, it receives the same answer in both the hybrids.

Claim 4.12. *Consider a* Decide *query* (V_0, V_1, z) *where* V_0, V_1 *are* $2^{n/4}$*-query verifiers. Then for every fixed* f *and* $y \in \{0,1\}^{n+1}$,

$$\Pr_{x \leftarrow \{0,1\}^n}\left[\mathsf{Decide}^f(V_0, V_1, z) \neq \mathsf{Decide}^{f_{x\mapsto y}}(V_0, V_1, z)\right] < 2^{-3n/4}$$

Proof. We want to use Lemma 4.5 on the oracles f and $f_{x\mapsto y}$. Using Lemma 4.5 requires that both the oracles are in $\mathfrak{S}$ that is are injective. As long as x is not in the critical set of the Decide query, it would return the same answer. We show that since x is chosen randomly and that the critical set is small, this has to be the case.

Consider the critical set $\mathbf{C}$ as given by Lemma 4.5 for the query $\mathsf{Decide}^f(V_0, V_1, z)$(that is in hybrid $\mathbf{H_3}$). We know that $|\mathbf{C}| < q = 2^{n/4}$. The condition $f|_{\mathbf{C}} \equiv f_{x\mapsto y}|_{\mathbf{C}}$ is equivalent to $x \notin \mathbf{C}$ because the two oracles are injective and differ only at x.

So, as long as $x \notin \mathbf{C}$, Decide will return the same answer. As x is chosen randomly independent of f, y and z,

$$\Pr_x[x \notin \mathbf{C}] < \frac{|\mathbf{C}|}{2^n} < 2^{-3n/4}$$

This proves the claim. □

Given Claim 4.12, the rest of the proof is essentially a union bound.

Let Q_0 be the set of queries the adversary $\mathcal{A}$ makes to f directly. The probability that $x \in Q_0$ is also bounded by $\frac{|Q_0|}{2^n} \leq 2^{-3n/4}$ because x is chosen independent of the adversary's view (that is (f, y) in the ideal world).

If the adversary makes $< 2^{n/4}$ Decide queries, then the total probability of distinguishing is at most $2^{n/4} \times 2^{-3n/4} \leq 2^{-n/2}$

□

So, the adversary's inversion probability is bounded by $2^{-n/2} + 2^{-n} < 2^{-n/3}$. This completes the proof. □

4.5 Indistinguishability Obsfuscation (and IOWFs) in the Presence of Decide

Now we extend our oracle Ψ to one that can implement both injective one-way functions and indistinguishability obfuscation. Using this definition, we show that both injective one-way functions together with indistinguishability obfuscation cannot separate $\mathsf{NP} \cap \mathsf{coNP}$ from P even in the worst-case in a fully black-box manner. We define the oracle in Definition 4.13.

Definition 4.13 (The Oracle Ψ)**.** The oracle $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$ consists of three parts:

- $f = \{f_n\}_{n\in\mathbb{N}}$ on input $x \in \{0,1\}^n$ answers with $f_n(x)$, where f_n is a random injective one-bit expanding function $f_n \leftarrow \mathbf{I}_n^{n+1}$.
- $\mathcal{O} = \{\mathcal{O}_n\}_{n\in\mathbb{N}}$ on input $(C, r) \in \{0,1\}^n \times \{0,1\}^n$ answers with $\widehat{C} := \mathcal{O}_n(C, r)$ where $\mathcal{O}_n$ is a random injective function $\mathcal{O}_n \leftarrow \mathbf{I}_{2n}^{5n}$ into $\{0,1\}^{5n}$.
- $\mathsf{EvalMap} : \left\{\mathsf{EvalMap}_n : \{0,1\}^{5n} \to \{0,1\}^n\right\}$ maps obfuscations to circuits. It has the following properties
  - It is not externally visible directly.
  - $\mathsf{EvalMap}$ maps every obfuscated circuit description $\widehat{C}$ of size $5t$ to a random circuit of size t.
- $\mathsf{Eval}^{f,\mathcal{O}}(\widehat{C}, x)$ checks if $\widehat{C}$ is in the image of $\mathcal{O}_n$. If it is, it finds $(C, r) = \mathcal{O}_n^{-1}(\widehat{C})$ and returns the answer $C^f(x)$. If $\widehat{C}$ is not in the image, it uses $\mathsf{EvalMap}$ to answer. That is

$$\mathsf{Eval}^{f,\mathcal{O}}(\widehat{C}, x) = \begin{cases} C^f(x) & \text{If } \widehat{C} \in \text{Image}(\mathcal{O}_n) \text{ and } \mathcal{O}_n(C,r) = \widehat{C} \\ \mathsf{EvalMap}(\widehat{C})^f(x) & \text{If } \widehat{C} \notin \text{Image}(\mathcal{O}_n) \end{cases}$$

Verifiability The oracle defined here in Definition 4.13 differs from Definition 3.17 because it has an added $\mathsf{EvalMap}$ which modifies Eval behavior. The only effect this has is that Eval answers on strings $\widehat{C}$ that are not valid obfuscations. This is intentional for two reasons:

1. Even when Eval answers on invalid alleged obfuscations, this is still a legitimate implementation of $i\mathcal{O}$ as defined in Definition 2.2. So, given this implementation, a black-box reduction should be able to construct a language in $\mathsf{NP} \cap \mathsf{coNP}$.

2. If $i\mathcal{O}$ has this additional property of '*verifiability*'(which Definition 3.17 has), then we can use this to construct a worst-case hard $\mathsf{NP} \cap \mathsf{coNP}$ language. To illustrate this, we show it in a simpler setting — an IOWF f along with the ability to verify if a given y is in the image of f.

$$\mathsf{Verify}^f(y) = 1 \iff \exists x,\ f(x) = y$$

Let b be a hard-core predicate for f. Then the language L is — $y \in L$ if $\exists x,\ y = f(x) \wedge b(x) = 1$. This language is in $\mathsf{NP} \cap \mathsf{coNP}$. To see that $L \in \mathsf{coNP}$, observe that if $y \notin \text{Image}(f)$ then no witness is needed (use $\mathsf{Verify}^f(y)$) and for others, the pre-image x serves to certify that $b(x) = 0$ is the witness. And being able to decide this language implies being able to invert f because b is a hard-core predicate.

The oracle Γ now consists of $(\Psi, \mathsf{Decide}^\Psi)$. To complete the argument, we need to show three things —

1. All $\mathsf{NP}^\Psi \cap \mathsf{coNP}^\Psi$ languages are easy to decide.
2. f is a one-way function.
3. $\mathcal{O}, \mathsf{Eval}$ can be used to implement indistinguishability obfuscation.

Lemma 4.14. $\mathsf{NP}^\Psi \cap \mathsf{coNP}^\Psi \subseteq \mathsf{P}^\Gamma$.

Proof. Follows from the definition of Decide. Given any pair of verifiers V_0, V_1, on input z, return $\mathsf{Decide}(V_0, V_1, z)$. □

We say that an adversary is a q-query adversary if it makes at most q-queries to the oracle and further more each $\mathsf{Decide}(V_0, V_1, z)$ query, the verification circuits V_0, V_1 make at most q queries on any particular witness to either accept or reject.

Lemma 4.15. *f is a one-way function. That is for every q-query adversary $\mathcal{A}$,*

$$\Pr_{\Psi, x \in \{0,1\}^n}\left[\mathcal{A}^\Gamma(f_n(x)) = x\right] < 2^{-n/3}$$

where $q = 2^{n/4}$.

Proof. This is a reduction to Theorem 4.8. Given an adversary $\mathcal{A}$ that can invert in this setting, we can convert it to an adversary $\mathcal{B}$ that given oracle access to f, Decide^f, can invert f with the same probability. The reduction samples the oracles $\mathcal{O}, \mathsf{EvalMap}$ at random and then answers $\mathcal{A}$'s queries accordingly. When the adversary makes $\mathsf{Decide}^{f,\mathcal{O},\mathsf{Eval}}$ queries, the reduction translates them to Decide^f queries by including a table for $\mathcal{O}, \mathsf{Eval}$. The success probability is the same and there is no increased query overhead. These reductions are legitimate because we only care about the query complexity of the verifiers in Decide and not their computational complexity. □

The construction of this obfuscator is identical to that in Construction 3.19.

Construction 4.16 (The Obfuscator $i\mathcal{O}^{\Psi}$)**.** *Let* $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$*. Given an oracle-aided circuit* $C \in \{0,1\}^n$*,* $i\mathcal{O}^{\Psi}(1^n, C)$ *samples a random* $r \leftarrow \{0,1\}^n$*, computes* $\widehat{C} = \mathcal{O}(C, r)$*, and returns an oracle aided circuit* $E_{\widehat{C}}$ *that given input* x*, computes* $\mathsf{Eval}^{f,\mathcal{O}}(\widehat{C}, x)$*.*

As earlier, $i\mathcal{O}^{f,\mathcal{O},\mathsf{Eval}}$ satisfies the functionality requirement of Definition 2.2 for f-aided circuits; indeed, this follows by the fact that $\mathcal{O}$ is injective, and by the definition of $i\mathcal{O}$ and the oracles $\mathcal{O}, \mathsf{Eval}$. The fact that we now have Eval giving answers outside the image of $\mathcal{O}$ does not alter this. We now show that it also satisfies indistinguishability, with an exponentially small distinguishing gap, even given an exponential number of oracle queries to $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{\mathcal{O},f})$ and the decide oracle Decide^{Ψ}.

Theorem 4.17. *Let* $q(n) \leq 2^{n/4}$*. Then for any* q*-query adversary* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$

$$\left|\Pr\left[\mathsf{Exp}^{\mathsf{iO}}_{\Gamma,i\mathcal{O},\mathcal{A}}(n) = 1\right] - \frac{1}{2}\right| \leq O(2^{-n/3})$$

where the random variable $\mathsf{Exp}^{\mathsf{iO}}_{\Gamma,i\mathcal{O},\mathcal{A}}(n)$ *the adversary's winning probability in the IO security game (Definition 2.2) relative to* $\Psi = (f, \mathcal{O}, \mathsf{Eval}^{f,\mathcal{O}})$ *and* Decide^{Ψ}*.*

Proof. The proof strategy is quite similar to Theorem 4.8. We will consider an ideal world where the given challenge is uncorrelated from b and hence the adversary has no advantage in this world. We will then show that this world is indistinguishable from the Real world. To that end, we need some notation.

Notation As earlier, $\mathcal{O}_{(C,r)\mapsto\widehat{C}}$ refers to a function identical to $\mathcal{O}$ everywhere except at (C, r) where it returns $\widehat{C}$. We need two modifications to Eval. We similarly define $\mathsf{EvalMap}_{\widehat{C}\mapsto C}$ to refer to a function identical to $\mathsf{EvalMap}$ everywhere except at $\widehat{C}$ where it returns C.

The hybrids we use are given in Table 4.1. We briefly describe them here:

- $\mathbf{H_1}$ This is the conventional security game for IO. The adversary gives the challenger a pair of equivalent circuits (C_0, C_1) and gets back the obfuscation $\widehat{C} = \mathcal{O}(C_b, r)$ of one of them. And the adversary has to guess which one.

- $\mathbf{H_2}$ This hybrid is distributionally identical to $\mathbf{H_1}$. We sample the obfuscation oracle a bit differently — by first sampling one from the right distribution $\mathcal{O}_n$ and then planting a value $\widehat{D}$ outside the image at a fixed location to get $\mathcal{O}_{(C_b,r)\mapsto\widehat{D}}$. This has identical distribution to $\mathcal{O}_n$. As earlier, we stress that in $\mathbf{H_2}$, the circuit C_b is defined according to the circuits (C_0, C_1) that $\mathcal{A}_1$ would have chosen given the non-tweaked oracle $\Gamma(\mathcal{O})$ (so there is no circularity).[2]

- $\mathbf{H_3}$ This hybrid is the *ideal world* where the adversary cannot win. We are picking a random $\widehat{D}$ outside $\mathrm{Image}(\mathcal{O}_n)$ and returning that as the obfuscation. This has no correlation with b and hence implies no advantage. Along with this, to ensure that Eval returns the correct answers

[2] In more detail, we first look at an execution of $\mathcal{A}_1$ with $\Gamma(\mathcal{O})$, as in $\mathbf{H_1}$, with respect to the sampled $\mathcal{O}, b, r$ (and coins of $\mathcal{A}_1$). This defines circuits (C_0, C_1), one of which is the challenge circuit C_b. Then we consider an execution with exactly the same samples $\mathcal{O}, b, r$, but with a pre-challenge oracle $\Gamma(\mathcal{O}_{(C_b,r)\mapsto\widehat{D}})$.

Hybrid	$\mathbf{H_1}$ (Real)	$\mathbf{H_2}$	$\mathbf{H_3}$ (Ideal)
Obfuscation Oracle	$\mathcal{O}_n \leftarrow \mathbf{I}_{2n}^{5n}$		
Evaluation Map	$\mathsf{EvalMap}_n$		
Randomness	$b \leftarrow \{0,1\}, r \leftarrow \{0,1\}^n$		
Planted Challenge	$\widehat{D} \leftarrow \{0,1\}^{5n} \setminus \text{Image}(\mathcal{O}_n)$		
Pre-challenge Oracle	$\mathcal{O}_n$ $\mathsf{EvalMap}$	$\mathcal{O}_{(C_b,r)\mapsto\widehat{D}}$ $\mathsf{EvalMap}$	$\mathcal{O}_n$ $\mathsf{EvalMap}_{\widehat{D}\mapsto C_0}$
Challenge	$\mathcal{O}_n(C_b, r)$	$\widehat{D}$	$\widehat{D}$
Post-Challenge Oracle	$\mathcal{O}_n$ $\mathsf{EvalMap}$	$\mathcal{O}_{(C_b,r)\mapsto\widehat{D}}$ $\mathsf{EvalMap}$	$\mathcal{O}_n$ $\mathsf{EvalMap}_{\widehat{D}\mapsto C_0}$
Winning Condition	Find b		

Table 4.2: The hybrid experiments.

on $\widehat{D}$, we set $\mathsf{EvalMap}_{\widehat{D}\mapsto C_0}$ that is $\mathsf{Eval}(\widehat{D}, x) = C_0^f(x)$. Since C_0^f and C_1^f are identical, this gives a consistent response on Eval irrespective of which C_b was sampled.

In the previous argument we split the hybrids differently. Notably we did not change the pre-challenge oracle and the post-challenge oracle together. We first changed the pre-challenge oracle to $\mathcal{O}_{(C_b,r)\mapsto 0^{5n}}$ and then switched the post-challenge oracle. We also changed Eval so that it answered $\perp$ on 0^{5n}. We cannot do that here, because this change to the oracle puts the oracle outside the family. So, we do both the changes simultaneously here. We need to show that the hybrids are indistinguishable to the adversary. Consider any $2^{n/4}$-query adversary $\mathcal{A}$.

Claim 4.18. *For every q-query adversary $\mathcal{A}$ where $q = 2^{n/4}$,*

$$\Pr_{\Gamma,b,r}[\mathcal{A} \textit{ wins in } \mathbf{H_1}] = \Pr_{\Gamma,\widehat{D},b,r}[\mathcal{A} \textit{ wins in } \mathbf{H_2}]$$

Proof. As described, these two hybrids are semantically identical with only syntactic differences. In hybrid $\mathbf{H_2}$, we are sampling the obfuscation oracle a bit differently — first choosing an oracle $\mathcal{O}_n$ at random and then at one fixed location (C_b, r) we are changing the value to $\widehat{D} \notin \text{Image}(\mathcal{O}_n)$. This new function $\mathcal{O}_{(C_b,r)\mapsto\widehat{D}}$ is identically distributed to $\mathcal{O}_n$. And the challenge the adversary gets in both the cases is the same — the value at position (C_b, r) that is $\mathcal{O}_n(C_b, r)$ in $\mathbf{H_1}$ and $\mathcal{O}_{(C_b,r)\mapsto\widehat{D}}(C_b, r)$ in $\mathbf{H_2}$.

□

Like the proof of Theorem 4.8, this is the more interesting hybrid indistinguishability argument. Because we are switching both the pre-challenge and post-challenge oracles simultaneously, we will have to argue that both in the pre-challenge phase and in the post-challenge phase, the adversary cannot distinguish between the hybrids.

Claim 4.19. *For every q-query adversary $\mathcal{A}$ where $q = 2^{n/4}$,*

$$\left| \Pr_{\Gamma,b,r}[\mathcal{A} \text{ wins in } \mathbf{H_2}] - \Pr_{\Gamma,b,r}[\mathcal{A} \text{ wins in } \mathbf{H_3}] \right| < O(2^{-n/2})$$

Proof. First we show that the adversary cannot distinguish between the hybrids in the pre-challenge phase. Let $\widehat{C} = \mathcal{O}_n(C_b, r)$.

Pre-challenge phase In the pre-challenge phase, the two oracles differ at two places:

1. A (C_b, r) query returns $\widehat{D}$ in $\mathbf{H_2}$ and $\mathcal{O}_n(C_b, r)$ in the other.

2. $\mathsf{Eval}(\widehat{C}, \cdot)$ may get a different answer in the two oracles as well. In $\mathbf{H_3}$ it would be answered according to C_b and in $\mathbf{H_2}$ according to $\mathsf{EvalMap}(\widehat{C})$ as it is not in the image.

The only way in which the adversary could distinguish between the two hybrids pre-challenge would be to query one of these two either directly or indirectly. We show that the probability of that happening is exponentially low. The fact that r is chosen independently implies that with very high probability, the adversary will not query either $\mathsf{Eval}(\widehat{C}, \cdot)$ or $\mathcal{O}_n(C_b, r)$.

Let Q_0 be the set of queries $\mathcal{A}$ makes to $\mathcal{O}_n$ directly. Let $\{(V_{i,0}, V_{i,1}, z_i)\}_{i<q}$ be the set of Decide queries $\mathcal{A}$ makes in $\mathbf{H_3}$ before the challenge.

We will use Lemma 4.5 to see that with high probability, every Decide query will return the same answer. We consider some critical set for each of the decide query. Let us say that the critical set for the i-th query is $\mathbf{C}_i$. Then consider $Q = Q_0 \cup_{i<q} \mathbf{C}_i$.

Let $\mathbf{Hit} = \mathbf{Hit}(\mathcal{A}_1, \mathcal{O}, r)$ be the event that $\mathcal{A}^{\Gamma(\mathcal{O})}(1^n)$ queries $\mathcal{O}$ on (C_b, r) directly or indirectly that is $(C_b, r) \in Q \vee \mathsf{Eval}(\widehat{C}, \cdot))$ in the hybrid $\mathbf{H_3}$. We choose to start with $\mathbf{H_3}$ because in this world, (C_b, r) is independent of everything the adversary sees. So,

$$\Pr_{\Gamma,r}\left[\mathcal{A}_1^{\Gamma(\mathcal{O})}(1^n) \not\equiv \mathcal{A}_1^{\Gamma(\mathcal{O}_{(C_b,r)\mapsto 0^{5n}})}(1^n)\right] \leq \Pr_{\Gamma,r}[\mathbf{Hit}]$$

This is true because if $\overline{\mathbf{Hit}}$ occurs, the oracles are identical and hence the adversary cannot distinguish. Both the oracles only differ at (C_b, r). As long as $(C_b, r) \notin Q \wedge \mathsf{Eval}(\widehat{C}, \cdot) \notin Q$, every Decide query will return the same answer in both $\mathbf{H_2}$ and $\mathbf{H_3}$. This follows from Lemma 4.5 because $\mathcal{O}|_Q = \mathcal{O}_{(C_b,r)\mapsto 0^{5n}}|_Q$ as $(C_b, r) \notin Q$ and hence $\mathcal{O}|_{\mathbf{C}_i} = \mathcal{O}_{(C_b,r)\mapsto 0^{5n}}|_{\mathbf{C}_i}$.

To complete the proof, we bound $\Pr_{b,r}[\mathbf{Hit}]$ by bounding the size of Q using union bound.

$$|Q| \leq |Q_0| + (q-1)|\mathbf{C}_i| \leq q + q(q-1) = q^2$$

So, $\Pr_{b,r}[\mathbf{Hit}] \leq \frac{2|Q|}{2^n} \leq 22^{-n/2}$. This completes the proof.

Post-challenge indistinguishability Now we need to show that post-challenge the oracles are indistinguishable. The proof is identical. The two oracles differ only at (C_b, r) and $\mathsf{EvalMap}(\widehat{C}, \cdot)$. Again in the ideal world, (C_b, r) is sampled independently of the challenge. So, again consider $Q = Q_0 \cap_i \mathbf{C}_i$ be the set of queries that affect the adversary $\mathcal{A}_2$. We know that the size of this set is small $|Q| \le 2^{n/2}$. And as r is chosen independently, the failure probability is bounded by $\frac{2|Q|}{2^n} \le 2 \times 2^{-n/2}$.

Hence the total failure probability in this hybrid is $4 \times 2^{n/2}$. This completes the argument. □

Claim 4.20. *The adversary has no advantage in* $\mathbf{H_3}$.

$$\Pr_{f,x,y}\left[\mathcal{A} \textit{ wins in } \mathbf{H_3}\right] = \frac{1}{2}$$

Proof. $\widehat{D}$ is chosen outside $\text{Image}(\mathcal{O}_n)$ and independent of b. In hybrid $\mathbf{H_3}$, no information about b is leaked. So, the adversary can win with probability half. □

To complete the proof, we add up the distinguishing advantage from the hybrids to get that

$$\Pr_{\mathcal{O}_n, b, r}\left[\mathcal{A} \text{ guesses } b\right] < \frac{1}{2} + 2^{-n/3}$$

□

4.6 Extension to Relativizing Separations

We showed a statement of the form $\mathsf{NP}^\Psi \cap \mathsf{coNP}^\Psi \subseteq \mathsf{P}^\Gamma$. We can strengthen that statement to $\mathsf{NP}^\Gamma \cap \mathsf{coNP}^\Gamma = \mathsf{P}^\Gamma$. In the parlance of black-box reductions [Fis12, HR04], the original statement ruled out *fully black-box* reductions while the current statement would rule out *relativizing reductions*.

Theorem 4.21. $\mathsf{NP}^\Gamma \cap \mathsf{coNP}^\Gamma = \mathsf{P}^\Gamma$.

Proof. Our strategy is the following: We will show that $\mathsf{NP}^{\Psi, \mathsf{Decide}^\Psi} \cap \mathsf{coNP}^{\Psi, \mathsf{Decide}^\Psi} \subseteq \mathsf{NP}^\Psi \cap \mathsf{coNP}^\Psi$. And then since we know $\mathsf{NP}^\Psi \cap \mathsf{coNP}^\Psi \subseteq \mathsf{P}^\Gamma$ the result follows. Given any pair of verifiers V_0, V_1 that make queries to both $\Psi, \mathsf{Decide}^\Psi$, we want to convert them to a pair of verifiers (S_0, S_1) which can simulate V_0, V_1 while actually not making any queries to Decide^Ψ. We do this by exploiting is the ability to get more information from the witness — specifically witnesses for the Decide queries. So, consider some input z and a witness w such that $V_b^{\Psi, \mathsf{Decide}^\Psi}(z, w) = 1$. And we would like to simulate this computation only using Ψ. The new witness we consider is the following: $w' = \left(w, \{(b_i, w_i)\}_{i<q}\right)$ where w is the original witness and the when running the computation $V_b^{\Psi, \mathsf{Decide}^\Psi}(z, w)$, b_i is the decision the i-th Decide^Ψ query returns and w_i is the witness for that. So the simulator when it encounters a $\mathsf{Decide}^\Psi(V_{i,0}, V_{i,1}, z_i)$ query reads the witness for the corresponding decision-witness pair and verifies that $V_{b_i}^\Psi(z_i, w_i) = 1$ and keeps going. We need to show that these simulated circuits define the same language and the simulated verifiers incur only a polynomial blowup in the query size.

More specifically given any $\mathsf{NP}^{\Psi,\mathsf{Decide}^\Psi} \cap \mathsf{coNP}^{\Psi,\mathsf{Decide}^\Psi}$ language described by q-query verifiers (V_0, V_1), there exist $q^2 + q$-query verifiers (S_0, S_1) that accept the same language having access to Ψ alone. We describe the simulated verifiers S below.

Simulator description The simulator S_b on input z along with a witness $(w, \{b_i, w_i\}_{i\in[q]})$ simulates $V_b^{\Psi,\mathsf{Decide}^\Psi}(z, w)$. There are three possible steps.

1. $V_b(z, w)$ does internal computation. The simulator S_b simulates it itself.
2. $V_b(z, w)$ simulation makes a query x to Ψ. The simulator queries x to Ψ and returns the response to V_b.
3. $V_b(z, w)$ makes a $\mathsf{Decide}(V_{i,0}, V_{i,1}, z_i)$ to the Decide oracle where $V_{i,0}$ and $V_{i,1}$ only query Ψ. In this case, S_b does the following:
   (a) Reads the next part of the witness (b_i, w_i).
   (b) Verifies that $V_{i,0}, V_{i,1}$ are valid verifiers. That is for all $\Psi' \in \mathfrak{S}$, $V_{i,0}^{\Psi'}, V_{i,1}^{\Psi'}$ are in $\mathsf{NP}^{\Psi'} \cap \mathsf{coNP}^{\Psi'}$. It can verify this because this requires no queries to Ψ.
   (c) S_b then verifies that $V_{i,b_i}^{\Psi}(z_i, w_i) = 1$. While doing this, it makes q-queries to Ψ. If $V_{i,b_i}^{\Psi}(z_i, w_i) = 0$, S_b rejects the witness and aborts. Otherwise continues.

   Finally at the end of the simulation, if $V_b(x, w) = 1$ in the simulation, it accepts the witness $(w, \{b_i, w_i\}_{i\in[q]})$. Otherwise it rejects.

Correctness We want to show that S_0^Ψ, S_1^Ψ define the same language as $V_0^{\Psi,\mathsf{Decide}^\Psi}, V_1^{\Psi,\mathsf{Decide}^\Psi}$. This follows from the fact that for every witness w, there exists the corresponding witness $w' = (w, \{b_i, w_i\}_{i\in[q]})$ where b_i is the correct Decide response and w_i certifies that. We know they exist. So, every accepting witness w for z in V_0, V_1, we have a corresponding accepting witness in S_0, S_1. Furthermore, $S_b(z, w') = 1$ implies that $V_b(z, w) = 1$ where $w' = (w, \{b_i, w_i\}_{i\in[q]})$ because S_b simply verifies this V_b computation.

Efficiency The number of queries S_b's make is $q^2 + q$. Each Decide query is simulated using q-queries and there are q such queries. Further more, V_b may make q queries to Ψ which create q additional queries.

This completes the argument. □

Chapter 5

Collision-Resistance from IO and SZK-Hardness

Asharov and Segev [AS15] showed that collision-resistant hashing cannot be constructed from (even subexponentially hard) indistinguishability obfuscation (IO) and one-way permutations (OWPs) relying on common IO techniques. Slightly more accurately, they rule out fully black-box constructions where (as in previous sections) IO is defined with respect to circuits with OWP oracle gates. In this section, we show that, assuming IO and a strong form of SZK-hardness, there is indeed a construction of collision-resistant hashing (CRH).

The high-level idea behind the construction. The starting point for our construction is the work of Ishai, Kushilevitz, and Ostrovsky [IKO05] that shows how to construct collision-resistant hash functions from commitments that are additively homomorphic (for simplicity, say over $\mathbb{F}_2$). The idea is simple: we can hash ℓ bits to m bits, where m is the size of a single bit commitment and ℓ can be arbitrarily longer, as follows. The hash key is a commitment $\gamma := (\mathsf{com}(\beta_1), \dots, \mathsf{com}(\beta_\ell))$ to a random vector $\beta \in \mathbb{F}_2^\ell$, and hashing $x \in \mathbb{F}_2^\ell$, is done by homomorphically computing a commitment to the inner product $\mathsf{CRH}_\gamma(x) = \mathsf{com}(\langle \beta, x \rangle)$.

This idea can, in fact, be abstracted to work with any commitment scheme wherein given a commitment $\mathsf{com}(\beta)$ for a random key for a 2-universal hash allows to homomorphically compute a commitment $\mathsf{com}(\mathsf{2UH}_\beta(x))$ to the hash at any point x, so that the resulting commitment is compact in the sense that it depends only on the size of $\mathsf{2UH}_\beta(x)$ and not on the size of x. Intuitively, the reason this works is that any collision in CRH_γ implies a collision in the underlying 2-universal hash $\mathsf{2UH}_\beta$, which leaks information about the hash key β (concretely, any fixed x, x' form a collision in a random hash function with small probability) thereby violating the hiding of the commitment.

At a high-level, we aim to mimic the above construction based on obfuscation. As a key for the collision-resistant hash we can obfuscate a program Π_β associated with a secret hash key β that given x outputs a commitment $\mathsf{com}(\mathsf{2UH}_\beta(x))$, where the commitment is derandomized using a PRF. The obfuscation $i\mathcal{O}(\Pi_\beta)$ can be thought of as the commitment to β, and evaluating this program at x, corresponds to homomorphic evaluation. Despite the clear intuition behind this construction, it is not clear how to prove its security based on IO. In fact, by [AS15], it cannot be proven based on a black-box reduction as long as plain statistically-binding commitments are used, as these can be constructed from OWPs in a fully black-box manner.

We show, however, that relying on a relaxed notion of perfectly-hiding commitments, as well as subexponential hardness of IO and puncturable PRFs, the construction can be proven secure. The perfect hiding of the commitment is leveraged in a probabilistic IO argument [CLTV15] that involves a number of hybrids larger than the overall number of commitments. We then observe that these relaxed commitments follow from appropriate average-case hardness of SZK.[1]

5.1 Definitions and Tools

We define our notion of relaxed perfectly-hiding commitments and the SZK-hardness they follow from. We also define 2-universal hashing and puncturable pseudorandom functions, which are used in our construction.

Relaxed perfectly-hiding commitments. We consider two message bit commitment schemes $(\mathcal{R}, \mathcal{S})$, where the receiver $\mathcal{R}$ samples a first message σ, and $\mathcal{S}(\sigma, b)$ samples a commitment ξ to a bit b. We require that the commitment is computationally binding and that there exists a distribution $\widetilde{\mathcal{R}}$ (not necessarily efficiently samplable) that is computationally indistinguishable from that generated by $\mathcal{R}$ and under which the commitment is perfectly hiding.

Definition 5.1 (Relaxed Statistically-Hiding Commitments)**.** $(\mathcal{R}, \mathcal{S})$ is a relaxed statistically-hiding commitment scheme if it satisfies:

1. **Computational Binding:** for any non-uniform PPT sender $\mathcal{S}^*$,

$$\Pr_{\sigma \leftarrow \mathcal{R}(1^n)} \left[\begin{array}{l} r_0, r_1 \leftarrow \mathcal{S}^*(\sigma) \\ \mathcal{S}(\sigma, 0; r_0) = \mathcal{S}(\sigma, 1; r_1) \end{array} \right] \leq \mathsf{negl}(n) \enspace .$$

2. **Relaxed Perfect-Hiding:** there exists a (possibly inefficient) sampler $\widetilde{\mathcal{R}}$ such that
   - $\mathcal{R}(1^n)$ and $\widetilde{\mathcal{R}}(1^n)$ are computationally indistinguishable,
   - for any $\widetilde{\sigma}$ in the support of $\widetilde{\mathcal{R}}(1^n)$, $\mathcal{S}(\widetilde{\sigma}, 0)$ and $\mathcal{S}(\widetilde{\sigma}, 1)$ are identically distributed.

Relaxed perfectly-hiding commitments are implied by the standard definition of 2-message perfectly hiding commitments. The standard definition is stronger in the sense that perfect-hiding holds for *any* (even maliciously chosen) σ. We next show that this definition is implied by appropriate average-hardness of SZK.

[1]Similar SZK-hardness is known to imply statistically-hiding commitments against malicious receivers, but with a larger (constant) number of rounds [OV08].

Average-case hardness of $\mathbf{SD}^{0,1}$. Roughly speaking, we require average-case hardness of an extreme case of the statistical-distance problem, referred to as $\mathbf{SD}^{0,1}$ in [Vad99]. Here YES-instances consist of pairs of samplers with disjoint support, whereas NO-instances consist of samplers with identical distribution.

Definition 5.2 (Average-Case Hardness of $\mathbf{SD}^{0,1}$)**.** The promise problem $\mathbf{SD}^{0,1} = (\mathbf{SD}_Y^{0,1}, \mathbf{SD}_N^{0,1})$ is given by

$$\begin{aligned}\mathbf{SD}_Y^{0,1} &= \{(C_0, C_1) \mid \Delta(\boldsymbol{C}_0, \boldsymbol{C}_1) = 1\} \quad , \\ \mathbf{SD}_N^{0,1} &= \{(C_0, C_1) \mid \Delta(\boldsymbol{C}_0, \boldsymbol{C}_1) = 0\} \quad .\end{aligned}$$

We say that the problem is hard on average if there exists a PPT sampler S with support $\mathbf{SD}^{0,1}$ such that for any non-uniform PPT decider D,

$$\Pr_{(C_0,C_1)\leftarrow \mathsf{S}(1^n)}\left[\begin{array}{c} B \leftarrow D(C_0, C_1) \\ (C_0, C_1) \in \mathbf{SD}_B^{0,1} \end{array}\right] \leq \frac{1}{2} + \mathsf{negl}(n) \quad .$$

The above definition should be contrasted with the standard definition of the statistical distance problem $\mathbf{SD} = \mathbf{SD}^{\frac{1}{3},\frac{2}{3}}$ (Definition 3.1) where the notions of statistical farness and closeness are not absolute (but given by the constants $\frac{1}{3}, \frac{2}{3}$). We note that the ploarization lemma in [SV03] gives an efficient reduction from deciding $\mathbf{SD}^{\frac{1}{3},\frac{2}{3}}$ to deciding $\mathbf{SD}^{2^{-n},1-2^{-n}}$, but such a reduction is not known if we replace $\mathbf{SD}^{2^{-n},1-2^{-n}}$ by $\mathbf{SD}^{0,1}$. Average-case hardness of $\mathbf{SD}^{0,1}$ is known under number-theoretic assumptions such as Decision-Diffie-Hellman and Quadratic Residuosity [GMR85], which are already known to imply collision-resistance directly. However, it may also follow from problems that are not known to imply collision-resistance, and may be of a non-algebraic nature. For instance, average-case hardness of $\mathbf{SD}^{0,1}$ would follow from the average-case hardness of Graph Non-Isomorphism [GMW91].

Claim 5.3. *Average-case hardness of* $\mathbf{SD}^{0,1}$ *implies relaxed perfectly-hiding commitments.*

Proof sketch. We define the receiver and sender $(\mathcal{R}, \mathcal{S})$. The receiver $\mathcal{R}$ is simply the instance sampler S, which outputs first messages of the form $\sigma = (C_0, C_1) \in \mathbf{SD}^{0,1}$. The sender $\mathcal{S}(C_0, C_1, b)$ outputs a random sample from $\boldsymbol{C}_b$.

We next prove binding and hiding. Denote by $\mathsf{S}_B(1^n)$ the distribution given by sampling (C_0, C_1) from $\mathsf{S}(1^n)$ conditioned on $(C_0, C_1) \in \mathbf{SD}_B^{0,1}$. We note that $\mathsf{S}_Y(1^n)$, $\mathsf{S}_N(1^n)$, and $\mathsf{S}(1^n)$ are computationally indistinguishable by the average-case hardness of S. Computational binding now holds by the fact that $\mathsf{S}_Y(1^n)$ and $\mathsf{S}(1^n)$ are computationally indistinguishable and $\mathsf{S}_Y(1^n)$ samples C_0, C_1 with disjoint supports. The fake receiver sampler $\widetilde{\mathcal{R}}$ is S_N, which is as required indistinguishable from $\mathcal{R} = \mathsf{S}$, and in which commitments are samples from $\boldsymbol{C}_0$ or $\boldsymbol{C}_1$, which are identically distributed. □

Remark 5.4. In our definition of *relaxed* perfectly-hiding commitments. The commitment scheme itself (corresponding to the honest $\mathcal{R}$) is neither perfectly hiding nor perfectly hiding according to the common definition. We note that assuming stronger notions of hardness in $\mathbf{SD}^{0,1}$ we they can be made such. Specifically, if S_Y is efficient, then using it in the actual scheme would make it perfectly binding. Alternatively, if S_N is efficient, then using it in the actual scheme would make it perfectly hiding.

Puncturable pseudo-random functions. We consider a simple case of the puncturable pseudo-random functions (PRFs) where any PRF may be punctured at a single point. The definition is formulated as in [SW14b], and is satisfied by the GGM [GGM86] PRF [BW13, KPTZ13, BGI14] and can be constructed from any one-way function. One-way functions are, in turn, implied by the average case hardness of $\mathbf{SD}^{0,1}$ [OW93b]. (Here we will need sub-exponential security of the PRF and thus sub-exponentially-hard one-way functions.)

Definition 5.5 (Puncturable PRFs)**.** Let k, ℓ, m be polynomially bounded functions. An efficiently computable family of functions

$$\mathcal{PRF} = \left\{ \mathsf{PRF}_\alpha : \{0,1\}^{\ell(n)} \to \{0,1\}^{m(n)} \;\middle|\; \alpha \in \{0,1\}^{k(n)}, n \in \mathbb{N} \right\} \;,$$

is a puncturable PRF if there exists a poly-time puncturing algorithm Punc that takes as input a key α, and a point x^*, and outputs a punctured key $\alpha\{x^*\}$, so that the following conditions are satisfied:

1. **Functionality is preserved under puncturing:** For every $x^* \in \{0,1\}^{\ell(n)}$,

$$\Pr_{\alpha \leftarrow \{0,1\}^{k(n)}} \left[\forall x \neq x^* : \mathsf{PRF}_\alpha(x) = \mathsf{PRF}_{\alpha\{x^*\}}(x) \;\middle|\; \alpha\{x^*\} = \mathsf{Punc}(\alpha, x^*) \right] = 1 \;.$$

2. **Indistinguishability at punctured points:** for any nonuniform PPT distinguisher D there exists a negligible function $\mathsf{negl}(\cdot)$, such that for all $n \in \mathbb{N}$, and any $x^* \in \{0,1\}^{\ell(n)}$,

$$|\Pr[D(x^*, \alpha\{x^*\}, \mathsf{PRF}_\alpha(x^*)) = 1] - \Pr[D(x^*, \alpha\{x^*\}, u) = 1]| \leq \mathsf{negl}(n) \;,$$

   where $\alpha \leftarrow \{0,1\}^{k(n)}, \alpha\{x^*\} = \mathsf{Punc}(\alpha, x^*)$, and $u \leftarrow \{0,1\}^{m(n)}$.

   We further say that $\mathcal{PRF}$ satisfies δ-indistinguishability if the above negligible indistinguishability gap is smaller than δ.

2-Universal hashing. We rely on 2-universal families of hash functions, which are known to exist unconditionally [WC81].

Definition 5.6 (2-Universal Hashing)**.** Let k, ℓ, m be polynomially bounded functions. An efficiently computable family of functions

$$2\mathcal{UH} = \left\{ \mathsf{2UH}_\beta : \{0,1\}^{\ell(n)} \to \{0,1\}^{m(n)} \;\middle|\; \beta \in \{0,1\}^{k(n)}, n \in \mathbb{N} \right\} \;,$$

is 2-universal if for any two distinct $x, x' \in \{0,1\}^{\ell(n)}$

$$\Pr_{\beta \leftarrow \{0,1\}^{k(n)}} \left[\mathsf{2UH}_\beta(x) = \mathsf{2UH}_\beta(x') \right] \leq 2^{-m(n)} \quad .$$

5.2 The Construction

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

Theorem 5.7. *Assuming average-case hardness of* $\mathbf{SD}^{0,1}$ *(or more generally, relaxed perfectly-binding commitments) and the existence of indistinguishability obfuscators and one-way functions that are subexponentially secure, there exists a collision-resistant hash function family.*

Let $\tau(\cdot)$ be an expansion parameter. To get a collision-resistant family with expansion τ, we rely on the following ingredients:

- A relaxed perfectly-hiding commitment scheme $(\mathcal{R}, \mathcal{S})$. We denote by $\ell(n)$ the size of bit commitments.
- An indistinguishability obfuscator $i\mathcal{O}$ with $2^{-\tau(n)\cdot\ell(n)} \cdot \mathsf{negl}(n)$-indistinguishability.
- A puncturable pseudo-random function family $\mathcal{PRF}$ satisfying $2^{-\tau(n)\cdot\ell(n)} \cdot \mathsf{negl}(n)$-indistinguishability.
- A 2-universal hashing family $2\mathcal{UH}$ mapping $\{0,1\}^{\tau(n)\cdot\ell(n)}$ to $\{0,1\}$.

We construct a collision-resistant hashing family

$$\mathcal{CRH} = \left\{ \mathsf{CRH}_\gamma : \{0,1\}^{\tau(n)\cdot\ell(n)} \to \{0,1\}^{\ell(n)} \;\middle|\; \gamma \in \{0,1\}^{k(n)}, n \in \mathbb{N} \right\} \quad ,$$

with an associated key generator $\mathsf{Gen}_{\mathcal{CRH}}$.

Construction 5.8 (A Collision-Resistant Hashing Family)**.** *$\mathcal{CRH}$ is given by:*

1. $\mathsf{Gen}_{\mathcal{CRH}}(1^n)$*:*
   - *generate a receiver message* $\sigma \leftarrow \mathcal{R}(1^n)$*,*
   - *sample a key* $\beta \leftarrow \{0,1\}^{k(n)}$ *for a 2-universal hash,*
   - *sample a key* $\alpha \leftarrow \{0,1\}^{k(n)}$ *for a puncturable PRF,*
   - *construct a circuit* $\Pi = \Pi[\sigma, \beta, \alpha]$ *that*
     - *given input* $x \in \{0,1\}^{\tau(n)\cdot\ell(n)}$
     - *computes the hash bit* $\rho_x := \mathsf{2UH}_\beta(x)$*,*
     - *outputs a commitment* $\xi_x := \mathcal{S}(\sigma, \rho_x; \mathsf{PRF}_\alpha(x))$*.*
   - *obfuscate* $\gamma \leftarrow i\mathcal{O}(\Pi, 1^n)$ *and output* γ *as the key.*
2. $\mathsf{CRH}_\gamma(x)$*:*

- *parse γ as a circuit and output $\xi_x = \gamma(x)$.*

Proposition 5.2.1. *$\mathcal{CRH}$ is collision-resistant.*

Proof. To prove the proposition, we shall prove the following two claims.

Claim 5.9. *Fix any two keys $\beta_0, \beta_1 \in \{0,1\}^{k(n)}$ for the 2-universal family, and let $\gamma|_{\beta_0}$ (respectively, $\gamma|_{\beta_1}$) be the distributions on CRH keys conditioned on hashing key β_0 (respectively, β_1). Then the two distributions are computationally indistinguishable.*

Claim 5.10. *Assume there exists an efficient A that finds collisions in $\mathcal{CRH}$ with probability δ over the choice of key γ. Then there exists an efficient predictor P that given random β_0, β_1, and $\gamma|_{\beta_b}$ for a random $b \leftarrow \{0,1\}$, predicts b with advantage $\frac{\delta - \mathsf{negl}(n)}{4}$.*

The two claims together imply that a collision finder A for $\mathcal{CRH}$ cannot exist.

Proof of Claim 5.9. First, we note that by the computational indistinguishability of honest receiver messages generated by $\mathcal{R}(1^n)$ and receiver messages generated by $\widetilde{\mathcal{R}}(1^n)$, it is enough to prove the claim for an alternative experiment where γ is sampled as usual except that $\widetilde{\sigma}$ is sampled from $\widetilde{\mathcal{R}}(1^n)$ rather than σ from $\mathcal{R}(1^n)$. In this new experiment, commitments to 0 and 1 are identically distributed. We now prove indistinguishability of $\gamma|_{\beta_0}$ and $\gamma|_{\beta_1}$ based on a standard probabilistic IO argument [CLTV15]. We sketch the argument for the sake of completeness.

For each input $x \in \{0,1\}^{\tau(n)\cdot\ell(n)}$, we consider a hybrid where a circuit $\Pi^x = \Pi^x[\widetilde{\sigma}, \beta_0, \beta_1, \alpha]$ is obfuscated, where

$$\begin{aligned} &\text{for } x' < x: && \Pi^x(x') = \Pi[\widetilde{\sigma}, \beta_0, \alpha](x') = \mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_0}(x'); \mathsf{PRF}_\alpha(x'))\ , \\ &\text{for } x' \geq x: && \Pi^x(x') = \Pi[\widetilde{\sigma}, \beta_1, \alpha](x') = \mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_1}(x'); \mathsf{PRF}_\alpha(x'))\ . \end{aligned}$$

Each two consecutive hybrids differ only at a single point x where one answers according to $\mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_0}(x); \mathsf{PRF}_\alpha(x))$ and the other according to $\mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_1}(x); \mathsf{PRF}_\alpha(x))$. We can then puncture α at x and hardwire these outputs, relying on IO, then replace them with truly random samples from $\mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_0}(x))$ and $\mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_1}(x))$, relying on pseudo-randomness at punctured points, and finally rely on the fact that the two circuits sample from two identical distributions. Since both $i\mathcal{O}$ and $\mathcal{PRF}$ satisfy $2^{-\tau(n)\ell(n)} \cdot \mathsf{negl}(n)$-indistinguishability, and the two samples from $\mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_0}(x))$ and $\mathcal{S}(\widetilde{\sigma}, \mathsf{2UH}_{\beta_1}(x))$ are perfectly indistinguishable, we get $O(2^{-\tau(n)\ell(n)} \cdot \mathsf{negl}(n))$-indistinguishability between any two consecutive hybrids. This allows us to deduce a negligible difference between the first and the last hybrid corresponding to $\gamma|_{\beta_0}$ and $\gamma|_{\beta_1}$.[2] □

Proof of Claim 5.10. We start by describing the predictor P.

Given β_0, β_1, and $\gamma = \gamma|_{\beta_b}$, P outputs a prediction b^* (for b) as follows:

[2]Above, since $\widetilde{\sigma}$ may not be efficiently samplable, we formally need to rely on the non-uniform security of $i\mathcal{O}$ and $\mathcal{PRF}$. Alternatively, we can require a less relaxed commitment notion where $\widetilde{\mathcal{R}}$ is also efficient (these still follow from standard perfectly-hiding commitments).

- Apply the collision-finder $A(\gamma)$.
- If A finds $x \neq x'$ such that
  - $\mathsf{CRH}_\gamma(x) = \mathsf{CRH}_\gamma(x')$,
  - $\mathsf{2UH}_{\beta_{b^*}}(x) = \mathsf{2UH}_{\beta_{b^*}}(x')$,
  - $\mathsf{2UH}_{\beta_{1-b^*}}(x) \neq \mathsf{2UH}_{\beta_{1-b^*}}(x')$,

  output b^*. We will below refer to this event as **Col**.
- Otherwise, output a random $b^* \leftarrow \{0,1\}$.

We now show that $b^* = b$ with probability $\frac{1+\delta/2-\mathsf{negl}(n)}{2}$. Indeed, for any $b \in \{0,1\}$, and sampling $\beta_0, \beta_1, \gamma|_{\beta_b}$ as above:

$$\begin{aligned}
&\Pr\left[P(\beta_0,\beta_1,\gamma|_{\beta_b}) = b\right] = \\
&\Pr\left[\mathbf{Col}\right] \cdot \Pr\left[P(\beta_0,\beta_1,\gamma|_{\beta_b}) = b \mid \mathbf{Col}\right] + \Pr\left[\overline{\mathbf{Col}}\right] \cdot \Pr\left[P(\beta_0,\beta_1,\gamma|_{\beta_b}) = b \;\middle|\; \overline{\mathbf{Col}}\right] = \\
&\Pr\left[\mathbf{Col}\right] \cdot 1 + \Pr\left[\overline{\mathbf{Col}}\right] \cdot \frac{1}{2} = \\
&\frac{1+\Pr\left[\mathbf{Col}\right]}{2} \quad .
\end{aligned}$$

It is left to note that

$$\Pr\left[\mathbf{Col}\right] \geq \delta/2 - \mathsf{negl}(n) \enspace .$$

Indeed, by our assumption A finds a collision $x \neq x'$ in $\gamma|_{\beta_b}$ with probability δ. Also, by 2-universality, choosing β_{1-b} at random, $\mathsf{2UH}_{\beta_{1-b}}(x) = \mathsf{2UH}_{\beta_{1-b}}(x')$ with probability at most $1/2$. (Note that β_{1-b} is independent of the A's view and thus from x, x'.) To conclude the argument we claim that the probability that x, x' is a collision in $\mathsf{CRH}_{\gamma|_{\beta_b}}$ but not in $\mathsf{2UH}_{\beta_b}$ is negligible by the computational binding of commitment scheme. To see this, note that whenever the latter event occurs, we know that

$$\mathcal{S}(\sigma, \mathsf{2UH}_{\beta_b}(x), \mathsf{PRF}_\alpha(x)) = \mathcal{S}(\sigma, \mathsf{2UH}_{\beta_b}(x'), \mathsf{PRF}_\alpha(x')) \qquad \text{but} \qquad \mathsf{2UH}_{\beta_b}(x) = \mathsf{2UH}_{\beta_b}(x') \enspace .$$

. □

This completes the proofs of the two claims and the proposition. □

Remark 5.11 (Statistical Hiding instead of Perfect Hiding.). In the above, we have defined and relied on perfectly-hiding commitments. It is natural to ask whether we can relax this requirement to statistical hiding. The bottleneck here is the probabilistic IO argument used in our proof, which requires $2^{-\tau\ell}$-indistinguishability of the commitments where ℓ is the size of a bit commitment and τ is the expansion factor. Accordingly, we can make do with a strong statistical guarantee where the statistical distance is say $2^{-s} \cdot \mathsf{negl}(n)$ where s is the size of a single bit commitment.

In our setting, where the commitment is implemented using the hard statistical distance problem, the above requirement translates to a hard samplable distribution on $\mathbf{SD}^{2^{-s}\cdot\mathsf{negl}(n),1}$ where s is a bound on the output length of samplers in the support of this distribution. This holds for example for known statistically-hiding commitments based on collision-resistant hashing [DPP93, HM96]. However, it cannot be achieved generically by, say by amplifying $\mathbf{SD}^{\frac{1}{3},\frac{2}{3}}$ (via the polarization lemma [SV03] mentioned above), since such amplification increases the size of samples.

Remark 5.12 (Relation with Section 3)*.* We note that the notion of statistical distance hardness required here is stronger than that ruled out in two ways. First, it requires hardness even of $\mathbf{SD}^{0,1}$, whereas in Section 3, we discuss $\mathbf{SD}^{\frac{1}{3},\frac{2}{3}}$. Second, it requires average-case hardness rather than the worst-case hardness considered in Section 3.

We note that the construction of collision-resistant hash functions in this section, in conjunction the result of [AS15] (ruling out fully black-box construction of CRH from OWP and IO for circuits with OWP gates), give an alternative proof to the statement that there is no fully black-box construction of hard on average problems in $\mathbf{SD}^{0,1}$ from OWPs. Indeed, this statement also follows from the more general statement in Section 3, saying that that is no fully black-box construction of worst-case hard problems in $\mathbf{SD}^{\frac{1}{3},\frac{2}{3}}$ from OWPs and IO for circuits with OWP gates. (Roughly speaking, the reason that this alternative proof covers constructions of hard statistical distance problems from OWPs but not from IO is that the result of [AS15] only covers IO for circuits with OWP gates, *but not with IO gates*. Indeed, in our construction the circuits representing the hard statistical distance problem are obfuscated.)

Chapter 6

Open Questions and Conclusions

In this work we ask whether cryptographic primitives imply hardness in natural "structured" complexity classes, such as SZK and $\mathsf{NP} \cap \mathsf{coNP}$, and give negative answers for a wide class of cryptographic primitives. Several questions remain open.

SZK: In the case of SZK, we provide a nearly complete classification: while homomorphic encryption (HE), private information retrieval (PIR) and re-randomizable encryption imply hardness in SZK, essentially every other cryptographic primitive could exist in a world where $\mathsf{SZK} = \mathsf{P}$.[1]

A notable exception is collision-resistant hashing for which we do not know the status relative to SZK. Does collision-resistant hashing exist in a world where $\mathsf{SZK} = \mathsf{P}$? Does (average-case) hardness of SZK imply the existence of collision-resistant hashing (perhaps in conjunction with indistinguishability obfuscation)? We do not know. As for the second question, we provide a partial answer by showing that a strong form of average-case hardness in SZK, together with indistinguishability obfuscation, in fact does imply collision-resistant hashing.

$\mathsf{NP} \cap \mathsf{coNP}$: In the case of $\mathsf{NP} \cap \mathsf{coNP}$, we show that every primitive implied (in a black-box way) by the existence of indistinguishability obfuscation and one-way functions could exist in a world where $\mathsf{NP} \cap \mathsf{coNP} = \mathsf{P}$. On the other hand, we know that one-way permutations cannot exist in such a world. This leaves us with a nearly complete picture except for fully homomorphic encryption, private information retrieval and a handful of other objects.

Capturing other techniques. This work captures the commonly used paradigm for constructions based on IO. This paradigm covers most but not all constructions from IO (see Section 3.2). It would be interesting to see if black-box separations can be shown for more relaxed constructions.

[1] This statement is proved in a relativized sense.

Bibliography

[AGGM06] Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on np-hardness. In Kleinberg [Kle06], pages 701–710.

[AR04] Dorit Aharonov and Oded Regev. Lattice problems in NP cap conp. In *45th Symposium on Foundations of Computer Science (FOCS 2004), 17-19 October 2004, Rome, Italy, Proceedings*, pages 362–371. IEEE Computer Society, 2004.

[AR16] Benny Applebaum and Pavel Raykov. From private simultaneous messages to zero-information arthur-merlin protocols and back. In Eyal Kushilevitz and Tal Malkin, editors, *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part II*, volume 9563 of *Lecture Notes in Computer Science*, pages 65–82. Springer, 2016.

[AS15] Gilad Asharov and Gil Segev. Limits on the power of indistinguishability obfuscation and functional encryption. In *Symposium on the Foundations of Computer Science*, 2015.

[AS16] Gilad Asharov and Gil Segev. On constructing one-way permutations from indistinguishability obfuscation. In *Theory of Cryptography*, pages 512–541. Springer, 2016.

[Bar13] Boaz Barak. Structure vs. combinatorics in computational complexity. http://windowsontheory.org/2013/10/07/structure-vs-combinatorics-in-computational-complexity/, 2013.

[BB15] Andrej Bogdanov and Christina Brzuska. On basing size-verifiable one-way functions on np-hardness. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *Theory of Cryptography - 12th Theory of Cryptography Conference, TCC 2015, Warsaw, Poland, March 23-25, 2015, Proceedings, Part I*, volume 9014 of *Lecture Notes in Computer Science*, pages 1–6. Springer, 2015.

[BBF13] Paul Baecher, Christina Brzuska, and Marc Fischlin. Notions of black-box reductions, revisited. In *Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I*, pages 296–315, 2013.

[BGI+01] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. In Joe Kilian, editor, *Advances in Cryptology - CRYPTO 2001, 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19-23, 2001, Proceedings*, volume 2139 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2001.

[BGI14] Elette Boyle, Shafi Goldwasser, and Ioana Ivan. Functional signatures and pseudorandom functions. In Hugo Krawczyk, editor, *PKC*, volume 8383 of *Lecture Notes in Computer Science*, pages 501–519. Springer, 2014.

[BGL+15] Nir Bitansky, Sanjam Garg, Huijia Lin, Rafael Pass, and Sidharth Telang. Succinct randomized encodings and their applications. In *Symposium on Theory of Computing, STOC 2015*, 2015.

[BGS75] Theodore Baker, John Gill, and Robert Solovay. Relativizations of the p=?np question. *SIAM Journal on computing*, 4(4):431–442, 1975.

[BKSY11] Zvika Brakerski, Jonathan Katz, Gil Segev, and Arkady Yerukhimovich. Limits on the power of zero-knowledge proofs in cryptographic constructions. In Ishai [Ish11], pages 559–578.

[BL13] Andrej Bogdanov and Chin Ho Lee. Limits of provable security for homomorphic encryption. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, volume 8042 of *Lecture Notes in Computer Science*, pages 111–128. Springer, 2013.

[BM09] Boaz Barak and Mohammad Mahmoody-Ghidary. Merkle puzzles are optimal - an $o(n^2)$-query attack on any key exchange from a random oracle. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, volume 5677 of *Lecture Notes in Computer Science*, pages 374–390. Springer, 2009.

[BP15] Nir Bitansky and Omer Paneth. Zaps and non-interactive witness indistinguishability from indistinguishability obfuscation. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *Theory of Cryptography - 12th Theory of Cryptography Conference, TCC 2015, Warsaw, Poland, March 23-25, 2015, Proceedings, Part II*, volume 9015 of *Lecture Notes in Computer Science*, pages 401–427. Springer, 2015.

[BPR15] Nir Bitansky, Omer Paneth, and Alon Rosen. On the cryptographic hardness of finding a nash equilibrium. In Venkatesan Guruswami, editor, *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1480–1498. IEEE Computer Society, 2015.

[BPW16] Nir Bitansky, Omer Paneth, and Daniel Wichs. Perfect structure on the edge of chaos - trapdoor permutations from indistinguishability obfuscation. In *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I*, pages 474–502, 2016.

[Bra79] Gilles Brassard. Relativized cryptography. In *20th Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 29-31 October 1979*, pages 383–391. IEEE Computer Society, 1979.

[BT03] Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for NP problems. In *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Cambridge, MA, USA, Proceedings*, pages 308–317. IEEE Computer Society, 2003.

[BV11] Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In Rafail Ostrovsky, editor, *FOCS*, pages 97–106. IEEE, 2011. Invited to SIAM Journal on Computing.

[BW13] Dan Boneh and Brent Waters. Constrained pseudorandom functions and their applications. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT (2)*, volume 8270 of *Lecture Notes in Computer Science*, pages 280–300. Springer, 2013.

[CDT09] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Settling the complexity of computing two-player nash equilibria. *J. ACM*, 56(3), 2009.

[CHJV15] Ran Canetti, Justin Holmgren, Abhishek Jain, and Vinod Vaikuntanathan. Succinct garbling and indistinguishability obfuscation for RAM programs. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 429–437, 2015.

[CLTV15] Ran Canetti, Huijia Lin, Stefano Tessaro, and Vinod Vaikuntanathan. Obfuscation of probabilistic circuits and applications. In *TCC*, 2015.

[Cra12] Ronald Cramer, editor. *Theory of Cryptography - 9th Theory of Cryptography Conference, TCC 2012, Taormina, Sicily, Italy, March 19-21, 2012. Proceedings*, volume 7194 of *Lecture Notes in Computer Science*. Springer, 2012.

[DBL00] *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA*. IEEE Computer Society, 2000.

[DGP06] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a nash equilibrium. In Kleinberg [Kle06], pages 71–78.

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

[DHT12] Yevgeniy Dodis, Iftach Haitner, and Aris Tentes. On the instantiability of hash-and-sign RSA signatures. In Cramer [Cra12], pages 112–132.

[DLMM11] Dana Dachman-Soled, Yehuda Lindell, Mohammad Mahmoody, and Tal Malkin. On the black-box complexity of optimally-fair coin tossing. In Ishai [Ish11], pages 450–467.

[DPP93] Ivan Damgård, Torben P. Pedersen, and Birgit Pfitzmann. On the existence of statistically hiding bit commitment schemes and fail-stop signatures. In *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, pages 250–265, 1993.

[Fis12] Marc Fischlin. Black-box reductions and separations in cryptography. In Aikaterini Mitrokotsa and Serge Vaudenay, editors, *Progress in Cryptology - AFRICACRYPT 2012 - 5th International Conference on Cryptology in Africa, Ifrance, Morocco, July 10-12, 2012. Proceedings*, volume 7374 of *Lecture Notes in Computer Science*, pages 413–422. Springer, 2012.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *STOC*, pages 169–178, 2009.

[GGH+13] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 40–49. IEEE Computer Society, 2013.

[GGKT05] Rosario Gennaro, Yael Gertner, Jonathan Katz, and Luca Trevisan. Bounds on the efficiency of generic cryptographic constructions. *SIAM J. Comput.*, 35(1):217–246, 2005.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.

[GK93] Oded Goldreich and Eyal Kushilevitz. A perfect zero-knowledge proof system for a problem equivalent to the discrete logarithm. *Journal of Cryptology*, 6(2):97–116, 1993.

[GKLM12] Vipul Goyal, Virendra Kumar, Satyanarayana V. Lokam, and Mohammad Mahmoody. On black-box reductions between predicate encryption schemes. In Cramer [Cra12], pages 440–457.

[GKM+00] Yael Gertner, Sampath Kannan, Tal Malkin, Omer Reingold, and Mahesh Viswanathan. The relationship between public key encryption and oblivious transfer. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA* [DBL00], pages 325–335.

[GM82] Shafi Goldwasser and Silvio Micali. Probabilistic encryption and how to play mental poker keeping secret all partial information. In Harry R. Lewis, Barbara B. Simons, Walter A. Burkhard, and Lawrence H. Landweber, editors, *Proceedings of the 14th Annual ACM Symposium on Theory of Computing, May 5-7, 1982, San Francisco, California, USA*, pages 365–377. ACM, 1982.

[GMM07] Yael Gertner, Tal Malkin, and Steven Myers. Towards a separation of semantic and CCA security for public key encryption. In Salil P. Vadhan, editor, *Theory of Cryptography, 4th Theory of Cryptography Conference, TCC 2007, Amsterdam, The Netherlands, February 21-24, 2007, Proceedings*, volume 4392 of *Lecture Notes in Computer Science*, pages 434–455. Springer, 2007.

[GMR85] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof-systems (extended abstract). In Robert Sedgewick, editor, *Proceedings of the 17th Annual ACM Symposium on Theory of Computing, May 6-8, 1985, Providence, Rhode Island, USA*, pages 291–304. ACM, 1985.

[GMR01] Yael Gertner, Tal Malkin, and Omer Reingold. On the impossibility of basing trapdoor functions on trapdoor predicates. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, pages 126–135. IEEE Computer Society, 2001.

[GMW91] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity for all languages in NP have zero-knowledge proof systems. *J. ACM*, 38(3):691–729, 1991.

[GSV98] Oded Goldreich, Amit Sahai, and Salil Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 399–408. ACM, 1998.

[GT00] Rosario Gennaro and Luca Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA* [DBL00], pages 305–313.

[HH09] Iftach Haitner and Thomas Holenstein. On the (im)possibility of key dependent encryption. In Omer Reingold, editor, *Theory of Cryptography, 6th Theory of Cryptography Conference, TCC 2009, San Francisco, CA, USA, March 15-17, 2009. Proceedings*, volume 5444 of *Lecture Notes in Computer Science*, pages 202–219. Springer, 2009.

[HHRS15a] Iftach Haitner, Jonathan J Hoch, Omer Reingold, and Gil Segev. Finding collisions in interactive protocols—tight lower bounds on the round and communication complexities of statistically hiding commitments. *SIAM Journal on Computing*, 44(1):193–242, 2015.

[HHRS15b] Iftach Haitner, Jonathan J. Hoch, Omer Reingold, and Gil Segev. Finding collisions in interactive protocols - tight lower bounds on the round and communication complexities of statistically hiding commitments. *SIAM J. Comput.*, 44(1):193–242, 2015.

[HM96] Shai Halevi and Silvio Micali. Practical and provably-secure commitment schemes from collision-free hashing. In *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*, pages 201–215, 1996.

[HR04] Chun-Yuan Hsiao and Leonid Reyzin. Finding collisions on a public road, or do secure hash functions need secret coins? In *Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings*, pages 92–105, 2004.

[IKO05] Yuval Ishai, Eyal Kushilevitz, and Rafail Ostrovsky. Sufficient conditions for collision-resistant hashing. In *Theory of Cryptography, Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, February 10-12, 2005, Proceedings*, pages 445–456, 2005.

[IR89] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 44–61. ACM, 1989.

[Ish11] Yuval Ishai, editor. *Theory of Cryptography - 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28-30, 2011. Proceedings*, volume 6597 of *Lecture Notes in Computer Science*. Springer, 2011.

[Kle06] Jon M. Kleinberg, editor. *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*. ACM, 2006.

[KLW15] Venkata Koppula, Allison Bishop Lewko, and Brent Waters. Indistinguishability obfuscation for turing machines with unbounded memory. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 419–428, 2015.

[KPTZ13] Aggelos Kiayias, Stavros Papadopoulos, Nikos Triandopoulos, and Thomas Zacharias. Delegatable pseudorandom functions and applications. In Ahmad-Reza Sadeghi, Virgil D. Gligor, and Moti Yung, editors, *CCS*, pages 669–684. ACM, 2013.

[KSS11] Jeff Kahn, Michael E. Saks, and Clifford D. Smyth. The dual BKR inequality and rudich's conjecture. *Combinatorics, Probability & Computing*, 20(2):257–266, 2011.

[KST99] Jeong Han Kim, Daniel R. Simon, and Prasad Tetali. Limits on the efficiency of one-way permutation-based hash functions. In *40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA*, pages 535–542. IEEE Computer Society, 1999.

[LV16] Tianren Liu and Vinod Vaikuntanathan. On basing private information retrieval on np-hardness. In Eyal Kushilevitz and Tal Malkin, editors, *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I*, volume 9562 of *Lecture Notes in Computer Science*, pages 372–386. Springer, 2016.

[MM11] Takahiro Matsuda and Kanta Matsuura. On black-box separations among injective one-way functions. In *Theory of Cryptography*, pages 597–614. Springer, 2011.

[MP91] Nimrod Megiddo and Christos H. Papadimitriou. On total functions, existence theorems and computational complexity. *Theor. Comput. Sci.*, 81(2):317–324, 1991.

[MV03] Daniele Micciancio and Salil P. Vadhan. Statistical zero-knowledge proofs with efficient provers: Lattice problems and more. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science*, pages 282–298. Springer, 2003.

[OV08] Shien Jin Ong and Salil P. Vadhan. An equivalence between zero knowledge and commitments. In *Theory of Cryptography, Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008.*, pages 482–500, 2008.

[OW93a] Rafail Ostrovsky and Avi Wigderson. One-way fuctions are essential for non-trivial zero-knowledge. In *ISTCS*, pages 3–17, 1993.

[OW93b] Rafail Ostrovsky and Avi Wigderson. One-way fuctions are essential for non-trivial zero-knowledge. In *ISTCS*, pages 3–17, 1993.

[Pap94] Christos H. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *J. Comput. Syst. Sci.*, 48(3):498–532, 1994.

[Pas06] Rafael Pass. Parallel repetition of zero-knowledge proofs and the possibility of basing cryptography on np-hardness. In *21st Annual IEEE Conference on Computational Complexity (CCC 2006), 16-20 July 2006, Prague, Czech Republic*, pages 96–110. IEEE Computer Society, 2006.

[Pas13] Rafael Pass. Unprovable security of perfect NIZK and non-interactive non-malleable commitments. In *TCC*, pages 334–354, 2013.

[RAD78] R. Rivest, L. Adleman, and M. Dertouzos. On data banks and privacy homomorphisms. In *Foundations of Secure Computation*, pages 169–177. Academic Press, 1978.

[RSA78] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, 1978.

[RTV04] Omer Reingold, Luca Trevisan, and Salil P. Vadhan. Notions of reducibility between cryptographic primitives. In *Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Cambridge, MA, USA, February 19-21, 2004, Proceedings*, pages 1–20, 2004.

[Rud84] Steven Rudich. *Limits on the provable consequences of one-way functions.* PhD thesis, University of California, Berkeley, 1984.

[Rud91] Steven Rudich. The use of interaction in public cryptosystems (extended abstract). In Joan Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91, 11th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1991, Proceedings*, volume 576 of *Lecture Notes in Computer Science*, pages 242–251. Springer, 1991.

[Shm14] David B. Shmoys, editor. *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*. ACM, 2014.

[Sim98] Daniel R Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *Advances in CryptologyâĂŤEUROCRYPT'98*, pages 334–345. Springer, 1998.

[SV03] Amit Sahai and Salil Vadhan. A complete problem for statistical zero knowledge. *Journal of the ACM (JACM)*, 50(2):196–249, 2003.

[SW14a] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In Shmoys [Shm14], pages 475–484.

[SW14b] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In Shmoys [Shm14], pages 475–484.

[Vad99] Salil Pravin Vadhan. *A study of statistical zero-knowledge proofs.* PhD thesis, Massachusetts Institute of Technology, 1999.

[Wat15] Brent Waters. A punctured programming approach to adaptively secure functional encryption. In *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part II*, pages 678–697, 2015.

[WC81] Mark N. Wegman and Larry Carter. New hash functions and their use in authentication and set equality. *J. Comput. Syst. Sci.*, 22(3):265–279, 1981.