Electronic Colloquium on Computational Complexity, Report No. 63 (2024)

One-Way Functions and Zero Knowledge

Shuichi Hirahara[*] Mikito Nanashima[†]

Abstract

The fundamental theorem of Goldreich, Micali, and Wigderson (J. ACM 1991) shows that the existence of a one-way function is sufficient for constructing computational zero knowledge (CZK) proofs for all languages in NP. We prove its converse, thereby establishing characterizations of one-way functions based on the worst-case complexities of zero knowledge. Specifically, we prove that the following are equivalent:

1. A one-way function exists.
2. $\mathsf{NP} \subseteq \mathsf{CZK}$ and NP is hard in the worst case.
3. CZK is hard in the worst case and the problem $\mathsf{GapMCSP}$ of approximating circuit complexity is in CZK.

The characterization above also holds for statistical and computational zero-knowledge argument systems. We further extend this characterization to a proof system with knowledge complexity $O(\log n)$. In particular, we show that the existence of a one-way function is characterized by the worst-case hardness of CZK if $\mathsf{GapMCSP}$ has a proof system with knowledge complexity $O(\log n)$. We complement this result by showing that NP admits an interactive proof system with knowledge complexity $\omega(\log n)$ under the existence of an exponentially hard auxiliary-input one-way function (which is a weaker primitive than an exponentially hard one-way function). We also characterize the existence of a robustly-often nonuniformly computable one-way function by the nondeterministic hardness of CZK under the weak assumption that $\mathsf{PSPACE} \not\subseteq \mathsf{AM}$.

We present two applications of our results. First, we simplify the proof of the recent characterization of a one-way function by NP-hardness of a meta-computational problem and the worst-case hardness of NP given by Hirahara (STOC'23). Second, we show that if NP has a laconic zero-knowledge argument system, then there exists a public-key encryption scheme whose security can be based on the worst-case hardness of NP. This improves previous results which assume the existence of an indistinguishable obfuscation.

[*] National Institute of Informatics, Japan. s_hirahara@nii.ac.jp

[†] Tokyo Institute of Technology, Japan. nanashima@c.titech.ac.jp

ISSN 1433-8092

Contents

1 Introduction

A *one-way function* [DH76] is a polynomial-time computable function that is hard to invert on average. This is arguably the most fundamental cryptographic primitive because the existence of a one-way function is equivalent to the existence of many other cryptographic primitives, such as a pseudorandom generator [HILL99], a pseudorandom function generator [GGM86], a private-key encryption [GM84], a commitment scheme [Nao91], and a digital signature [Rom90]. It is also equivalent to the average-case hardness of various learning tasks, such as PAC learning [BFKL93], learning adaptively changing distributions [NR06], agnostic learning [HN23], and distributional learning [HN23]. Moreover, it is equivalent to the average-case hardness of *meta-computational problems*, i.e., the problems that ask about computational complexity. Examples include the problems of computing time-bounded Kolmogorov complexity [LP20] and time-bounded universal probability [IL90] and the Minimum Circuit Size Problem (MCSP) [IRS22]. A one-way function is the focal point at which cryptography, learning theory, and (meta-)complexity theory meet.

Despite the long list of cryptographic primitives whose existence is equivalent to the existence of a one-way function, there is one exception — zero knowledge — whose relation to a one-way function remains elusive. Goldwasser, Micali, and Rackoff [GMR89] introduced the notion of zero knowledge, which has had fundamental impacts on complexity theory and cryptography. A zero-knowledge proof system for a language L is a system in which a prover convinces an efficient verifier that an input is in L without revealing any other information. The fundamental theorem of Goldreich, Micali, and Wigderson [GMW91] shows that the existence of a one-way function is *sufficient* for constructing zero-knowledge proof systems for all languages in NP, i.e., $\mathsf{NP} \subseteq \mathsf{CZK}$, where CZK denotes the class of promise problems that admit zero-knowledge proof systems. It is a long-standing open question whether a one-way function is *necessary*. Ostrovsky and Wigderson [OW93] showed a partial converse of the theorem of [GMW91]: If CZK is worst-case hard ($\mathsf{CZK} \not\subseteq \mathsf{BPP}$), then there exists an *auxiliary-input one-way function* (AIOWF), which has a conceptually weaker form of one-wayness than the standard one. An auxiliary-input one-way function $f = \{f_x\}_{x\in\{0,1\}^*}$ is a polynomial-time computable function such that for every efficient algorithm A, there exists some auxiliary input x such that $A(x, \text{-})$ fails to invert f_x. They also showed that if CZK is *average-case hard*, then a (standard) one-way function exists. Their results leave as a major open problem the gap between the worst-case complexity of zero-knowledge and the existence of a one-way function. Indeed, Ostrovsky and Wigderson [OW93] noted that it seems impossible to close the gap between worst-case and average-case complexities.[1]

2 Our Results

In this paper, we present new characterizations of the existence of a one-way function by *worst-case complexities* of zero knowledge. There are four variants of zero knowledge: statistical zero-knowledge proof systems (SZK), computational zero-knowledge proof systems (CZK), statistical zero-knowledge argument systems (SZKA), computational zero-knowledge argument systems (CZKA). An *argument system* is a system in which the soundness is guaranteed only against effi-

[1] They wrote "We show that it is possible to obtain an average case complexity result assuming only the nonexistence of uniform one-way functions. On the other hand, it seems that to obtain a worst case complexity result it is impossible to avoid non-uniformity in the definition of one-way function, due to the (non-uniform) input to the proof system." [OW93, page 4]

cient provers. This is in contrast to a *proof system*, in which the soundness is guaranteed to hold for every prover. SZK is the smallest class among the four that is known to be in $\mathsf{AM} \cap \mathsf{coAM}$ [AH91; For89] and thus unlikely to contain NP. For the other three types of proof systems, we characterize the existence of a one-way function as follows.

Theorem 2.1. *For any $\mathfrak{C} \in \{\mathsf{SZKA}, \mathsf{CZK}, \mathsf{CZKA}\}$ and any constant $\epsilon \in (0, 1/2)$, the following are equivalent.*

1. *There exists a one-way function secure against $\mathsf{P/poly}$, i.e., polynomial-size circuits.*
2. $\mathsf{NP} \subseteq \mathfrak{C}$ *and* $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$*.*
3. $\mathsf{Gap}_\epsilon\mathsf{MCSP} \in \mathfrak{C}$ *and* $\mathfrak{C} \not\subseteq \mathsf{i.o.P/poly}$*.*
4. $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U}) \in \mathsf{Avg}\mathfrak{C}$ *and* $\mathfrak{C} \not\subseteq \mathsf{i.o.P/poly}$*, where $\mathcal{U}$ denotes the uniform distribution.*

Here, $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ denotes the approximate version of the Minimum Circuit Size Problem: Given the truth table of a function $f \colon \{0, 1\}^n \to \{0, 1\}$, the task is to decide whether there exists a circuit of size at most $2^{\epsilon n}$ or any circuit that computes f is of size at least $2^{(1-\epsilon)n}$. AvgCZK is the class of distributional problems that can be solved by an errorless average-case CZK scheme.

For $\mathfrak{C} = \mathsf{CZK}$, Theorem 2.1 provides the converse of the aforementioned theorem of Goldreich, Micali, and Wigderson [GMW91]. Note that the existence of a one-way function implies that NP cannot be computed by polynomial-size circuits almost everywhere, i.e., $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$. This fact together with [GMW91] shows Item 1 ⇒ Item 2. We prove its converse, thereby proving that a one-way function is *necessary* to construct zero-knowledge proof systems for all languages in NP.

For $\mathfrak{C} = \mathsf{SZKA}$, Nguyen, Ong, and Vadhan [NOV06] showed that a one-way function is sufficient for constructing statistical zero-knowledge argument systems for all languages in NP, i.e., $\mathsf{NP} \subseteq \mathsf{SZKA}$. Theorem 2.1 shows the converse to their theorem.

A salient feature of Theorem 2.1 is that the statement of Item 2 only involves the fundamental notions defined before the 1980s when the notion of computational zero knowledge was introduced [GMR89], yet it characterizes the fundamental cryptographic primitive introduced by Diffie and Hellman [DH76]. Previously, the existence of a one-way function was characterized in terms of worst-case complexities by introducing new (somewhat artificial) problems based on the notion of computational depth [HN23; LP23] or introducing a new meta-computational problem [Hir23].

We remark that our result and the prior work [OW93] are incomparable to each other. [OW93] does not assume any structure for $\mathfrak{C} \in \{\mathsf{SZKA}, \mathsf{CZK}, \mathsf{CZKA}\}$ than the nontriviality (i.e., the worst-case hardness) but show only a conceptually weak form of one-wayness (i.e., AIOWFs). By contrast, we derive a standard one-way function from the nontriviality of $\mathfrak{C}$ but require an additional structure that $\mathfrak{C}$ contains $\mathsf{Gap}_\epsilon\mathsf{MCSP}$, which is satisfied in the context of the converse of [GMW91].

A key ingredient in the proof is worst-case to average-case reductions. It should be noted that the equivalence between Items 1 and 2 does not involve the notion of meta-complexity, yet the notion of meta-complexity plays an important role: We actually prove a stronger implication from Item 4 to Item 1; that is, we construct a one-way function based on the worst-case hardness of CZK and an average-case CZK scheme for $\mathsf{Gap}_\epsilon\mathsf{MCSP}$. Note that Item 2 implies Item 3 just because $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ is a (promise) problem in NP, and that Item 3 implies Item 4 just because an average-case problem is easier than its worst-case counterpart.

Items 3 and 4 elucidate that the gap between the existence of a one-way function and the worst-case hardness of CZK is due to the lack of zero-knowledge proofs for meta-computational problems. In fact, there is evidence that proving $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U}) \in \mathsf{AvgCZK}$ unconditionally might be within reach of current techniques. Ilango, Ren, and Santhanam [IRS22] showed that $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ is infinitely often in CZK on average with respect to any locally samplable distribution. Note that the notion of average-case complexity in [IRS22] is *error-prone*, meaning that the zero-knowledge proof system does not know when it makes a mistake. In contrast, $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U}) \in \mathsf{AvgCZK}$ refers to the existence of an *errorless* average-case CZK scheme, meaning that the zero-knowledge proof system knows when it makes a mistake; i.e., it must output either a correct answer or a special symbol "$\perp$", which indicates the failure of an algorithm. This gap between errorless and error-prone average-case complexities can be closed if $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ admits an average-polynomial-time instance checker whose query distribution is locally samplable [HS22].

Currently, $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ is an NP-intermediate status [AH19], though many variants of MCSP have been shown to be NP-complete (see [Hir22; Ila23] and references therein). Hirahara [Hir18] showed that the worst-case and average-case complexities of $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ are equivalent for polynomial-time algorithms. This is proved by a non-black-box reduction and hence does not generalize to AvgCZK. Theorem 2.1 shows that the worst-case and average-case complexities of $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ are equivalent for zero-knowledge proof systems under the assumption that $\mathsf{CZK} \not\subseteq \mathsf{i.o.P/poly}$. Even more surprisingly, under the same assumption, it shows that the complexities of NP and $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ are equivalent for CZK in the sense that $\mathsf{NP} \subseteq \mathsf{CZK}$ if and only if $\mathsf{Gap}_\epsilon\mathsf{MCSP} \in \mathsf{CZK}$.

2.1 An Unconditional Study of Computational Knowledge Complexity

Our results raise the following natural question: What is the weakest assumption for characterizing the existence of a one-way function by the worst-case hardness of CZK? Theorem 2.1 identifies the sufficient condition that $\mathsf{Gap}_\epsilon\mathsf{MCSP} \in \mathsf{CZK}$. A natural relaxation of zero knowledge is to consider knowledge complexity not necessarily zero.

Definition 2.2 (Knowledge complexity [GP99][2])**.** *An interactive proof system (P, V) for a promise problem $\Pi = (\Pi_{\mathrm{yes}}, \Pi_{\mathrm{no}})$ has computational knowledge complexity $k(n)$ if there exist a probabilistic polynomial-time algorithm S and a function $\kappa = \{\kappa_n\}$ where $\kappa_n \colon \{0,1\}^n \times \{0,1\}^{\mathsf{poly}(n)} \to \{0,1\}^{k(n)}$ such that for every $x \in \Pi_{\mathrm{yes}}$, the distributions of $S(x, \kappa_{|x|}(x,r); r)$ for $r \sim \{0,1\}^{\mathsf{poly}(|x|)}$ is computationally indistinguishable from the verifier's view of the interaction between P and V on the common input x.*

The case of knowledge complexity being zero corresponds to the standard notion of computational zero knowledge. We strengthen Theorem 2.1 to interactive proof systems with knowledge complexity $O(\log n)$; i.e., we allow a prover to leak $O(\log n)$ bits of information to a verifier.

Theorem 2.3. *The following are equivalent:*

1. *There exists a one-way function secure against* $\mathsf{P/poly}$*.*
2. $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$ *and* NP *has an interactive proof system of computational knowledge complexity $O(\log n)$ with a negligible soundness error.*
3. $\mathsf{CZK} \not\subseteq \mathsf{i.o.P/poly}$ *and* $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ *has interactive proof systems of computational knowledge complexity $O(\log n)$ with a negligible soundness error.*

[2] Our definition is equivalent to that of [GP99] defined in the oracle sense, which is originally defined as the query complexity of the simulator given access to an oracle that provides an advice bit for each access. The equivalence is easily verified by identifying $\kappa(x, r)$ with the (concatenated) answers from the oracle for $S(x; r)$.

Thus, constructing an $O(\log n)$-knowledge proof system for $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ is sufficient for characterizing the worst-case hardness of CZK by the existence of a one-way function. We complement this result by showing that it is possible to construct a proof system with non-trivial knowledge complexity under an assumption weaker than the existence of a one-way function.

Theorem 2.4. *If there exists an auxiliary-input one-way function secure against* $\mathsf{P/poly}$*, then* NP *has an interactive proof system of computational knowledge complexity* n^ϵ *(with negligible completeness and soundness error) for every constant* $\epsilon > 0$*.*

Note that the worst-case hardness of CZK implies the existence of an auxiliary-input one-way function [OW93]. Thus, if the knowledge complexity n^ϵ of Theorem 2.4 were smaller than the knowledge complexity $O(\log n)$ of Theorem 2.3, then we would characterize the existence of a one-way function by the worst-case hardness of CZK. If we assume the exponential hardness of an auxiliary-input one-way function, the gap in the knowledge complexity becomes quite small.

Theorem 2.5. *If there exist a constant* $\epsilon > 0$ *and an auxiliary-input one-way function exponentially secure against* $\mathsf{SIZE}[2^{\epsilon n}]$*, then for every increasing function* $k(n) = \omega(\log n)$*,* NP *has an interactive proof system of computational knowledge complexity* $k(n)$ *(with negligible completeness and soundness error).*

2.2 Nonuniform One-Way Functions and Nondeterministic Hardness of Zero Knowledge

Next, we consider a weaker cryptographic primitive — a robustly-often $\mathsf{P/poly}$-computable one-way function, i.e., a one-way function computable by a polynomial-size circuit. This is an intermediate notion between a one-way function and an auxiliary-input one-way function. Under the weak assumption that $\mathsf{PSPACE} \not\subseteq \mathsf{AM}$, we characterize the existence of a robustly-often $\mathsf{P/poly}$-computable one-way function by the nondeterministic hardness of CZK, such as $\mathsf{i.o.N{\cdot}CZK} \not\subseteq \mathsf{i.o.AM} = \mathsf{i.o.N{\cdot}BPP}$. Here, "$\mathsf{N}\cdot$" denotes the operator of adding an existential quantifier: For a complexity class $\mathfrak{C}$, the class $\mathsf{N}\cdot\mathfrak{C}$ is the class of promise problems that can be accepted by nondeterministic $\mathfrak{C}$-type algorithms. For example, $\mathsf{N{\cdot}P} = \mathsf{NP}$ (see Definition 7.2 for the formal definition of "$\mathsf{N}\cdot$"). One can think of the hardness assumption that $\mathsf{i.o.N{\cdot}CZK} \not\subseteq \mathsf{i.o.AM} = \mathsf{i.o.N{\cdot}BPP}$ as the nondeterministic version of the worst-case hardness of CZK. This assumption is stronger than the worst-case hardness of CZK: $\mathsf{i.o.N{\cdot}CZK} \not\subseteq \mathsf{i.o.N{\cdot}BPP}$ implies $\mathsf{CZK} \not\subseteq \mathsf{BPP}$. Yet, we characterize the *weaker* primitive, a $\mathsf{P/poly}$-computable one-way function, than a one-way function. In the following result, we obtain equivalent statements not only for $\mathfrak{D} = \mathsf{AM}$.

Theorem 2.6. *For every complexity class* $\mathfrak{D}$ *satisfying* $\mathsf{MA} \subseteq \mathfrak{D}$ *and* $\mathsf{PSPACE} \not\subseteq \mathfrak{D}$*, the following are equivalent:*

1. *There exists a robustly-often* $\mathsf{P/poly}$*-computable one-way function.*
2. $\mathsf{PSPACE} \subseteq \mathsf{i.o.CZK/poly}$
3. $\mathsf{PSPACE} \subseteq \mathsf{i.o.N{\cdot}CZK}$
4. $\mathsf{i.o.CZK} \not\subseteq \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$
5. $\mathsf{i.o.N{\cdot}CZK} \not\subseteq \mathsf{i.o.}\mathfrak{D}$

6. $\mathsf{i.o.CZK} \not\subseteq \mathsf{i.o.NP/poly}$

7. $\mathsf{i.o.N{\cdot}CZK} \not\subseteq \mathsf{i.o.MA}$

An intriguing aspect of this result is that it shows the equivalence between the computational power of non-uniform advice "/poly" and the nondeterministic operator "N·" in the sense that $\mathsf{PSPACE} \subseteq \mathsf{i.o.CZK/poly}$ and $\mathsf{PSPACE} \subseteq \mathsf{i.o.N{\cdot}CZK}$ are equivalent under the assumption that $\mathsf{PSPACE} \not\subseteq \mathsf{MA}$.

Consider the case of $\mathfrak{D} = \mathsf{PH}$. In this case, Theorem 2.6 shows the equivalence between $\mathsf{i.o.CZK} \not\subseteq \mathsf{i.o.MA/poly}$ and $\mathsf{i.o.CZK} \not\subseteq \mathsf{i.o.PH/poly}$ under the plausible assumption that $\mathsf{PSPACE} \not\subseteq \mathsf{PH}$. That is, the computational power of $\mathsf{i.o.MA/poly}$ and $\mathsf{i.o.PH/poly}$ is equivalent for computing $\mathsf{i.o.CZK}$.

2.3 Simplifying Hirahara's Characterization of One-Way Functions

Recently, Hirahara [Hir23] captured the existence of a one-way function by NP-hardness of a new meta-complexity problem. The t-time-bounded *distributional Kolmogorov complexity* $\mathsf{dK}^{t,A}_{\lambda}(x \mid \mathcal{D})$ of a string x given a distribution $\mathcal{D}$ with respect to an oracle A is defined as the length of a shortest A-oracle program that prints x in time t on input y with probability at least λ over a random string y drawn from $\mathcal{D}$. The existence of a one-way function was characterized by the worst-case hardness of NP and "structured" NP-hardness of approximating distributional Kolmogorov complexity.

Theorem 2.7 ([Hir23]; informal)**.** *The following are equivalent.*

1. *There exists a one-way function secure against* $\mathsf{P/poly}$*.*

2. $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$ *and for some constant* $\epsilon > 0$*, there exists a parametric-honest*[3] *randomized polynomial-time many-one*[4] *reduction from* NP *to a* $(1+\epsilon)$*-factor approximation of* $\mathsf{dK}^{\tau,A}$ *for all large polynomials* τ *and all oracles* $A \in \mathsf{P/poly}$*.*

The proof of Theorem 2.7 is quite involved, as it combines many ideas developed in the literature of meta-complexity. As an application of our results, we present a simple proof of Item 2 ⇒ 1. The key observation is that the NP-hardness of approximating distributional Kolmogorov complexity implies the existence of a SZKA protocol for NP.

Theorem 2.8. *If there exist a constant* $\epsilon > 0$ *and a parametric-honest randomized polynomial-time many-one reduction from* NP *to a* $(1+\epsilon)$*-factor approximation of* $\mathsf{dK}^{\tau,A}$ *for all large polynomials* τ *and all oracles* $A \in \mathsf{P/poly}$*, then* $\mathsf{NP} \subseteq \mathsf{SZKA}$*.*

Together with Theorem 2.1, Theorem 2.8 immediately implies Item 2 ⇒ 1 in Theorem 2.7. This clarifies that the ideas behind Theorem 2.7 are, in fact, the construction of statistical zero knowledge argument systems for NP.

It is instructive to point out that we do not know how to construct SZKA systems for meta-computational problems themselves, such as $\mathsf{Gap}_{\epsilon}\mathsf{MCSP}$ and the problem of approximating dK. (Constructing SZKA systems for such meta-computational problems is sufficient for characterizing

[3] A reduction to dK is said to be *parametric-honest* if there exists a constant $\gamma > 0$ such that the size parameter s in any query of the reduction on inputs of length n satisfies $s \geq n^{\gamma}$.

[4] The actual result of [Hir23] is stronger than one stated here: the equivalence holds even for nonadaptive reductions that make polynomially many queries. We present a simplified proof for the weak version of the result of [Hir23].

the existence of a one-way function by the worst-case hardness of SZKA because of Theorem 2.1.) Nevertheless, (the proof of) Theorem 2.8 shows that a reduction from a problem Π to meta-computational problems enables us to construct a SZKA system for Π.

2.4 Towards Basing Public-Key Cryptography on the Worst-Case Hardness of NP

As another corollary of our results, we show that a *laconic zero-knowledge argument system* for NP enables us to base the security of a public-key cryptosystem on the worst-case hardness of NP. Informally, a zero-knowledge argument system is said to be *laconic* [GH98; BDRV18] if the number of rounds and the total number of bits sent by a prover are sufficiently smaller than $O(\log n)$ on inputs of length n (see [BDRV18] for the formal definition). Berman, Degwekar, Rothblum, and Vasudevan [BDRV18] showed that constructing laconic zero-knowledge argument systems for NP is sufficient for basing the security of a public-key encryption scheme on the existence of a one-way function. Combining this result with Theorem 2.1, we obtain the following corollary:

Corollary 2.9.
- *If* NP *has a laconic zero-knowledge argument system, then there exists a public-key encryption scheme whose semantic security is based on* $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$.
- *For every* $\epsilon \in (0, 1/2)$*, if* $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ *has a laconic zero-knowledge argument system, then there exists a public-key encryption scheme whose semantic security is based on* $\mathsf{Gap}_\epsilon\mathsf{MCSP} \notin \mathsf{i.o.P/poly}$.

In terms of Impagliazzo's five worlds [Imp95], this shows an approach towards excluding Heuristica, Pessiland, and Minicrypt simultaneously: Constructing a laconic zero-knowledge argument system for an NP-complete problem suffices. Previously, it was known that the existence of an indistinguishable obfuscation suffices. This follows from the work of Komargodski, Moran, Naor, Pass, Rosen, and Yogev [KMNPRY14], which eliminates Heuristica and Pessiland, and Sahai and Waters [SW21], which eliminates Minicrypt, respectively, under the existence of an indistinguishability obfuscation. Since a laconic zero-knowledge argument system can be constructed for NP from an indistinguishability obfuscation [KMNPRY14], our results weaken the previous assumption needed to base the security of a public-key encryption scheme on the worst-case hardness of NP.

3 Proof Techniques

In this section, we mainly present the key ideas to show Theorems 2.1 and 2.3.

Notations. For a distribution $\mathcal{D}$, we use the notation $x \sim \mathcal{D}$ to refer to the sampling of x according to $\mathcal{D}$. For every promise problem $\Pi = (\Pi_{\text{yes}}, \Pi_{\text{no}})$ and every $x \in \Pi_{\text{yes}} \cup \Pi_{\text{no}}$, we define $\Pi(x)$ as

$$\Pi(x) = \begin{cases} 1 & \text{if } x \in \Pi_{\text{yes}}, \\ 0 & \text{if } x \in \Pi_{\text{no}}. \end{cases}$$

3.1 Starting Point: The Case of Statistical Zero-Knowledge Arguments

First, we consider the special case of Theorem 2.1 in which $\mathfrak{C} = \mathsf{SZKA}$, which is the starting point of this work. In this special case, Item 2 $\Rightarrow$ Item 1 of Theorem 2.1 can be shown by a careful combination of two previous results of Ostrovsky [Ost91] and Nanashima [Nan21].

Ostrovsky [Ost91] showed that the average-case hardness of SZK implies the existence of a one-way function. This is proved by a reduction R_{Ost} from any problem in SZK on input x to the task of inverting an auxiliary-input one-way function f_x on auxiliary input x. Nanashima [Nan21] showed that if NP is reducible to the task of inverting an auxiliary-input one-way function, then the security of a one-way function can be based on the worst-case hardness of NP. Combining the ideas behind these two results,[5] it is possible to prove the following: If $\mathsf{NP} \subseteq \mathsf{SZK}$ and $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$, then there exists a one-way function. However, it is unlikely that SZK contains NP, and Theorem 2.1 differs from this in that we assume $\mathsf{NP} \subseteq \mathsf{SZKA}$ instead of $\mathsf{NP} \subseteq \mathsf{SZK}$.

To prove Theorem 2.1 for $\mathfrak{C} = \mathsf{SZKA}$, we need to extend the two results of [Ost91; Nan21] as follows:

1. Applying the reduction R_{Ost} of Ostrovsky [Ost91] to SZKA, we show that if there exists a worst-case hard problem Π in SZKA, then there exists an auxiliary-input one-way function f. Note that this is not a (black-box) reduction from Π to inverting f anymore: We can still prove that Π can be solved by the reduction R^A_{Ost} together with an *efficient* algorithm A that inverts f; however, if A is not efficient, the reduction is not guaranteed to work correctly. This is because we leverage the soundness of SZKA, which holds only for *efficient provers*.

2. To use the result of Nanashima [Nan21], we need a reduction from NP to the task of inverting an auxiliary-input one-way function. In fact, as shown by Hirahara [Hir23], Nanashima's results can be extended to a certain type of *non-black-box* reductions, i.e., reductions that are only guaranteed to work correctly if an oracle is efficient. In particular, the result of Nanashima [Nan21] can be combined with the non-black-box reduction R_{Ost} of Ostrovsky [Ost91] applied to SZKA.

3.2 Generalizing Nanashima's Reduction

Extending the proof ideas in Section 3.1 to the case of $\mathfrak{C} = \mathsf{CZK}$ is highly nontrivial. A natural attempt would be to replace [Ost91] with [OW93]: Ostrovsky and Wigderson [OW93] extended the result of Ostrovsky [Ost91] to the case of CZK by showing that if CZK is worst-case hard, then there exists an auxiliary-input one-way function. This is proved by a certain type of non-black-box reductions, and if this could be combined with [Nan21], then we would be done. Unfortunately, in [OW93], the assumption of efficient inversion was applied in various ways: For example, a polynomial-time-computable function constructed from another efficient inverter needs to be inverted [OW93, Theorems B3 and 7]. Thus, the proof of [OW93] does not yield a reduction to inverting a particular function f; defining a function f to be inverted requires the assumption of the nonexistence of one-way functions. It is unclear whether such a proof can be combined with Nanashima's result.

To prove Theorem 2.1 for $\mathfrak{C} \in \{\mathsf{CZK}, \mathsf{CZKA}\}$ (as well as $\mathfrak{C} = \mathsf{SZK}$), we generalize Nanashima's reduction in an *instance-dependent* fashion, as proposed in the unconditional study of computational zero knowledge by Vadhan [Vad06]. For this purpose, we introduce two key properties of promise problems, which are sufficient for Nanashima's reduction to go through.

[5] The result of [Nan21] is stated only for *nonadaptive* reductions, whereas the reduction R_{Ost} is adaptive. However, by inspecting the proof of [Nan21], one can observe that the reduction can be adaptive in the result of [Nan21], as long as the reduction does not make adaptive queries to auxiliary inputs; thus, the result of [Nan21] is applicable to the reduction R_{Ost} of [Ost91].

To motivate the two properties, we briefly review the proof of Nanashima [Nan21], which consists of the following three steps:

1. Reduce an NP-hard language L to the task of inverting an auxiliary-input function $\{f_z\}_{z\in\{0,1\}^*}$;
2. Reduce the task of inverting $\{f_z\}_{z\in\{0,1\}^*}$ to the task of solving a distributional problem $(L, \mathcal{D})$ in the errorless setting for a samplable distribution $\mathcal{D}$;
3. Reduce the task of solving $(L, \mathcal{D})$ on errorless average to the task of inverting a polynomial-time-computable function f' on average.

The assumption that NP is reducible to inverting an auxiliary-input function is used in both steps 1 and 3. We introduce a new property, called a *BBR/OWF* property, that is sufficient for carrying out these steps. We also introduce AIOWF-hardness in order to carry out step 2. These two properties enable us to generalize Nanashima's results. We now explain the key properties in detail.

Black-Box-Reduction/One-Way-Function Property

The first key property is inspired by the SZK/OWF-characterization for $\mathfrak{C} \in \{\mathsf{SZKA}, \mathsf{CZK}, \mathsf{CZKA}\}$ presented by Vadhan [Vad06] and Ong and Vadhan [OV07]. Roughly speaking, it states that if a promise problem $\Pi = (\Pi_{\mathsf{yes}}, \Pi_{\mathsf{no}})$ is in $\mathfrak{C}$, then there exists a subset $I \subseteq \Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}$ such that (i) I represents *OWF instances* in the sense that there exists a one-way function indexed by an arbitrary $x \in I$ (where x is an auxiliary input) and secure almost everywhere on I; and (ii) $(\Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}) \setminus I$ represents *SZK instances* in the sense that a promise problem $(\Pi_{\mathsf{yes}} \setminus I, \Pi_{\mathsf{no}} \setminus I)$ admits a statistical zero-knowledge proof system. Thus, every instance x of Π provides either (i) a reduction R_{Ost} from solving Π on x to inverting an auxiliary-input function (if x is an SZK instance [Ost91]) or (ii) a one-way function computable with nonuniform advice x (if x is an OWF instance). This observation leads us to the following Black-Box-Reduction/One-Way-Function (BBR/OWF) property of promise problems, which generalizes the SZK/OWF characterization.

Definition 3.1 (BBR/OWF property)**.** *A promise problem Π is said to have a* BBR/OWF property *if there exist two polynomial-time-computable auxiliary-input functions $f = \{f_x\}_{x\in\{0,1\}^*}$, $g = \{g_x\}_{x\in\{0,1\}^*}$, a subset $I \subseteq \Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}$, and a polynomial-time oracle machine R satisfying that*

1. (OWF part) *$\{f_x\}_{x\in I}$ is one-way (secure against $\mathsf{P/poly}$) almost everywhere on I;*
2. (BBR part) *there exists a polynomial p such that for every $x \in (\Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}) \setminus I$ and every oracle A, if $\Pr_w[A(g_x(w)) \notin g_x^{-1}(g_x(w))] \le 1/p(|x|)$, then*

$$\Pr_R\left[R^A(x) = \Pi(x)\right] \ge 2/3.$$

This property enables steps 1 and 3 of Nanashima's reduction. In particular, step 3 is stated as follows.

Lemma 3.2. *Let Π be a promise problem that has the BBR/OWF property. If there exists a samplable distribution family $\mathcal{D} = \{\mathcal{D}_n\}$ on $\Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}$ such that $(\Pi, \mathcal{D}) \notin \mathsf{i.o.AvgP/poly}$ (i.e., $(\Pi, \mathcal{D})$ is errorless average-case hard), then there exists a one-way function secure against $\mathsf{P/poly}$.*

The proof outline of this lemma is as follows: let Π be a promise problem that has the BBR/OWF property with auxiliary-input functions $f = \{f_x\}_x$ and $g = \{g_x\}_x$ and a subset I satisfying the definition of the BBR/OWF property. Let $\mathcal{D} = \{\mathcal{D}_n\}$ be an arbitrary samplable distribution, and let D be its polynomial-time sampler (i.e., $D(1^n, r) \equiv \mathcal{D}_n$ for random seed r). We define a new polynomial-time-computable function $h = \{h_n\}_{n \in \mathbb{N}}$ as $h_n(w, r, r') = (x, f_x(r), g_x(r'))$, where $x = D(1^n, w)$, for random inputs w, r and r'. Notice that inverting h requires inverting both f_x and g_x simultaneously on average over $x \sim \mathcal{D}$.

We prove that h is one-way if $(\Pi, \mathcal{D})$ is errorless average-case hard. When the event $x \in I$ occurs with noticeable probability over $x \sim \mathcal{D}$, then h is (weak) one-way because f_x is one-way on $x \in I$. Thus, we can assume that $\Pr_{x \sim \mathcal{D}_n}[x \in I] \leq 1/p(n)$ for an arbitrarily small polynomial p. In this case, Nanashima's reduction from solving $(\Pi, \mathcal{D})$ on errorless average to inverting h on average works: for every inverting algorithm A implemented as a polynomial-size circuit family, the reduction that is given $x \sim \mathcal{D}_n$ tests whether A succeeds in inverting $(x, f_x(r), g_x(r'))$ with high probability over the choice of r and r'. If so, the reduction executes R^{A_g}, where R is the black-box reduction in the BBR/OWF property, and A_g is an inverting algorithm for g_x induced by A; otherwise, the reduction outputs $\perp$ and halts. By the assumption that $\Pr_{x \sim \mathcal{D}_n}[x \notin I] \geq 1 - 1/p(n)$, for every A that successfully inverts h with high probability, R^{A_g} solves Π on errorless average over $x \sim \mathcal{D}_n$ conditioned on that $x \notin I$. By contrast, when $x \in I$, the efficient adversary A must fail to invert $(x, f_x(r), g_x(r'))$ since f_x is one-way. In this case, the reduction outputs $\perp$, and the probability is bounded above by $1/p(n)$ for an arbitrarily small polynomial p. Therefore, the reduction solves $(\Pi, \mathcal{D})$ on errorless average.

Next, we discuss when a promise problem Π has the BBR/OWF property. It is easy to see that every $\Pi \in \mathsf{CZKA}$ has the BBR/OWF property because of the SZK/OWF characterization [Vad06; OV07]. We can also show that any promise problem with an interactive proof system of computational knowledge complexity $O(\log n)$ has the BBR/OWF property, as we explain later. Thus, the class of promise problems with the BBR/OWF property can be larger than the classes that admit SZK/OWF characterizations. In addition, we observe that, in Lemma 3.2, the reduction R of the BBR/OWF property can be further weakened to an errorless average-case scheme because the goal of step 3 is to solve a distributional problem on errorless average. This observation leads us to an errorless average-case variant of BBR/OWF property. For a formal argument, see Section 5.4.

AIOWF-Hardness

The second key property is *AIOWF-hardness*, which is required for step 2. We define the AIOWF-hardness of a distributional problem $(\Pi, \mathcal{D})$ as the existence of an efficient reduction from the task of inverting every auxiliary-input one-way function to the task of solving $(\Pi, \mathcal{D})$ on errorless average.

Definition 3.3 (AIOWF-hard)**.** *A distributional problem $(\Pi, \{\mathcal{D}_n\})$ is said to be AIOWF-hard if for every polynomial-time-computable auxiliary-input function $f = \{f_x\}_{x \in \{0,1\}^*}$, there exists a randomized polynomial-time oracle machine $R_f^?$ such that for every $n \in \mathbb{N}$ and for every oracle A that solves Π on errorless average under $\mathcal{D}_n$, the reduction R_f^A successfully inverts f_x for all $x \in \{0,1\}^n$.*

The BBR/OWF property and AIOWF-hardness enable steps 1 and 2, respectively. Namely, they provide a (non-black-box) worst-case-to-average-case reduction.

Lemma 3.4. *If a promise problem Π has the BBR/OWF property, and a distributional problem $(\Gamma, \mathcal{D})$ is AIOWF-hard, then $\Pi \notin \mathsf{i.o.P/poly}$ implies $(\Gamma, \mathcal{D}) \notin \mathsf{i.o.AvgP/poly}$, i.e., Γ is hard on errorless average under $\mathcal{D}_n$ for all large $n \in \mathbb{N}$.*

The proof of Lemma 3.4 is outlined as follows: First, we use the BBR/OWF-property and obtain two polynomial-time-computable auxiliary-input functions $f = \{f_x\}_x$ and $g = \{g_x\}$. Then, we construct an auxiliary-input function $h = \{h_x\}$ defined as $h_x(r, r') := (f_x(r), g_x(r'))$ for each x. Next, we use the AIOWF-hardness of $(\Gamma, \mathcal{D})$ with respect to h. Suppose that there exists a nonuniform polynomial-time algorithm A that solves Γ on errorless average under $\mathcal{D}_n$ on infinitely many n for contraposition. Then, by the AIOWF-hardness, we obtain an algorithm $B := R_h^A$, where R_h is the reduction in the definition of AIOWF hardness, that inverts h_x for all $x \in \{0,1\}^n$. Now, B is implemented as a nonuniform polynomial-time algorithm. Therefore, the set I of OWF instances must satisfy $I \cap \{0,1\}^n = \emptyset$; otherwise, B must fail to invert f_x and h_x for some $x \in \{0,1\}^n$ and infinitely many $n \in \mathbb{N}$ (where we use the efficiency of A, and thus our reduction is non-black-box). For each $n \in \mathbb{N}$ with $I \cap \{0,1\}^n = \emptyset$, the algorithm R^{B_g}, where R is the reduction of the BBR/OWF property and B_g is the inverter for g induced by B, correctly solves Π on input size n with high probability by the requirement for the BBR part. Since R^{B_g} is implemented as a nonuniform polynomial-time algorithm, we conclude that $\Pi \in \mathsf{i.o.P/poly}$.

In the previous work [Nan21], the AIOWF-hardness of $(L, \mathcal{D})$ for an NP-hard problem and a samplable distribution $\mathcal{D}$ was implicitly employed. We can also observe that $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U})$ is AIOWF-hard because (intuitively) an errorless algorithm for $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ can distinguish truth-tables of random functions from ones of auxiliary-input pseudorandom functions, which are constructed from an auxiliary-input one-way function [GGM86; HILL99] and thus have small circuit complexity (when the input length to the function is properly chosen). This is the reason why we can replace an NP-hard language with $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ in Theorem 2.1.

Putting It All Together

Theorems 2.1 and 2.3 are proved by combining the BBR/OWF property and the AIOWF-hardness. Here, we show that if $\mathsf{Gap}_\epsilon\mathsf{MCSP} \in \mathsf{CZKA}$ and $\mathsf{CZKA} \not\subseteq \mathsf{i.o.P/poly}$, then there exists a one-way function secure against $\mathsf{P/poly}$ (Item 3 $\Rightarrow$ Item 1 of Theorem 2.1). Let $\Pi \in \mathsf{CZKA} \setminus \mathsf{i.o.P/poly}$. Since $\Pi \in \mathsf{CZKA}$, the promise problem Π has the BBR/OWF property. By the AIOWF-hardness of $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U})$ and Lemma 3.4, $\Pi \notin \mathsf{i.o.P/poly}$ implies that $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U}) \notin \mathsf{i.o.AvgP/poly}$. By Lemma 3.2, this implies the existence of a one-way function secure against $\mathsf{P/poly}$.

3.3 Extension to $O(\log n)$ Computational Knowledge Complexity

In order to extend Theorem 2.1 to an interactive proof system with knowledge complexity $O(\log n)$, it suffices to show that such a system admits the BBR/OWF property.

Theorem 3.5. *If a promise problem Π has an interactive proof system (P, V) of computational knowledge complexity $O(\log n)$ and negligible soundness error, then Π has the BBR/OWF property.*

At a high level, Theorem 3.5 is shown by extending the work of Petrank and Tardos [PT02], who proved that every problem that has an interactive proof system of *statistical* knowledge complexity $O(\log n)$ is contained in $\mathsf{AM} \cap \mathsf{coAM}$. We extend this to the case of *computational* knowledge in an instance-dependent fashion, as in the work of Vadhan [Vad06].

Petrank and Tardos [PT02] proved that for every problem Π that has an interactive proof system (P, V) of *statistical* knowledge complexity $O(\log n)$, if S is a simulator that guesses the knowledge uniformly at random, then Π can be solved on an instance x by estimating the probability that the quantity

$$\log \frac{\Pr[S(x) \text{ produces } \tau]}{\Pr[(P_S, V) \text{ produces } \tau \text{ on input } x]}$$

is small (with respect to the knowledge complexity) over a choice of $\tau \sim S(x)$, where P_S represents the simulation-based prover that returns a message for a current history τ_{hist} of transcripts according to the conditional distribution of $S(x)$ given that the prefix is consistent with τ_{hist}. In [PT02], the quantity above is estimated on average based on an $\mathsf{AM} \cap \mathsf{coAM}$ protocol that approximates the Shannon entropy of a given sampler. In the proof of Theorem 3.5, we first introduce a measure that determines whether S simulates the real conversation (i) statistically or (ii) computationally (and not statistically), given the condition that *the random prediction of knowledge is correct*. In the former case (i), we employ a strategy similar to [PT02], but instead of the $\mathsf{AM} \cap \mathsf{coAM}$ protocol, we use the average-case estimator of a probability that can be constructed by a black-box reduction to inverting a one-way function [IL90; HN23]. In the latter case (ii), we show that the simulator S yields false entropy [HILL99], which is known to imply the existence of a pseudorandom generator and a one-way function. Thus, we obtain the BBR/OWF property for Π, where the BBR and OWF parts correspond to the cases (i) and (ii), respectively. For the formal argument, see Section 5.7.

3.4 Quick Tours of Ideas for Remaining Theorems

In this section, we briefly present ideas for proving other results.

3.4.1 Theorems 2.4 and 2.5: Saving Computational Knowledge Complexity

We present the idea for saving computational knowledge complexity based on an auxiliary-input one-way function.

In the first place, why is it difficult to construct zero-knowledge proof systems based on auxiliary-input one-way function $f = \{f_x\}_x$? The main difficulty is that auxiliary inputs making f one-way depend on adversaries. To deal with this issue, we may use the small-support Min-Max theorem [LY94] (Theorem 6.2), which ensures the existence of a uniform distribution $\mathcal{P}$ over an exponential-size multi-set of auxiliary inputs such that f_x is one-way for all polynomial-size adversaries when $x \sim \mathcal{P}$.

This idea leads us to the following simple strategy: The prover first sends an auxiliary input $x \sim \mathcal{P}$ (by consuming unbounded computational resources) to the verifier and then both parties execute the zero-knowledge protocol [GMW91] based on the auxiliary-input one-way function f_x. Then the knowledge complexity is bounded above by the length of the auxiliary input x, where the knowledge mapping κ (from a pair of an instance and a random seed to advice) just returns the auxiliary input indicated by the random seed. Moreover, when the auxiliary-input one-way function is exponentially hard, the prover can use a shorter auxiliary input for polynomial security in the length of the common input, which reduces the knowledge complexity. Using this idea, we will show the following meta-theorem.

Theorem 3.6. *Let $s, k \colon \mathbb{N} \to \mathbb{N}$ be functions satisfying that $s(k(n)) = n^{\omega(1)}$. If there exists an auxiliary-input one-way function with sufficiently large security against* $\mathsf{SIZE}[\mathsf{poly}(s(m))]$ *with* *success probability at most* $\mathsf{negl}(s(m))$ *(where* m *represents the length of auxiliary inputs), then* NP *has an interactive proof system of computational knowledge complexity* $k(n)$ *and negligible soundness error.*

Theorems 2.4 and 2.5 are derived from this result by choosing $s(n)$ and $k(n)$ appropriately. For details, see Section 6.

3.4.2 Theorem 2.6: Non-Deterministic Hardness of ZK

We present the idea for showing Theorem 2.6. We focus on the following implication.

Lemma 3.7 (Item 1 ⇒ Item 7)**.** *If there is no robustly-often* $\mathsf{P/poly}$*-computable one-way function, then* $\mathsf{i.o.N{\cdot}CZK} \subseteq \mathsf{i.o.MA}$*.*

Note that Item 1 ⇒ Item 6 can be proved in a similar way, and the other implications in Theorem 2.6 follow from the known construction of zero-knowledge proof systems [GMW91] and the hardness assumptions (see Section 7).

It is not difficult to prove a slightly weaker conclusion that $\mathsf{i.o.N{\cdot}CZK} \subseteq \mathsf{i.o.AM}$. This follows from the work of Vadhan [Vad06, Theorem 7.4], who proved that $\mathsf{CZK} \subseteq \mathsf{HV\text{-}CZK} = \mathsf{HV\text{-}SZK} \subseteq \mathsf{AM}$ under the non-existence of robustly-often $\mathsf{P/poly}$-computable one-way functions. Thus, we obtain $\mathsf{i.o.N{\cdot}CZK} \subseteq \mathsf{i.o.N{\cdot}AM} = \mathsf{i.o.N{\cdot}N{\cdot}BPP} = \mathsf{i.o.N{\cdot}BPP}$.

To prove $\mathsf{i.o.N{\cdot}CZK} \subseteq \mathsf{i.o.MA}$, we use nondeterminism to guess an adversary. First, we show that the non-existence of robustly-often $\mathsf{P/poly}$-computable one-way functions implies that there exists a polynomial p such that every (multi-output) circuit C of size s can be inverted by a circuit I_C of size $p(s)$. Under the the non-existence of robustly-often $\mathsf{P/poly}$-computable one-way functions, for every $\Pi \in \mathsf{i.o.N{\cdot}CZK}$, there exists a $\Gamma \in \mathsf{CZK} = \mathsf{HV\text{-}SZK}$ such that for infinitely many $n \in \mathbb{N}$ and for every $x \in \{0,1\}^n$,

- If $x \in \Pi_{\mathrm{yes}}$, then there exists $w \in \{0,1\}^{\mathsf{poly}(n)}$ such that $(x, w) \in \Gamma_{\mathrm{yes}}$;
- If $x \in \Pi_{\mathrm{no}}$, then for all $w \in \{0,1\}^{\mathsf{poly}(n)}$, it holds that $(x, w) \in \Gamma_{\mathrm{no}}$.

For every (x, w), determining whether $(x, w) \in \Gamma_{\mathrm{yes}}$ or $(x, w) \in \Gamma_{\mathrm{no}}$ is reducible to inverting an auxiliary-input function $f_{x,w}$ (interpreted as a circuit embedded x and w) by the argument of [Ost91]. Thus, for each $x \in \{0,1\}^n$, we use nondeterminism to guess w and an inverter $I_{x,w}$, check whether $I_{x,w}$ successfully inverts $f_{x,w}$, and if so, we execute the reduction to determine whether $(x, w) \in \Gamma_{\mathrm{yes}}$ or $(x, w) \in \Gamma_{\mathrm{no}}$. This algorithm can be implemented in MA, and thus we conclude that $\Pi \in \mathsf{i.o.MA}$. For a formal proof, see Section 7.

3.4.3 Theorem 2.8: NP-Hardness of MdKP and Zero-Knowledge

We explain how to construct a statistical zero-knowledge argument system by using NP-hardness of approximating the distributional Kolmogorov complexity $\mathsf{dK}(x \mid \mathcal{D})$. As in [Hir23], we use the k-wise direct product generator $\mathsf{DP}_k(x; z)$, which is defined as $\mathsf{DP}_k(x; z) := \left(z, \langle x, z^1\rangle_{\mathbb{F}_2} \circ \cdots \circ \langle x, z^k\rangle_{\mathbb{F}_2}\right) \in \{0,1\}^{|x|k} \times \{0,1\}^k$, where $z = z^1 \circ \cdots \circ z^k$, $|z^i| = |x|$ for each i, and $\langle , \rangle_{\mathbb{F}_2}$ represents the inner product in $\mathbb{F}_2$. Let R be a reduction from an NP-complete problem L to the problem $\mathsf{GapMdKP}$ of approximating distributional Kolmogorov complexity.

The SZKA system (P, V) for L is extremely simple and operates as follows on input ξ:

1. The verifier V runs the reduction R on input ξ to obtain a random instance $(x, \mathcal{D})$ of $\mathsf{GapMdKP}$ produced by R, and samples a string y from the distribution $\mathcal{D}$. Next, V chooses a secret bit $b \sim \{0,1\}$. If $b = 0$, then V sends (y, w) to the prover for a uniformly random string w. Otherwise, V sends $(y, \mathsf{DP}_k(x; z))$ to the prover for a uniformly random string z.

2. The prover P receives (y, w) and sends back a bit $b' \in \{0,1\}$ such that $b' = 1$ if and only if the conditional Kolmogorov complexity of w given y is small (i.e., w is compressible using y).

3. The verifier V accepts ξ if and only if $b' = b$.

The intuition behind this argument system is as follows:

- If $\xi \in L$, then the reduction R produces a Yes instance $(x, \mathcal{D})$ of $\mathsf{GapMdKP}$; i.e., $\mathsf{dK}^{\mathsf{poly}}(x \mid \mathcal{D})$ is small. Thus, there exists a small program that prints x given $y \sim \mathcal{D}$, which implies that the conditional Kolmogorov complexity of $\mathsf{DP}_k(x; z)$ given y is also small.
- If $\xi \notin L$, then the reduction R produces a No instance $(x, \mathcal{D})$ of $\mathsf{GapMdKP}$. If an efficient prover P can distinguish (y, w) from $(y, \mathsf{DP}_k(x; z))$, then by the reconstruction property of DP_k, we would get an upper bound on $\mathsf{dK}^{\mathsf{poly}}(x \mid \mathcal{D})$ using P, which is a contradiction.

Details can be found in Section 8.

4 Preliminaries

All logarithms are base 2 unless specified otherwise. We use ε to represent an empty symbol. We distinguish ε from ϵ and often use ϵ for an accuracy parameter. Let $\langle , \rangle$ be a (standard) paring function that maps $\mathbb{N} \times \mathbb{N}$ to $\mathbb{N}$.

We use the notation negl to represent some negligible function, i.e., for any polynomial p and sufficiently large $n \in \mathbb{N}$, it holds that $\mathsf{negl}(n) < 1/p(n)$. We also use the notation poly to refer to some polynomial.

For each $n \in \mathbb{N}$, let $[n] := \{1, 2, \ldots, n\}$. For every $x, y \in \{0,1\}^*$, let $x \circ y$ denote the concatenation of x and y. For readability, we may omit $\circ$ from $x \circ y$. For each $x \in \{0,1\}^n$ and each $i \in [n]$, we let x_i denote the i-th bit of x. For every $f \colon \{0,1\}^n \to \{0,1\}$, let $\mathsf{tt}(f)$ be the truth table of f represented as the string of length 2^n.

For each $n \in \mathbb{N}$, we let $\mathcal{U}_n$ denote the uniform distribution over $\{0,1\}^n$ or a random variable selected uniformly at random from $\{0,1\}^n$ in context. For any distribution $\mathcal{D}$, we use the notation $x \sim \mathcal{D}$ to refer to the sampling of x according to D. For any finite set S, we use the notation $x \sim S$ to refer to the uniform sampling of x from S. For each distribution $\mathcal{D}$ and each $x \in \{0,1\}^*$, let $\mathcal{D}(x) = \Pr_{y \sim D}[y = x]$.

For a function $f \colon \{0,1\}^n \to \{0,1\}^m$ and $y \in \mathrm{Im} f$, we define $\mathsf{UnifInv}_f(y)$ as the uniform distribution over $f^{-1}(y) = \{x \in \{0,1\}^n : f(x) = y\}$.

For any distribution $\mathcal{D}$, let $H(\mathcal{D})$ denote the Shannon entropy of $\mathcal{D}$. For any distributions $\mathcal{D}$ and $\mathcal{E}$, let $\Delta_{\mathsf{TV}}(\mathcal{D}, \mathcal{E})$ denote the total variation distance between $\mathcal{D}$ and $\mathcal{E}$. Let $\mathrm{KL}(\mathcal{D} || \mathcal{E})$ represent the KL divergence between two distributions $\mathcal{D}$ and $\mathcal{E}$.

For an infinite set $S \subseteq \{0,1\}^*$ and two distribution families $\mathcal{D} = \{\mathcal{D}_x\}_{x \in S}$ and $\mathcal{E} = \{\mathcal{E}_x\}_{x \in S}$, we say that $\mathcal{D}$ and $\mathcal{E}$ are statistically indistinguishable if $\Delta_{\mathsf{TV}}((x, \mathcal{D}_x), (x, \mathcal{E}_x)) \leq \mathsf{negl}(|x|)$ for each

$x \in S$. We say that $\mathcal{D}$ and $\mathcal{E}$ are computationally indistinguishable if for every polynomial-size circuit A, every polynomial p, and for all but finitely many $x \in S$,

$$\left| \Pr_{z \sim \mathcal{D}_x}[A(x,z)=1] - \Pr_{z \sim \mathcal{E}_x}[A(x,z)=1] \right| \leq 1/p(|x|).$$

For each $t \colon \mathbb{N} \to \mathbb{N}$, we say that a family $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ of distributions (on binary strings) is $t(n)$-time samplable if there exists a $t(n)$-time deterministic algorithm D (called a sampling algorithm or a sampler for $\mathcal{D}$) such that, for each $n \in \mathbb{N}$, the distribution of $D(1^n, \mathcal{U}_{t(n)})$ is statistically identical to $\mathcal{D}_n$. We say that a family $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ of distributions is (polynomial-time) samplable if $\mathcal{D}$ is $p(n)$-samplable for some polynomial $p(n)$.

For every randomized algorithm A using $s(n)$ random bits on an n-bit input, we use the notation $A(x;r)$ to refer to the execution of $A(x)$ with a random tape r for each $x \in \{0,1\}^n$ and $r \in \{0,1\}^{s(n)}$.

For a promise problem Π, we use the notation Π_{yes} (resp. Π_{no}) to refer to the set of yes (resp. no) instances, i.e., $\Pi = (\Pi_{\mathrm{yes}}, \Pi_{\mathrm{no}})$.

4.1 Computational Complexity

For each $n \in \mathbb{N}$ and $x \in \{0,1\}^{2^n}$, we define the circuit complexity $\mathsf{cc}(x)$ of x as the minimum size of an n-input circuit whose truth table corresponds to x. We define the language MCSP as

$$\mathsf{MCSP} = \{(x, 1^s) : n, s \in \mathbb{N}, x \in \{0,1\}^{2^n}, \mathsf{cc}(x) \leq s\}.$$

For a constant $\epsilon \in (0, 1/2)$, $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ is a promise problem (Π_Y, Π_N) defined as $\Pi_Y = \{x \in \{0,1\}^{2^n} : n \in \mathbb{N}, \mathsf{cc}(x) \leq 2^{\epsilon n}\}$ and $\Pi_N = \{x \in \{0,1\}^{2^n} : n \in \mathbb{N}, \mathsf{cc}(x) > 2^{(1-\epsilon)n}\}$. In this work, we fix the constant $\epsilon \in (0,1/2)$ arbitrarily and omit the subscript ϵ from $\mathsf{Gap}_\epsilon\mathsf{MCSP}$.

We define a distributional problem as a pair of a promise problem Π and a distribution $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ on instances. For simplicity, we always assume that $\mathrm{supp}(\mathcal{D}) \subseteq \Pi_{\mathrm{yes}} \cup \Pi_{\mathrm{no}}$ in this work. Note that our results also hold when $\mathrm{supp}(\mathcal{D}) \not\subseteq \Pi_{\mathrm{yes}} \cup \Pi_{\mathrm{no}}$, where we regard every output as correct for an instance $x \notin \Pi_{\mathrm{yes}} \cup \Pi_{\mathrm{no}}$. For convenience, we omit the description "$= \{\mathcal{D}_n\}_{n\in\mathbb{N}}$" from $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ when the intention is clear in context.

We say that an algorithm A solves a promise problem Π on errorless average over $\mathcal{D}$ with failure probability $\delta \in (0,1)$ if (1) A outputs $\Pi(x)$ or $\perp$ (which represents "failure") for every $x \in \mathrm{supp}(\mathcal{D})$, and (2) the failure probability that $A(x)$ outputs $\perp$ over the choice of $x \sim \mathcal{D}$ is bounded above by δ. We say that a distributional problem $(\Pi, \mathcal{D})$ has an infinitely-often *errorless* heuristic algorithm A with failure probability $\delta \colon \mathbb{N} \to (0,1)$ if for infinitely many $n \in \mathbb{N}$, the algorithm A solves a promise problem Π on errorless average over $\mathcal{D}_n$ with failure probability $\delta(n)$. Let $\mathsf{i.o.Avg}_\delta\mathsf{P/poly}$ be the class of distributional problems that have an infinitely-often *errorless* heuristic algorithm A with failure probability δ implemented as a polynomial-size circuit family.

4.2 Zero-Knowledge Proofs and Arguments

We formally introduce zero-knowledge proof systems and its variants.

An interactive protocol (A, B) is a pair of algorithms that compute *a next-message function* that maps a common input x and a transcript τ and auxiliary-input advice α to the next message m. A transcript τ is initialized as the empty string ε, and the computation of each party is executed alternatively. Whenever a message m is computed, it is added to the current transcript τ, and we regard this as that a party sends message m to the other party. One party can terminate the interaction by outputting the resulting message (0 or 1 by default). For simplicity, we regard a resulting message as a part of a transcript. We use the notation $\langle A(\alpha), B(\beta)\rangle(x)$ to represent the resulting message when (i) A is given auxiliary-input advice α, (ii) B is given auxiliary-input advice β, and (iii) both A and B are given the common input x. A party can be randomized. We define B's view (denoted by $\mathsf{view}_B(\langle A(\alpha), B(\beta)\rangle(x))$) of the interaction $\langle A(\alpha), B(\beta)\rangle(x)$ as the 4-tuple of the common input x, the auxiliary-input advice β, the final transcript τ, and the internal randomness used by B.

Definition 4.1 (Interactive proof/argument systems)**.** *Let $c, s \colon \mathbb{N} \to [0,1]$ be functions satisfying that $1 - c(n) \geq s(n) + 1/\mathsf{poly}(n)$. An interactive protocol (P, V) is an interactive proof system for a promise problem Π (with completeness error $c(\cdot)$ and soundness error $s(\cdot)$) if the following requirements are satisfied:*

- (*Efficiency*) *The round of (P, V) is polynomially bounded in the length of common input, and V is efficiently computable (in probabilistic polynomial time by default) and outputs a resulting message.*
- (*Completeness*) *For every $x \in \Pi_{\text{yes}}$, $\langle P, V\rangle(x) = 1$ with probability at least $1 - c(|x|)$.*
- (*Soundness*) *For every $x \in \Pi_{\text{no}}$ and every algorithm P^*, $\langle P^*, V\rangle(x) = 0$ with probability at least $1 - s(|x|)$.*

We also define an interactive argument system (P, V) in the same manner except that the soundness is relaxed as follows:

- (*Computational soundness*) *For every $x \in \Pi_{\text{no}}$ and every* nonuniform polynomial-time *algorithm P^*, $\langle P^*, V\rangle(x) = 0$ with probability at least $1 - s(|x|)$.*

The party P (resp. V) is referred to as a prover (resp. verifier). In this work, we consider negligible completeness and soundness error unless otherwise stated.

Next, we introduce the formal definition of zero-knowledge discussed in this work.

Definition 4.2 (Zero-knowledge)**.** *An interactive proof* (*or argument*) *system (P, V) for a promise problem Π is said to be statistically* (*resp. computationally*) *zero-knowledge if for every polynomial-time randomized verifier V^*, there exists a polynomial-time randomized algorithm S^* such that $\{S(x, \alpha)\}_{x,\alpha}$ is statistically* (*computationally*) *indistinguishable from $\{\mathsf{view}_{V^*}(\langle P, V^*(\alpha)\rangle(x))\}_{x,\alpha}$ for $x \in \Pi_{\text{yes}}$ and advice $\alpha \in \{0,1\}^{\mathsf{poly}(|x|)}$.*

Moreover, (P, V) is said to be honest-verifier statistical (*resp. computational*) *zero-knowledge if it satisfies the condition for statistical* (*resp. computational) zero-knowledge for V* (*instead of polynomial-time randomized verifiers V^**). *We call the simulator with respect to V an* honest-verifier *simulator.*

We define SZK, SZKA, CZK, and CZKA as the classes of promise problems that have a statistical zero-knowledge interactive proof system, a statistical zero-knowledge interactive argument system, a computational zero-knowledge interactive proof system, and a computational zero-knowledge interactive argument system, respectively. For each $\mathfrak{C} = \{\mathsf{SZK}, \mathsf{SZKA}, \mathsf{CZK}, \mathsf{CZKA}\}$, we also define $\mathsf{HV}\text{-}\mathfrak{C}$ in the same manner as $\mathfrak{C}$ except that we consider *honest-verifier* zero-knowledge.

Theorem 4.3 ([OV07])**.** *For every $\Pi \in \mathsf{NP}$, $\Pi \in \mathsf{HV}\text{-}\mathsf{SZKA}$ if and only if $\Pi \in \mathsf{SZKA}$*

For zero-knowledge interactive proof (or argument) system and its honest-verifier simulator S, we define the simulation-based prover (resp. verifier) as a specific prover P_S (resp. verifier V_S) that computes a message m on common input x and a current transcript τ according to the distribution of $S(x)$ conditioned on the event that the prefix of $S(x)$ corresponds to τ.

In this work, we assume that verifiers always send its internal randomness before making the decision, which does not affect the completeness and soundness. This trick enables us to identify the verifier's view (containing the verifier's internal randomness) with a transcript.

We say that a transcript τ is valid if it is consistent with some possible verifier's view. In addition, we say that τ is an accepting transcript if the verifier's decision is acceptance in τ. In this work, we assume that any simulator S always outputs a valid and accepting transcript; otherwise, we slightly modify the verifier so that it accepts the input when its randomness is $0^{\mathsf{poly}(n)}$ and make S output the trivial transcript for that randomness $0^{\mathsf{poly}(n)}$ whenever it produces invalid one (it affects the soundness and the quality of the simulation only with negligible probability). In addition, without loss of generality, for each input x, we assume that an interactive proof (argument) system $\langle P, V\rangle(x)$ exchanges exactly $2 \cdot \ell(|x|) = \mathsf{poly}(|x|)$ messages, and P's (resp. V's) messages are sent at odd (resp. even) rounds. Moreover, we assume that $S(x)$ uses $r_S(|x|) = \mathsf{poly}(|x|)$ random bits.

Finally, we introduce the notion of efficient provers in the case of $\Pi \in \mathsf{NP}$.

Definition 4.4 (Efficient prover)**.** *Let (P, V) be an interactive proof (or argument) system for $\Pi \in \mathsf{NP}$, and let R be the NP-relation for Π (i.e., $x \in \Pi_{\mathsf{yes}}$ if and only if $\exists w \in \{0,1\}^{\mathsf{poly}(|x|)}$ s.t. $(x, w) \in R$). The prover P is said to be efficient if for every $x \in \Pi_{\mathsf{yes}}$, the prover P halts in polynomial time when x is given as common input and its witness w such that $(x, w) \in R$ is given as auxiliary-input advice for P.*

4.3 Cryptography

In this work, we consider nonuniform randomized polynomial-time algorithms as a class of adversaries by default.

Definition 4.5 (One-way function)**.** *A polynomial-time-computable function $f = \{f_n \colon \{0,1\}^{\mathsf{poly}(n)} \to \{0,1\}^{\mathsf{poly}(n)}\}_{n\in\mathbb{N}}$ is said to be a one-way function secure against a class $\mathcal{C}$ of adversaries if for every adversary A in $\mathcal{C}$ and for every sufficiently large $n \in \mathbb{N}$,*

$$\Pr\left[f_n(A(1^n, f_n(\mathcal{U}_{\mathsf{poly}(n)}))) = f_n(\mathcal{U}_{\mathsf{poly}(n)})\right] < \mathsf{negl}(n).$$

The parameter n in the definition above is often called a *security parameter*. We may omit the subscript n from f_n and 1^n from the input to adversaries for readability.

Next, we introduce an auxiliary-input variant of one-way functions, introduced by Ostrovsky and Wigderson [OW93]. Roughly speaking, auxiliary-input primitives are defined as a collection of candidates for secure primitives indexed by an auxiliary input $z \in \{0,1\}^*$ and have a relaxed security condition that for each adversary A, there exists an auxiliary input $z_A \in \{0,1\}^*$ depending on A such that the primitive indexed by z_A is secure for A. In this work, we discuss the sufficiently large security of auxiliary-input one-way functions.

We define an auxiliary-input function as a function family $f = \{f_z\}_{z\in\{0,1\}^*}$ indexed by binary strings z. We say that f is polynomial-time computable if each $f_z(x)$ is polynomial-time computable from (z, x).

Definition 4.6 (Auxiliary-input one-way function)**.** *A polynomial-time-computable auxiliary-input function $f = \{f_z \colon \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z\in\{0,1\}^*}$ is said to be an auxiliary-input one-way function secure against a class $\mathcal{C}$ of adversaries if for every adversary A in $\mathcal{C}$ and for every sufficiently large $n \in \mathbb{N}$, there exists $z := z_{n,A} \subseteq \{0,1\}^n$ such that*

$$\Pr\left[f_z(A(z, f_z(\mathcal{U}_{\mathsf{poly}(n)}))) = f_z(\mathcal{U}_{\mathsf{poly}(n)})\right] < \mathsf{negl}(n).$$

Moreover, we say that f is one-way almost everywhere on $I \subseteq \{0,1\}^*$ if for every nonuniform polynomial-time adversary A and for all but finitely many $z \in I$,

$$\Pr\left[f_z(A(z, f_z(\mathcal{U}_{\mathsf{poly}(|z|)}))) = f_z(\mathcal{U}_{\mathsf{poly}(|z|)})\right] < \mathsf{negl}(|z|).$$

For simplicity, when we consider a circuit family $\{A_n\}_{n\in\mathbb{N}}$ that tries to invert a polynomial-time-computable function family $\{f_z\}_{z\in\{0,1\}^*}$, we measure the circuit complexity of A as a function in $|z|$, instead of the actual length of the pair of z and the output of f_z.

We also define a public-key encryption scheme with semantic security.

Definition 4.7 (public-key encryption)**.** *A semantic secure public-key encryption scheme is a triple* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *of polynomial-time randomized algorithms satisfying the following:*

- (*Syntax*) *For every $\lambda \in \mathbb{N}$, the algorithm* $\mathsf{Gen}(1^\lambda)$ *outputs a pair* $(\mathsf{sk}, \mathsf{pk})$ *of strings.*
- (*Correctness*) *For every $\lambda \in \mathbb{N}$ and every $m \in \{0,1\}$, and for* $(\mathsf{sk}, \mathsf{pk}) \sim \mathsf{Gen}(1^\lambda)$*,*

  $$\Pr[\mathsf{Dec}(1^\lambda, \mathsf{sk}, \mathsf{Enc}(1^\lambda, \mathsf{pk}, m)) = m] \geq 1 - \mathsf{negl}(\lambda),$$

  where the probability is taken over the choice of randomness for $\mathsf{Gen}, \mathsf{Enc}$*, and* Dec*.*
- (*Semantic security*) *For every nonuniform polynomial-time algorithm A and sufficiently large $\lambda \in \mathbb{N}$,*

  $$\left|\Pr[A(1^\lambda, \mathsf{pk}, \mathsf{Enc}(1^\lambda, \mathsf{pk}, 0)) = 1] - \Pr[A(1^\lambda, \mathsf{pk}, \mathsf{Enc}(1^\lambda, \mathsf{pk}, 1)) = 1]\right| \leq \mathsf{negl}(\lambda),$$

  where the probability is taken over the choice of randomness for Gen *and* Enc*.*

4.4 Facts from Information Theory

We introduce some lemmas.

Lemma 4.8 ([cf. Vad06, Proof of Lemma 3.10])**.** *Let $\{(\mathcal{X}_x, \mathcal{Y}_x)\}_{x\{0,1\}^*}$ be a family of efficiently samplable joint distributions. For a subset $I \subseteq \{0,1\}^*$, if there exists a polynomial p such that for every $x \in I$, there exists a (possibly not efficiently samplable) joint distribution $(\mathcal{X}'_x, \mathcal{Y}'_x)$ satisfying*

$$H(\mathcal{X}'_x|\mathcal{Y}'_x) - H(\mathcal{X}_x|\mathcal{Y}_x) \geq 1/p(|x|),$$

then there exists a polynomial-time-computable function $f = \{f_x\}_{x\in\{0,1\}^*}$ that is one-way against $\mathsf{P/poly}$ *almost everywhere on I.*

Lemma 4.9. *For any joint distributions* $(\mathcal{X}_0, \mathcal{Y}_0)$ *and* $(\mathcal{X}_1, \mathcal{Y}_1)$ *over a universe* U*, and for any (independent) Bernoulli trial* E*, define a joint distributions as*

$$(\mathcal{X}, \mathcal{Y}) = \begin{cases} (\mathcal{X}_0, \mathcal{Y}_0) & \text{if } E = 0 \\ (\mathcal{X}_1, \mathcal{Y}_1) & \text{if } E = 1. \end{cases}$$

If $\operatorname{supp}(\mathcal{Y}_0) \cap \operatorname{supp}(\mathcal{Y}_1) = \emptyset$*, then*

$$H(\mathcal{X}|\mathcal{Y}) = \Pr[E = 0] \cdot H(\mathcal{X}_0|\mathcal{Y}_0) + \Pr[E = 1] \cdot H(\mathcal{X}_1|\mathcal{Y}_1).$$

Proof. Since $\operatorname{supp}(\mathcal{Y}_0)$ and $\operatorname{supp}(\mathcal{Y}_1)$ are disjoint, $D_0 := \operatorname{supp}(\mathcal{X}_0, \mathcal{Y}_0)$ and $D_1 := \operatorname{supp}(\mathcal{X}_1, \mathcal{Y}_1)$ are also disjoint. Thus,

$$\begin{aligned} H(\mathcal{X}|\mathcal{Y}) = &- \sum_{(x,y) \in D_0} \Pr[(\mathcal{X}, \mathcal{Y}) = (x, y)] \log \frac{\Pr[(\mathcal{X}, \mathcal{Y}) = (x, y)]}{\Pr[\mathcal{Y} = y]} \\ &- \sum_{(x,y) \in D_1} \Pr[(\mathcal{X}, \mathcal{Y}) = (x, y)] \log \frac{\Pr[(\mathcal{X}, \mathcal{Y}) = (x, y)]}{\Pr[\mathcal{Y} = y]} \\ = &- \Pr[E = 0] \sum_{(x,y) \in D_0} \Pr[(\mathcal{X}_0, \mathcal{Y}_0) = (x, y)] \log \frac{\Pr[E = 0] \Pr[(\mathcal{X}_0, \mathcal{Y}_0) = (x, y)]}{\Pr[E = 0] \Pr[\mathcal{Y}_0 = y]} \\ &- \Pr[E = 1] \sum_{(x,y) \in D_1} \Pr[(\mathcal{X}_1, \mathcal{Y}_1) = (x, y)] \log \frac{\Pr[E = 1] \Pr[(\mathcal{X}_1, \mathcal{Y}_1) = (x, y)]}{\Pr[E = 1] \Pr[\mathcal{Y}_1 = y]} \\ = &\Pr[E = 0] \cdot H(\mathcal{X}_0|\mathcal{Y}_0) + \Pr[E = 1] \cdot H(\mathcal{X}_1|\mathcal{Y}_1). \end{aligned}$$

□

5 On Transforming Worst-Case Hardness into Cryptography

In this section, we introduce two key properties of promise problems for transforming the worst-case hardness into cryptography and prove Theorems 2.1 and 2.3.

5.1 Black-Box Reduction/One-Way Function Property

First, we introduce the BBR/OWF-property of a promise problem that enables a non-black-box security reduction from errorless average-case hardness to one-wayness of a function.

Definition 5.1 (BBR/OWF-property)**.** *A promise problem* Π *is said to have a* BBR/OWF *property if there exist two polynomial-time-computable auxiliary-input functions* $f = \{f_x\}_{x \in \{0,1\}^*}$*,* $g = \{g_x\}_{x \in \{0,1\}^*}$*, a subset* $I \subseteq \Pi_{\mathrm{yes}} \cup \Pi_{\mathrm{no}}$*, and a polynomial-time oracle machine* R *satisfying that*

1. *(OWF part)* $\{f_x\}_{x \in I}$ *is one-way (secure against* $\mathsf{P/poly}$*) almost everywhere on* I*;*
2. *(BBR part) There exists a polynomial* p *such that for every* $x \in (\Pi_{\mathrm{yes}} \cup \Pi_{\mathrm{no}}) \setminus I$ *and every oracle* A*, if* $\Pr_w[A(g_x(w)) \notin g_x^{-1}(g_x(w))] \leq 1/p(|x|)$*, then*

$$\Pr_R[R^A(x) = \Pi(x)] \geq 2/3.$$

Lemma 5.2. *Let Π be a promise problem that has the BBR/OWF property. If there exist a samplable distribution family $\mathcal{D} = \{\mathcal{D}_n\}$ on $\Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}$ and a polynomial γ such that $(\Pi, \mathcal{D}) \notin$ $\mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$, then there exists a one-way function secure against $\mathsf{P/poly}$.*

Proof. Suppose that Π is a promise problem that has the BBR/OWF property and $(\Pi, \mathcal{D}) \notin$ $\mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ for a samplable distribution family $\mathcal{D} = \{\mathcal{D}_n\}$ on $\Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}$ and a polynomial γ. Let D be the polynomial-time sampler for $\mathcal{D}$ (i.e., $D(1^n; w) \equiv \mathcal{D}_n$, where w is a uniformly random seed).

Since Π has the BBR/OWF property, we have polynomial-time-computable functions $f = \{f_x\}_{x\in\{0,1\}^*}$ and $g = \{g_x\}_{x\in\{0,1\}^*}$, a subset $I \subseteq \Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}$, a polynomial-time oracle machine R, and a polynomial p that satisfy the conditions of Definition 5.1.

We construct a polynomial-time computable function $f' = \{f'_n\}_{n\in\mathbb{N}}$ as

$$f'_n(w, r, r') = (x, f_x(r), g_x(r'))$$

where $x = D(1^n, w)$, r is a random seed for f, and r' is a random seed for g.

We show that f' is (weak) one-way under the assumption that $(\Pi, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ by contraposition. For this, we assume that there exists a polynomial-time adversary A that inverts f' with failure probability at most $1/(4\gamma(n)p(n))$ for infinitely many $n \in \mathbb{N}$. Let $\mathcal{N}$ be the infinite set of such n.

We construct an errorless heuristic algorithm A' that solves $(\Pi, \mathcal{D})$ with failure probability at most $1/\gamma(n)$. For a given x drawn from $\mathcal{D}_n$, the algorithm A' first examines whether A successfully inverts f'_x with failure probability at most $1/(2p(n))$ by a randomized test T specified later. If x passes the test, A' executes $R^?(x)$ and outputs the same answer, where the oracle is simulated by $\mathcal{O}(y)$ that outputs the third element of $A(x, f_x(r), y)$ (i.e., the inverse for g_x) for uniformly random r; otherwise, it outputs $\perp$.

The randomized test T is based on the standard empirical estimation. For a given x, the test T selects $N := 8p(n)^2 n \ln 2$ independent random seeds $r_1, r'_1, \ldots, r_N, r'_N$ for f_x and g_x, executes $A(x, f_x(r_i), g_x(r_i{}'))$ for each i, and counts the number m of $i \in [N]$ for which A fails to invert f'. If $m \le 3N/(4p(n))$, then T accepts x; otherwise, T rejects x.

We show the following claims.

Claim 5.3. *For all $n \in \mathbb{N}$ and all $x \in \mathrm{supp}(\mathcal{D}_n)$, if*

$$\Pr_{A,r}\left[A(f'_n(r)) \notin {f'_n}^{-1}(f'_n(r)) \;\middle|\; \text{the first element of } f'_n(r) \text{ is } x\right] > 1/p(n),$$

then $\Pr_T[x \text{ passes the test } T] \le 2^{-n}$.

Claim 5.4. *For all $n \in \mathcal{N}$,*

$$\Pr_{x\sim\mathcal{D}_n}\left[\Pr_T[x \text{ passes the test } T] \ge 1 - 2^n\right] \ge 1 - 1/\gamma(n).$$

First, we assume the claims above and complete the proof. For any $n \in \mathbb{N}$ and $x \in \mathrm{supp}(\mathcal{D}_n)$ that passes the test T with probability at least $1 - 2^{-n}$, by Claim 5.3, A inverts f' given the first element is x with failure probability at most $1/p(n)$, and $\mathcal{O}$ inverts g_x with failure probability at most $1/p(n)$. In addition, we observe that such x is not contained in I because such A also inverts f_x with probability at least $1 - 1/p(n)$, which contradicts the property of I. Thus, by the property of R, the algorithm A outputs $\Pi(x)$ with probability at least $3/4$. By the union bound, A' outputs $\Pi(x)$ with probability at least $1 - (1/4 + 2^{-n}) \geq 2/3$ in the case. By Claim 5.4, such x is selected according to $\mathcal{D}_n$ with probability at least $1 - 1/\gamma(n)$ for any $n \in \mathcal{N}$. By contrast, any $x \in \mathrm{supp}(\mathcal{D}_n)$ for which the failure probability of $\mathcal{O}$ is larger than $1/p(n)$ (such x may cause the error of R) passes the test T only with negligible probability by Claim 5.3. Therefore, A outputs $\perp$ with probability at least $1 - 2^{-n}$ for such x. Recall that the success probability of A (over the randomness for A) can be enhanced to $1 - 2^{-\mathsf{poly}(n)}$ by the standard repetition technique, and the randomness can be fixed as a part of the nonuniform advice based on Adleman's technique [Adl78]. Therefore, we have $(\Pi, \mathcal{D}) \in \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$.

It remains to prove the claims above. For convenience, we introduce the notation γ_x as

$$\gamma_x := \Pr_{A,r}\left[A(f'_n(r)) \notin {f'_n}^{-1}(f'_n(r)) \;\middle|\; \text{the first element of } f'_n(r) \text{ is } x\right].$$

Proof of Claim 5.3. Suppose that x satisfies

$$\gamma_x = \Pr_{A,r}\left[A(f'_n(r)) \notin {f'_n}^{-1}(f'_n(r)) \;\middle|\; \text{the first element of } f'_n(r) \text{ is } x\right] > 1/p(n).$$

It is not hard to see that the test T examines the probability above by the empirical estimation. By Hoeffding's inequality, the probability that $m \notin [\gamma_x N \pm N/4p(n)]$ is bounded above by $2\exp(-2(1/4p(n))^2 N) \leq 2^{-n}$. Thus, with probability at least $1 - 2^n$, we have $m \geq \gamma_x N - N/4p(n) > 3N/(4p(n))$, and T rejects x in this case. ◇

Proof of Claim 5.4. Recall that for every $n \in \mathcal{N}$,

$$\mathop{\mathbb{E}}_{x \sim \mathcal{D}_n}[\gamma_x] = \Pr_{A,r}[A(f'_n(r)) \notin {f'_n}^{-1}(f'_n(r))] \leq \frac{1}{2\gamma(n)p(n)}$$

By Markov's inequality, we have

$$\Pr_{x \sim \mathcal{D}_n}[\gamma_x \leq 1/2p(n)] \geq 1 - 1/\gamma(n).$$

Fix x satisfying $\gamma_x \leq 1/2p(n)$ arbitrarily. By Hoeffding's inequality, the probability that $m \notin [\gamma_x N \pm N/4p(n)]$ is bounded above by $2\exp(-2(1/4p(n))^2 N) \leq 2^{-n}$. Thus, with probability at least $1 - 2^{-n}$, we have $m \leq \gamma_x N + N/4p(n) \leq 3N/(4p(n))$, and T accepts x in the case. Since $\Pr_{x \sim \mathcal{D}}[\gamma_x \leq 1/2p(n)] \geq 1 - 1/\gamma(n)$, we obtain the claim and complete the proof. ◇

□

We also extend the BBR/OWF property to the errorless average-case setting.

Definition 5.5 (Average-case BBR/OWF-property)**.** *A distributional problem $(\Pi, \mathcal{D})$ is said to have an* average-case BBR/OWF property *if there exists a polynomial-time algorithm M such that*

1. *for every $n, \gamma \in \mathbb{N}$, $\Pr_{x \sim \mathcal{D}_n}[M(x, 1^n, 1^\gamma) = \perp] \leq 1/\gamma$;*
2. *for every polynomial γ, the promise problem Π^γ has the BBR/OWF property, where $\Pi^\gamma = (\Pi^\gamma_{\mathsf{yes}}, \Pi^\gamma_{\mathsf{no}})$ is defined as follows: for every $n \in \mathbb{N}$,*

$$(x, 1^n, 1^{\gamma(n)}) \in \Pi^\gamma_{\mathsf{yes}} \iff x \in \Pi_{\mathsf{yes}} \cap \left\{x \in \mathrm{supp}(\mathcal{D}_n) : M(x, 1^n, 1^{\gamma(n)}) \neq \perp\right\};$$

$$(x, 1^n, 1^{\gamma(n)}) \in \Pi^\gamma_{\mathsf{no}} \iff x \in \Pi_{\mathsf{no}} \cap \left\{x \in \mathrm{supp}(\mathcal{D}_n) : M(x, 1^n, 1^{\gamma(n)}) \neq \perp\right\}.$$

Intuitively, if a distributional problem $(\Pi, \mathcal{D})$ satisfies the average-case BBR/OWF property, then for every polynomial γ, there exists a subset S of instances that has at least $1 - 1/\gamma(n)$ weight with respect to $\mathcal{D}_n$ for each n and satisfies that (i) the membership of S is efficiently determined (by the algorithm M) and (ii) Π restricted onto S (i.e., Π^γ) has the (worst-case) BBR/OWF property.

We obtain the analogue of Lemma 5.2 in the errorless average-case setting.

Lemma 5.6. *Let $\mathcal{D} = \{\mathcal{D}_n\}_{n\in\mathbb{N}}$ be a samplable distribution, and let $(\Pi, \mathcal{D})$ be a distributional problem that has the average-case BBR/OWF property. If there exists a polynomial γ such that $(\Pi, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$, then there exists a one-way function secure against $\mathsf{P/poly}$.*

Proof. Let M be the polynomial-time algorithm in the average-case BBR/OWF property for $(\Pi, \mathcal{D})$. Suppose that there exists a polynomial γ such that $(\Pi, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$. For each $n \in \mathbb{N}$, let $G_n = \left\{x \in \mathrm{supp}(\mathcal{D}_n) : M(x, 1^n, 1^{2\gamma(n)}) \neq \bot\right\}$.

For each $n \in \mathbb{N}$, let $\mathcal{D}'_n$ be the conditional distribution of $(x, 1^n, 1^{2\gamma(n)})$ for $x \sim \mathcal{D}$ given $x \in G_n$. Then, $\mathcal{D}' = \{\mathcal{D}'_n\}_{n\in\mathbb{N}}$ is samplable by using the sampler for $\mathcal{D}$ and M with negligible statistical error. Let $\mathcal{D}'' = \{\mathcal{D}''_n\}_n$ be the modified samplable distribution family obtained from $\mathcal{D}'$. Note that $\mathrm{supp}(\mathcal{D}'') = \Pi^{2\gamma}_{\mathsf{yes}} \cup \Pi^{2\gamma}_{\mathsf{no}}$, where $\Pi^{2\gamma}$ is the promise problem defined in Definition 5.5 for 2γ.

Since $\Pi^{2\gamma}$ has the BBR/OWF property, in the same way as the proof of Lemma 5.2, it is shown that there exists a polynomial-time-computable function $f = \{f_n \colon \{0,1\}^n \to \{0,1\}^{\mathsf{poly}(n)}\}_{n\in\mathbb{N}}$ and a polynomial-time oracle machine R such that for any nonuniform polynomial-time-computable adversary A that breaks f_n for infinitely many n, the algorithm R^A solves Π on errorless average under $\mathcal{D}''_n$ with failure probability at most $1/4\gamma(n)$. Now we consider the algorithm B that is given $x \sim \mathcal{D}_n$ and then executes $M(x, 1^n, 1^{2\gamma(n)})$. If M returns $\bot$, then B also outputs $\bot$; otherwise, B executes $R^A(x, 1^n, 1^{2\gamma(n)})$ and answers the same answer. Since R^A is errorless, B is also errorless. In addition, over the choice of $x \sim \mathcal{D}_n$, the probability that $B(x)$ outputs $\bot$ is at most

$$\begin{aligned}
&\Pr_{x\sim\mathcal{D}_n}[x \notin G_n] + \Pr_{x\sim\mathcal{D}_n}\left[R^A(x, 1^n, 1^{2\gamma(n)}) = \bot \;\middle|\; x \in G_n\right] \\
&\quad\le \frac{1}{2\gamma(n)} + \Delta_{\mathsf{TV}}(\mathcal{D}'_n, \mathcal{D}''_n) + \Pr_{(x,1^n,1^{2\gamma(n)})\sim\mathcal{D}''_n}\left[R^A(x) = \bot\right] \\
&\quad\le \frac{3}{4\gamma(n)} + \mathsf{negl}(n).
\end{aligned}$$

This contradicts $(\Pi, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ when n is sufficiently large. Thus, f is one-way under the assumption that $(\Pi, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$. □

5.2 AIOWF-Hardness

We define the AIOWF-hardness of a distributional problem $(\Pi, \mathcal{D})$ as the existence of a reduction from inverting auxiliary-input one-way functions to solving Π on average in errorless setting.

Definition 5.7 (AIOWF-hard)**.** *A distributional problem $(\Pi, \mathcal{D})$ is said to be* AIOWF-hard *if for every polynomial-time-computable auxiliary-input function $f = \{f_z : \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z\in\{0,1\}^*}$ and every polynomial p, there exist a polynomial γ and a polynomial-time oracle machine $R_f^?$ such that for every oracle A and every $n \in \mathbb{N}$, if A solves $(\Pi, \mathcal{D}_n)$ on errorless average with failure probability at most $1/\gamma(n)$, then R_f^A inverts f_z for all $z \in \{0,1\}^n$ with failure probability at most $1/p(n)$, i.e.,*

$$\Pr_{w,R_f}\left[R_f^A(z, f_z(w)) \in f_z^{-1}(f_z(w))\right] \ge 1 - 1/p(n).$$

Theorem 5.8. *If a promise problem* Π *has the BBR/OWF property, and a distributional problem* $(\Gamma, \mathcal{D})$ *is AIOWF-hard, then* $\Pi \notin \mathsf{i.o.P/poly}$ *implies* $(\Gamma, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ *for every polynomial* γ.

Proof. Suppose Π is a promise problem that has the BBR/OWF property, and $(\Gamma, \mathcal{D})$ is an AIOWF-hard distributional problem.

Since Π has the BBR/OWF property, we have a polynomial-time-computable functions $f = \{f_x\}_{x \in \{0,1\}^*}$ and $g = \{g_x\}_{x \in \{0,1\}^*}$, a subset $I \subseteq \Pi_{\mathsf{yes}} \cup \Pi_{\mathsf{no}}$, a polynomial-time oracle machine R, and a polynomial p that satisfy the conditions of Definition 5.1.

Next, we use the AIOWF-hardness of $(\Gamma, \mathcal{D})$ for the auxiliary-input function

$$h = \{h_x \colon \{0,1\}^{\mathsf{poly}(|x|)} \to \{0,1\}^{\mathsf{poly}(|x|)}\}_{x \in \{0,1\}^*}$$

defined as $h_x(r, r') = (f_x(r), g_x(r'))$ and for the polynomial p. Then, there exist a polynomial γ and a polynomial-time oracle machine $R_h^?$ such that for every oracle A and every $n \in \mathbb{N}$, if A solves $(\Gamma, \mathcal{D}_n)$ on errorless average with failure probability at most $1/\gamma(n)$, then R_h^A inverts h_x for all $x \in \{0,1\}^n$ with failure probability at most $1/p(n)$.

Now we show that $(\Gamma, \mathcal{D}) \in \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ implies $\Pi \in \mathsf{i.o.P/poly}$ (i.e., a worst-case-to-average-case reduction). Suppose that there exists a nonuniform polynomial-time algorithm A that solves $(\Gamma, \mathcal{D})$ with failure probability at most $1/\gamma(n)$ for infinitely many parameters n. Then, by the property of $R_h^?$, the algorithm $B := R_h^A$ inverts h_x for the same parameters $n \in \mathbb{N}$ and for all $x \in \{0,1\}^n$ with failure probability at most $1/p(n)$. We define B_f (resp. B_g) so that $B_f(x, y)$ outputs the first element of $B(x, y, g_x(r'))$ for a random seed r' (resp. $B_g(x, y)$ outputs the second element of $B(x, f_x(r), y)$ for a random seed r). Then, $B_f(x, \text{-})$ (resp. $B_g(x, \text{-})$) inverts f_x (resp. g_x) with failure probability at most $1/p(n)$. Since A is a nonuniform polynomial-time algorithm, B, B_f, and B_g are implemented as nonuniform polynomial-time algorithms. Thus, by the definition of I, such x is not contained in I. Therefore, by the property of R, the algorithm $R^{B_g(x, \text{-})}(x)$ outputs $\Pi(x)$ with probability at least $2/3$ for the same parameters $n \in \mathbb{N}$ and for all $x \in \{0,1\}^n$. Since R^{B_g} is implemented as a nonuniform randomized polynomial-time algorithm, it is also implemented as a circuit family based on Adleman's technique [Adl78]. Therefore, we have $\Pi \in \mathsf{i.o.P/poly}$. □

5.3 Problems Having Two Properties

In this section, we study when the BBR/OWF property and AIOWF-hardness are satisfied.

5.3.1 AIOWF-Hardness

Lemma 5.9 ([cf. HS17])**.** *For every* $\epsilon \in (0, 1/2)$, $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U})$ *is AIOWF-hard.*

Proof (sketch). Let $f = \{f_z\}_{z \in \{0,1\}^*}$ be an arbitrary polynomial-time-computable auxiliary-input function. For every polynomial p and every constant $c \in \mathbb{N}$, based on the GGM construction [GGM86], there exists a constant d such that for each $\tau \in \mathbb{N}$, we can construct a polynomial-time-computable auxiliary-input function $g = \{g_z \colon \{0,1\}^{|z|} \times \{0,1\}^{c \log |z|} \to \{0,1\}^{|z|}\}_{z \in \{0,1\}^*}$ that has a polynomial-time reduction $R^?$ such that for every $z \in \{0,1\}^*$ and for every oracle machine $A^?$ that distinguishes two cases (i) A is given access to $\mathcal{O}(\cdot) := g_z(w, \cdot)$ for $w \sim \{0,1\}^{|z|}$ and (ii) A is given access to $\mathcal{O}(\cdot) = h(\cdot)$ for $h \sim \mathcal{F}_{c \log |z|, |z|} := \{h' \colon \{0,1\}^{c \log |z|} \to \{0,1\}^{|z|}\}$ with non-negligible advantage, R^A inverts f_z with failure probability at most $1/p(|z|)$. Moreover, for every $z \in \{0,1\}^*$

and every $w \in \{0,1\}^{|z|}$, each bit of the function $g_{z,w} := g_z(w, \cdot)_1$ is computable by a circuit of size $(c \log |z|) \cdot d|z|^d$.

We select a sufficiently large constant c so that $d(\log N)N^{d/c} \leq N^\epsilon$ for every $N \geq 2$. Then, for any z with $|z| \geq 2$, each $g_{z,w}$ is computable by a circuit of size at most

$$(c \log |z|) \cdot d|z|^d \leq d \cdot (\log |z|^c)(|z|^c)^{d/c} \leq (|z|^c)^\epsilon$$

Let $n := c \log |z|$ be the input length of $g_{z,w}$. Then $\mathsf{cc}(g_{z,w}) \leq 2^{\epsilon n}$ for every $z \in \{0,1\}^*$ with $|z| \geq 2$ and every $w \in \{0,1\}^{|z|}$.

By contrast, for $\mathcal{F}_{n,1} := \{h : \{0,1\}^n \to \{0,1\}\}$,

$$\Pr_{h \sim \mathcal{F}_{n,1}}[\mathsf{cc}(h) < 2^{(1-\epsilon)n}] \leq \frac{2^{\mathsf{poly}(n) 2^{(1-\epsilon)n}}}{2^{2^n}} \leq \mathsf{negl}(2^n) = \mathsf{negl}(|z|).$$

Namely, for any $z \in \{0,1\}^*$ with $|z| \geq 2$ and any $w \in \{0,1\}^{|z|}$, it holds that $\mathsf{tt}(g_{z,w})$ is a yes instance for $\mathsf{Gap}_\epsilon\mathsf{MCSP}$, and $\mathsf{tt}(h)$ is a no instance for $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ with probability at least $1 - \mathsf{negl}(|z|)$. Therefore, if there exists a nonuniform polynomial-time algorithm A that solves $\mathsf{Gap}_\epsilon\mathsf{MCSP}$ on errorless average under $\mathcal{U}_{|z|^c}$ with failure probability at most $1/\mathsf{poly}(|z|)$, we can distinguish every $g_{z,w}$ from a truly random function h by interpreting the case where A outputs $\perp$ or 1 (resp. 0) as the case of pseudorandom functions (resp. random functions). Thus, based on R, we obtain the reduction from inverting $f = \{f_z\}_z$ to $(\mathsf{Gap}_\epsilon\mathsf{MCSP}, \mathcal{U})$. □

Lemma 5.10. *For any* NP*-hard problem* Π*, there exists a samplable distribution* $\mathcal{D}$ *such that* $(\Pi, \mathcal{D})$ *is AIOWF-hard.*

Proof. Since Π is NP-hard, there exists a polynomial-time algorithm M such that for every $x \in \{0,1\}^*$, (i) if $x \in \mathsf{GapMCSP}_{\mathrm{yes}}$, then $M(x) \in \Pi_{\mathrm{yes}}$; (ii) if $x \in \mathsf{GapMCSP}_{\mathrm{no}}$, then $M(x) \in \Pi_{\mathrm{no}}$. Then, $(\Pi, M(\mathcal{U}))$ is AIOWF-hard because for every $\gamma \colon \mathbb{N} \to \mathbb{N}$, $(\Pi, M(\mathcal{U})) \in \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ implies $(\mathsf{GapMCSP}, \mathcal{U}) \in \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$. □

5.3.2 BBR/OWF Property

Theorem 5.11 ([OV07])**.** *Any promise problem* $\Pi \in \mathsf{CZKA}$ *has the BBR/OWF property.*

Proof. Ong and Vadhan [OV07] proved that every promise problem $\Pi \in \mathsf{CZKA}$ admits the SZK/OWF characterization, which is stated as follows: there exists a subset $I \subseteq \Pi_{\mathrm{yes}} \cup \Pi_{\mathrm{no}}$ and a polynomial-time-computable function $f = \{f_z\}_{z \in \{0,1\}^*}$ such that (i) f is one-way against $\mathsf{P/poly}$ almost everywhere on I, and (ii) $(\Pi_{\mathrm{yes}} \setminus I, \Pi_{\mathrm{no}} \setminus I) \in \mathsf{SZK}$. Furthermore, Ostrovsky [Ost91] showed that every promise problem SZK admits a black-box reduction from solving Π for a given instance x to inverting a function f_x indexed by x (which is constructed from the simulator of the zero-knowledge proof system). By applying [Ost91] for the second (i.e., SZK) case of the SZK/OWF property, we also obtain the BBR/OWF property for Π. □

Moreover, we show that a potentially larger class on computational knowledge complexity than zero-knowledge yields the BBR/OWF property.

Theorem 5.12. *If a promise problem* Π *has an interactive proof system* (P, V) *that has computational knowledge complexity at most* $k(n) = O(\log n)$ *and negligible soundness error, then* Π *has the BBR/OWF property.*

We prove Theorem 5.12 in Section 5.7.

5.4 Errorless Average-Case Zero-Knowledge

We extend the argument to the case of average-case zero-knowledge. First, we formally introduce the notion of errorless average-case interactive proof/argument systems.

Definition 5.13 (Average-case interactive proof/argument)**.** *An average-case interactive proof system (P, V) for a distributional problem $(\Pi, \mathcal{D})$ is an interactive proof system satisfying the following:*

- (*Completeness*) *For every $n, \delta^{-1} \in \mathbb{N}$ and every $x \in \Pi_{\mathrm{yes}} \cap \mathrm{supp}(\mathcal{D}_n)$,*
$$\Pr_{P,V}[\langle P, V(1^{\delta^{-1}})\rangle(1^n, x) \in \{1, \bot\}] \geq 1 - \mathsf{negl}(n).$$
- (*Soundness*) *For every $n, \delta^{-1} \in \mathbb{N}$, every $x \in \Pi_{\mathrm{no}} \cap \mathrm{supp}(\mathcal{D}_n)$, and every prover P^*,*
$$\Pr_{P,V}[\langle P^*, V(1^{\delta^{-1}})\rangle(1^n, x) \in \{0, \bot\}] \geq 1 - \mathsf{negl}(n).$$
- (*Average-case requirement*) *For every $\delta^{-1} \in \mathbb{N}$, every prover P^*, and for every sufficiently large $n \in \mathbb{N}$,*
$$\Pr_{x \sim \mathcal{D}_n}\left[\Pr_{P^*,V}[\langle P^*, V(1^{\delta^{-1}})\rangle(x) = \bot] \geq 2/3\right] \leq \delta.$$

An average-case interactive argument system is defined in the same manner as above except that P^* is restricted to be a nonuniform polynomial-time prover.

We may omit 1^n from the input for readability.

Next, we extend the notion of zero-knowledge to the errorless average-case setting.

Definition 5.14 (Average-case zero-knowledge proof/argument)**.** *An average-case interactive proof system (P, V) for a distributional problem $(\Pi, \mathcal{D})$ is said to be statistically (resp. computationally) zero-knowledge if it is statistically (resp. computationally) zero-knowledge on $(\Pi_{\mathrm{yes}} \cup \Pi_{\mathrm{no}}) \cap \mathrm{supp}(\mathcal{D})$ with the following exception: For every $\delta^{-1} \in \mathbb{N}$, an honest-verifier simulator S outputs $\bot$ (with a negligible error) if and only if*
$$\Pr_{P,V}[\langle P, V(1^{\delta^{-1}})\rangle(x) = \bot] \geq 2/3.$$

Let AvgCZK *denote the class of distributional problems that have computational zero-knowledge average-case interactive proof systems.*

We also define zero-knowledge statistical (resp. computational) average-case interactive argument systems in the same manner and the class $\mathsf{AvgSZKA}$ *(resp.* $\mathsf{AvgCZKA}$*) as the class of distributional problems that have zero-knowledge statistical (resp. computational) average-case interactive argument systems.*

We show that every distributional problem $(\Pi, \mathcal{D})$ in $\mathsf{AvgCZKA}$ (thus, every $(\Pi, \mathcal{D}) \in \mathsf{AvgSZKA}$ and every $(\Pi, \mathcal{D}) \in \mathsf{AvgCZK}$) has the average-case BBR/OWF property.

Theorem 5.15. *Any promise problem $(\Pi, \mathcal{D}) \in \mathsf{AvgCZKA}$ has the average-case BBR/OWF property.*

Proof sketch. The theorem follows from Theorem 5.11. Let (P, V) be the average-case computational zero-knowledge interactive argument system for $(\Pi, \mathcal{D})$, and let S be its honest-verifier simulator.

We define the polynomial-time algorithm M of the average-case BBR/OWF property as follows: On input $(x, 1^n, 1^\gamma)$, where $n, \gamma \in \mathbb{N}$ and $x \in \mathrm{supp}(\mathcal{D}_n)$, the algorithm M executes $S(x, 1^\gamma)$ and if S outputs $\bot$, then M outputs $\bot$; otherwise, M outputs 1. By the average-case requirement of (P, V), the algorithm $M(x, 1^n, 1^\gamma)$ outputs $\bot$ with probability at most $1/\gamma$ over $x \sim \mathcal{D}_n$.

It is easily observed that $\Pi^\gamma \in \mathsf{CZKA}$ (where Π^γ is the promise problem defined in Definition 5.5) for every polynomial $\gamma \colon \mathbb{N} \to \mathbb{N}$, where for a given input $(x, 1^n, 1^{\gamma(n)})$, the interactive argument system executes $\langle P, V(1^{\gamma(n)}) \rangle (1^n, x)$. Thus, by Theorem 5.11, Π^γ has the BBR/OWF property for every polynomial γ. □

5.5 Proofs of Main Theorems

Now, we show the main theorems.

Proof of Theorem 2.1. Fix $\mathfrak{C} \in \{\mathsf{SZKA}, \mathsf{CZK}, \mathsf{CZKA}\}$ arbitrarily. Note that $\mathfrak{C} \subseteq \mathsf{CZKA}$ in any case.

Item 1 $\Longrightarrow$ Item 2 follows from (i) $\mathsf{NP} \subseteq \mathfrak{C}$ under the existence of one-way functions [GMW91; NOV06] and (ii) $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$ under the existence of one-way functions (secure against $\mathsf{P/poly}$).

Item 1 $\Longrightarrow$ Item 3 also follows from (i) $\mathsf{GapMCSP} \in \mathfrak{C}$ under the existence of one-way functions [GMW91; NOV06] and (ii) $\mathsf{GapMCSP} \notin \mathsf{i.o.P/poly}$ under the existence of one-way functions (secure against $\mathsf{P/poly}$). Item 3 $\Longrightarrow$ Item 4 is trivial.

Item 2 $\Longrightarrow$ Item 1. Suppose that $\mathsf{NP} \subseteq \mathfrak{C}\,(\subseteq \mathsf{CZKA})$ and $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$. Then, $\mathsf{SAT} \in \mathsf{CZKA}$, and SAT has the BBR/OWF property by Theorem 5.11. In addition, by Lemma 5.10, there exists a samplable distribution $\mathcal{D}$ such that $(\mathsf{SAT}, \mathcal{D})$ is AIOWF-hard. Thus, the assumption that $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$ implies that $(\mathsf{SAT}, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ for every polynomial γ by Theorem 5.8, and this implies the existence of a one-way function secure against $\mathsf{P/poly}$ by Lemma 5.2.

Item 4 $\Longrightarrow$ Item 1. Suppose that $(\mathsf{GapMCSP}, \mathcal{U}) \in \mathsf{Avg}\mathfrak{C}\,(\subseteq \mathsf{AvgCZKA})$ and $\mathfrak{C} \not\subseteq \mathsf{i.o.P/poly}$. Let $\Pi \in \mathfrak{C} \setminus \mathsf{i.o.P/poly}$. Since $\Pi \in \mathfrak{C} \subseteq \mathsf{CZKA}$, the promise problem Π has the BBR/OWF property by Theorem 5.11. By Lemma 5.9, $(\mathsf{GapMCSP}, \mathcal{U})$ is AIOWF-hard; thus by Theorem 5.8, $\Pi \not\subseteq \mathsf{i.o.P/poly}$ implies that $(\mathsf{GapMCSP}, \mathcal{U}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ for every polynomial γ. By Lemma 5.6 and Theorem 5.15, this implies the existence of a one-way function secure against $\mathsf{P/poly}$. □

Proof of Theorem 2.3. Item 1 $\Longrightarrow$ Items 2 and 3 follows from Item 1 $\Longrightarrow$ Items 2 and 3 of Theorem 2.1 and the fact that soundness error of zero-knowledge proof can be reduced by the parallel executions.

Item 2 $\Longrightarrow$ Item 1. If NP has an interactive proof system (P, V) that has computational knowledge complexity at most $O(\log n)$, then by Theorem 5.12, SAT has the BBR/OWF property. By Lemma 5.10, there exists a samplable distribution $\mathcal{D}$ such that $(\mathsf{SAT}, \mathcal{D})$ is AIOWF-hard. Thus, the assumption that $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$ implies that $(\mathsf{SAT}, \mathcal{D}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ for every polynomial γ by Theorem 5.8, and this implies the existence of a one-way function secure against $\mathsf{P/poly}$ by Lemma 5.2.

Item 3 $\Longrightarrow$ Item 1. If $\mathsf{GapMCSP}$ has an interactive proof system (P, V) that has computational knowledge complexity at most $O(\log n)$, then by Theorem 5.12, $\mathsf{GapMCSP}$ has the BBR/OWF property. Since $\mathsf{CZK} \not\subseteq \mathsf{i.o.P/poly}$, there exists $\Pi \in \mathsf{CZK} \setminus \mathsf{i.o.P/poly}$. Since $\Pi \in \mathsf{CZK}$, the promise problem Π has the BBR/OWF property by Theorem 5.11. By Lemma 5.9, $(\mathsf{GapMCSP}, \mathcal{U})$ is

AIOWF-hard; thus by Theorem 5.8, $\Pi \not\subseteq \mathsf{i.o.P/poly}$ implies that $(\mathsf{GapMCSP}, \mathcal{U}) \notin \mathsf{i.o.Avg}_{1/\gamma}\mathsf{P/poly}$ for every polynomial γ. By Lemma 5.2, this implies the existence of a one-way function secure against $\mathsf{P/poly}$. $\square$

5.6 Towards Basing Public-Key Cryptography on Worst-Case Hardness

As a corollary to our main result, we improve the consequence following from *laconic* zero-knowledge argument systems, presented in [BDRV18].

First, we review some terminologies.

Definition 5.16 (Laconic prover). *Let (P, V) be an interactive proof (or argument) system. For $q \colon \mathbb{N} \to \mathbb{N}$, the prover P is said to be q-laconic if P sends at most $q(\cdot)$-bit message (as a function in the length of input and auxiliary advice) for each round.*

Definition 5.17 (Cryptographic hardness). *Let $\Pi \in \mathsf{NP}$ be a promise problem with an NP-relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$. Let $\mathcal{D}^Y = \{\mathcal{D}_n^Y\}_n$ and $\mathcal{D}^N = \{\mathcal{D}_n^N\}_n$ be samplable distributions. We say that $(\Pi, \mathcal{D}^Y, \mathcal{D}^N)$ is cryptographically hard if for every $n \in \mathbb{N}$,*

- $\Pr_{(x,w)\sim\mathcal{D}_n^Y}[(x, w) \in R] \leq \mathsf{negl}(n)$*;*
- $\Pr_{x\sim\mathcal{D}_n^N}[x \in \Pi_{\mathrm{no}}] \leq \mathsf{negl}(n)$*;*
- *$\mathcal{D}^N$ and the distribution family of the first half element of $\mathcal{D}^Y$ are computationally indistinguishable.*

Berman, Degwekar, Rothblum, and Vasudevan [BDRV18] proved the following theorem.

Theorem 5.18 ([BDRV18, Theorem 3.6]). *For every cryptographic hard $(\Pi, \mathcal{D}^Y, \mathcal{D}^N)$, if Π has a statistical zero-knowledge argument system*[6] *with an efficient and q-laconic prover with $\ell(n)^2 \cdot q(n)^3 = O(\log n)$ (where $2 \cdot \ell(\cdot)$ is the round complexity), then there exists a semantic secure public-key encryption scheme.*

Now, we restate Corollary 2.9 more formally and prove it.

Corollary 5.19.

- *If NP has a statistical zero-knowledge argument system with an efficient and q-laconic prover with $\ell(n)^2 \cdot q(n)^3 = O(\log n)$ (where $2 \cdot \ell(\cdot)$ is the round complexity), then there exists a public-key encryption scheme whose semantic security is based on $\mathsf{NP} \not\subseteq \mathsf{i.o.P/poly}$.*
- *If $\mathsf{GapMCSP}$ has a statistical zero-knowledge argument system with an efficient and q-laconic prover with $\ell(n)^2 \cdot q(n)^3 = O(\log n)$ (where $2 \cdot \ell(\cdot)$ is the round complexity), then there exists a public-key encryption scheme whose semantic security is based on $\mathsf{GapMCSP} \notin \mathsf{i.o.P/poly}$.*

Proof. We only show the case of $\mathsf{GapMCSP}$. Note that the case of NP is shown by replacing $\mathsf{GapMCSP}$ with SAT in the proof.

We assume that $\mathsf{GapMCSP}$ has a statistical zero-knowledge argument system with an efficient and q-laconic prover with $\ell(n)^2 \cdot q(n)^3 = O(\log n)$ and $\mathsf{GapMCSP} \notin \mathsf{i.o.P/poly}$ and then derive the existence of a semantic secure public-key encryption scheme.

Since $\mathsf{GapMCSP} \in \mathsf{SZKA}$ and $\mathsf{GapMCSP} \notin \mathsf{i.o.P/poly}$, we have $\mathsf{SZKA} \not\subseteq \mathsf{i.o.P/poly}$. By Theorem 2.1, there exists a one-way function secure against $\mathsf{P/poly}$.

[6] The theorem in [BDRV18] and Theorem 2.1 actually hold for *honest-verifier* zero-knowledge.

We apply the same reduction from inverting function to distinguishing truth-tables of pseudorandom functions as Lemma 5.9. More precisely, we define a distribution (family) $\mathcal{D}^Y$ as a distribution of truth-tables of pseudorandom functions, which is samplable along with the witness (i.e., the description of the pseudorandom function itself), and define $\mathcal{D}^N$ as the uniform distribution over all truth tables. As in the proof of Lemma 5.9, we choose the input length of pseudorandom functions properly so that (i) every element in $\mathrm{supp}(\mathcal{D}_Y)$ is contained in $\mathsf{GapMCSP}_{\mathrm{yes}}$ with a valid witness and (ii) $\Pr_{x\sim\mathcal{D}^N}[x \in \mathsf{GapMCSP}_{\mathrm{no}}] \leq \mathsf{negl}$. Then, $(\mathsf{GapMCSP}, \mathcal{D}^Y, \mathcal{D}^N)$ is cryptographically hard, where the computational indistinguishability follows from the security of pseudorandom functions. Therefore, by Theorem 5.18, there exists a semantic secure public-key encryption scheme. □

5.7 Proof of Theorem 5.12

To prove Theorem 5.12, we introduce some notations.

For an interactive proof system (P, V) for Π of knowledge complexity $k(n)$ and its simulator S, we assume that S guesses the knowledge $\kappa(x, r)$ uniformly at random (instead of obtaining it as input), and S uses $r_S(n)$-bit randomness for each input size n. Then, for each $x \in \Pi_{\mathrm{yes}}$ with $|x| = n$, there exists a set $K_x \subseteq \{0,1\}^{r_S(n)}$ such that $|K_x|/|\{0,1\}^{r_S(n)}| = 2^{-k(n)}$ and $S(x;r)$ is computationally indistinguishable from $\langle P, V\rangle(x)$ given $r \sim K_x$. We call such a subset K_x a *useful* seed set. Note that a useful seed set is not efficiently recognizable since κ can be not efficiently computable in general. We also use $r_V(n)$ to refer to the number of random bits V requires for each input size n.

For any subset $K \subseteq \{0,1\}^{r_S(n)}$ of seeds, for each valid transcript τ with respect to the common input x, and for each $i \in [2\ell] \cup \{0\}$, where $2\ell := 2\ell(|x|)$ is the round complexity of (P, V), we let τ_i denote the prefix of τ that corresponds to the first i messages in τ (for simplicity, let $\tau_0 = \varepsilon$). For $s \in \{0,1\}^*$, we define T_s and K_s as follows

- $T_s = |\{w \in \{0,1\}^{r_S(|x|)} : \text{ the prefix of } S(x;w) \text{ is } s\}|$;
- $K_s = |\{w \in K : \text{ the prefix of } S(x;w) \text{ is } s\}|$.

In addition, we let $S_{|K}$ denote the simulator given that its randomness is selected uniformly at random from K. Namely, for each valid transcript τ,

$$\Pr_{S_{|K}}[\tau \leftarrow S_{|K}(x)] = \frac{K_\tau}{K_\varepsilon} = \frac{K_{\tau_{2\ell}}}{K_{\tau_0}}.$$

Let P_S and V_S be the simulation-based prover and the simulation-based verifier, respectively. We also consider an $S_{|K}$-based prover $P_{S|K}$ that returns a message m with respect to the history h with probability $K_{h\circ m}/K_h$ (if $K_h = 0$, the $S_{|K}$-based prover $P_{S|K}$ halts with an error message).

The following is a key lemma.

Lemma 5.20. *Let (P, V) be an interactive proof system, and let S be its simulator that uses $r_s(n)$ random bits. For every input $x \in \{0,1\}^*$, every subset $K \subseteq \{0,1\}^{r(|x|)}$, and every $\Delta \geq 0$ and $\delta \in (0,1]$, if $\mathrm{KL}(S_{|K}(x)||\langle P_{S|K}, V\rangle(x)) \leq \Delta$, then*

$$\Pr_{\tau\sim S(x)}\left[\log\frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \leq 2\delta^{-1}(\Delta + e^{-1}\log e) + \log 2\delta^{-1}\right] \geq (1-\delta)\cdot\frac{|K|}{2^{r_s(|x|)}}.$$

Proof. We use the following inequalities.

Claim 5.21. *For every $\delta \in (0,1]$,*

$$\Pr_{\tau \sim S_{|K}}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \le \log \frac{\Pr_{S_{|K}}[\tau \leftarrow S_{|K}(x)]}{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]} + \log \delta^{-1}\right] \ge 1-\delta.$$

Claim 5.22. *For every $\Delta \ge 0$ and $\delta \in (0,1]$, if $\mathrm{KL}(S_{|K}(x) || \langle P_{S|K}, V\rangle(x)) \le \Delta$, then*

$$\Pr_{\tau \sim S_{|K}}\left[\log \frac{\Pr_{S_{|K}}[\tau \leftarrow S_{|K}(x)]}{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]} \le \delta^{-1}(\Delta + e^{-1}\log e)\right] \ge 1-\delta.$$

We assume the claims above and first complete the proof of the lemma. By Claim 5.21, Claim 5.22, and the union bound, we have

$$\Pr_{\tau \sim S_{|K}}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \le 2\delta^{-1}(\Delta + e^{-1}\log e) + \log 2\delta^{-1}\right] \ge 1-\delta.$$

Thus, we have

$$\begin{aligned}\Pr_{\tau \sim S}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \le 2\delta^{-1}(\Delta + e^{-1}\log e) + \log 2\delta^{-1}\right] &\ge (1-\delta) \cdot \Pr_{w \sim \{0,1\}^{r_S(|x|)}}[w \in K] \\ &= (1-\delta) \cdot \frac{|K|}{2^{r_s(|x|)}}.\end{aligned}$$

Therefore, it suffices to show Claim 5.21 and Claim 5.22.

Proof of Claim 5.21.

$$\begin{aligned}&\mathop{\mathbb{E}}_{\tau \sim S_{|K}}\left[\frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \cdot \frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{S_{|K}}[\tau \leftarrow S_{|K}(x)]}\right] \\ &= \sum_\tau \frac{\Pr_{S_{|K}}[\tau \leftarrow S_{|K}]}{\Pr_{S_{|K}}[\tau \leftarrow S_{|K}]} \cdot \Pr_S[\tau \leftarrow S(x)] \cdot \frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \\ &= \sum_\tau 1 \cdot \frac{T_{\tau_{2\ell}}}{T_{\tau_0}} \cdot \frac{2^{-r_V(|x|)} \cdot \frac{K_{\tau_1}}{K_{\tau_0}} \cdot \frac{K_{\tau_3}}{K_{\tau_2}} \cdot \cdots \cdot \frac{K_{\tau_{2\ell-1}}}{K_{\tau_{2\ell-2}}}}{2^{-r_V(|x|)} \cdot \frac{T_{\tau_1}}{T_{\tau_0}} \cdot \frac{T_{\tau_3}}{T_{\tau_2}} \cdot \cdots \cdot \frac{T_{\tau_{2\ell-1}}}{T_{\tau_{2\ell-2}}}} \\ &= \sum_\tau \frac{K_{\tau_1}}{K_{\tau_0}} \cdot \frac{T_{\tau_2}}{T_{\tau_1}} \cdot \frac{K_{\tau_3}}{K_{\tau_2}} \cdot \cdots \cdot \frac{K_{\tau_{2\ell-1}}}{K_{\tau_{2\ell-2}}} \cdot \frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}} \\ &= \sum_\tau \Pr_{P_{S|K},V_S}[\tau \leftarrow \langle P_{S|K}, V_S\rangle(x)] \\ &\le 1,\end{aligned}$$

where τ is taken over valid transcripts such that $T_{\tau_i} \ne 0$ and $K_{\tau_i} \ne 0$ for each $i \in [2\ell]$.

By Markov's inequality,

$$\Pr_{\tau \sim S_{|K}(x)}\left[\frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \cdot \frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{S_{|K}}[\tau \leftarrow S_{|K}(x)]} \le \delta^{-1}\right] \ge 1-\delta.$$

By the assumption that S always outputs a valid transcript, it holds that $\Pr[\tau \leftarrow \langle P_{S|K}, V\rangle(x)] > 0$ for every transcript τ produced by $S_{|K}$. Thus, by arranging the above, we have

$$\Pr_{\tau \sim S_{|K}(x)}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} - \log \frac{\Pr_{S_{|K}}[\tau \leftarrow S_{|K}(x)]}{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]} \le \log \delta^{-1}\right] \ge 1 - \delta.$$

$\diamond$

Proof of Claim 5.22. Let $X = \{\tau : \Pr_{S_{|K}}[\tau \leftarrow S_{|K}(x)] > 0\}$, and let $A \subseteq X$ be a set of transcripts τ such that

$$\log \frac{\Pr_{S_{|K}(x)}[\tau \leftarrow S_{|K}(x)]}{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]} > \delta^{-1}(\Delta + e^{-1}\log e).$$

We show $\Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \le \delta$ by contradiction. Let $\eta := \delta^{-1}(\Delta + e^{-1}\log e)$ for readability.

Suppose that $\Pr_{\tau \sim S_{|K}(x)}[\tau \in A] > \delta$. Let $\bar{A} = X \setminus A$. We also define $K^A := \{w \in K : S(x;w) \in A\}$ and $K^{\bar{A}} := \{w \in K : S(x;w) \in \bar{A}\}$.

Now, we estimate the KL divergence as follows:

$$\begin{aligned}
&\mathrm{KL}(S_{|K}(x) || \langle P_{S|K}, V\rangle(x)) \\
&\quad = \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \mathop{\mathbb{E}}_{\tau \sim S_{|K^A}(x)}\left[\log \frac{\Pr_{S_{|K}(x)}[\tau \leftarrow S_{|K}(x)]}{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}\right] \\
&\qquad\qquad + \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \cdot \mathop{\mathbb{E}}_{\tau \sim S_{|K^A}(x)}\left[-\log \frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{S_{|K}(x)}[\tau \leftarrow S_{|K}(x)]}\right] \\
&\quad > \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \eta + \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \cdot \mathop{\mathbb{E}}_{\tau \sim S_{|K^A}(x)}\left[-\log \frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{S_{|K}(x)}[\tau \leftarrow S_{|K}(x)]}\right] \\
&\quad \ge \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \eta + \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \cdot \left(-\log \mathop{\mathbb{E}}_{\tau \sim S_{|K^{\bar{A}}}(x)}\left[\frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{S_{|K}(x)}[\tau \leftarrow S_{|K}(x)]}\right]\right) \\
&\quad = \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \eta \\
&\qquad\qquad + \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \cdot \left(-\log \mathop{\mathbb{E}}_{\tau \sim S_{|K^{\bar{A}}}(x)}\left[\frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{S_{|K^{\bar{A}}}(x)}[\tau \leftarrow S_{|K^{\bar{A}}}(x)] \Pr_{\tau' \sim S_{|K}(x)}[\tau' \in \bar{A}]}\right]\right) \\
&\quad = \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \eta + \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \cdot \left(-\log \mathop{\mathbb{E}}_{\tau \sim S_{|K^{\bar{A}}}(x)}\left[\frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]}{\Pr_{S_{|K^{\bar{A}}}(x)}[\tau \leftarrow S_{|K^{\bar{A}}}(x)]}\right]\right) \\
&\qquad\qquad - \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \log \frac{1}{\Pr_{\tau' \sim S_{|K}(x)}[\tau' \in \bar{A}]} \\
&\quad \ge \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \eta + \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \cdot (-\log 1) - \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \log \frac{1}{\Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}]} \\
&\quad = \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \eta - \Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}] \log \frac{1}{\Pr_{\tau \sim S_{|K}(x)}[\tau \in \bar{A}]} \\
&\quad \ge \Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \cdot \eta - e^{-1}\log e
\end{aligned}$$

$$
\begin{aligned}
&> \delta \cdot \eta - e^{-1} \log e \\
&= \Delta + e^{-1} \log e - e^{-1} \log e \\
&= \Delta,
\end{aligned}
$$

where the first inequality holds by the definition of A and $\Pr_{\tau \sim S_{|K}(x)}[\tau \in A] > \delta > 0$, the second inequality follows from Jensen's inequality, the third inequality follows from

$$\mathop{\mathbb{E}}_{\tau \sim S_{|K\bar{A}}(x)} \left[\frac{\Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V \rangle(x)]}{\Pr_{S_{|K\bar{A}}(x)}[\tau \leftarrow S_{|K\bar{A}}(x)]} \right] \leq \sum_{\tau} \Pr_{P_{S|K},V}[\tau \leftarrow \langle P_{S|K}, V \rangle(x)] \leq 1,$$

the fourth inequality holds because $p \log p^{-1} \leq e^{-1} \log e$ for every $p \in (0, 1]$, and the last inequality follows from $\Pr_{\tau \sim S_{|K}(x)}[\tau \in A] > \delta$.

The above contradicts the assumption that $\mathrm{KL}(S_{|K}(x) || \langle P_{S|K}, V \rangle(x)) \leq \Delta$. Thus, we have $\Pr_{\tau \sim S_{|K}(x)}[\tau \in A] \leq \delta$. ◇

□

By selecting K as the useful seed set for the simulator, we obtain the following.

Lemma 5.23. *Let (P, V) be an interactive proof system for a promise problem Π of computational knowledge complexity $k(n)$. Let S be its simulator whose useful seed set is $K := K_x$ for $x \in \Pi_{\text{yes}}$. Then, for every $\Delta \geq 1$, for every input $x \in \Pi_{\text{yes}}$ satisfying $\mathrm{KL}(S_{|K}(x) || \langle P_{S|K}, V \rangle(x)) \leq \Delta$, and for every $\delta \in (0, 1]$,*

$$\Pr_{\tau \sim S(x)} \left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V \rangle(x)]} \leq 3\delta^{-1} + 2\delta^{-1}\Delta \right] \geq (1 - \delta) \cdot 2^{-k(n)}.$$

We also show the following lemma.

Lemma 5.24. *Let (P, V) be an interactive proof system for a promise problem Π of computational knowledge complexity $k(n) = O(\log n)$, and let S be its simulator whose useful seed set is $K := K_x$ for $x \in \Pi_{\text{yes}}$. Then, there exists a polynomial-time-computable auxiliary-input function $f^Y = \{f_x^Y\}_{x \in \{0,1\}^*}$ such that f^Y is one-way almost everywhere on $I_Y := \{x \in \Pi_{\text{yes}} : \mathrm{KL}(S_{|K}(x) || \langle P_{S|K}, V \rangle(x)) > 3k(|x|)\}$.*

Proof. The lemma follows from Lemma 4.8, in which we (implicitly) construct a false entropy generator almost everywhere on I_Y, and it implies a secure pseudorandom generator and thus a one-way function almost everywhere on I_Y [cf. HILL99; Vad06, Appendix B].

Let $2\ell := 2 \cdot \ell(n)$ be the round complexity. For each $i \in [2\ell]$, let $S_{|K}(x)_i$ denote the prefix of $S_{|K}(x)$ up to the i-th message. We also define $\langle P, V \rangle(x)_i$ in the same manner.

For every $\tau \in \mathrm{supp}(S_{|K}(x))$,

$$\begin{aligned}
\log \frac{\Pr[\tau \leftarrow S_{|K}(x)]}{\Pr[\tau \leftarrow \langle P_{S|K}, V\rangle(x)]} &= \log \frac{\frac{K_{\tau_1}}{K_{\tau_0}} \cdot \frac{K_{\tau_2}}{K_{\tau_1}} \cdot \cdots \cdot \frac{K_{\tau_{2\ell}}}{K_{\tau_{2\ell-1}}}}{2^{-r_V(|x|)} \cdot \frac{K_{\tau_1}}{K_{\tau_0}} \cdot \frac{K_{\tau_3}}{K_{\tau_2}} \cdot \cdots \cdot \frac{K_{\tau_{2\ell-1}}}{K_{\tau_{2\ell-2}}}} \\
&= r_V(|x|) + \log \frac{K_{\tau_2}}{K_{\tau_1}} \frac{K_{\tau_4}}{K_{\tau_3}} \cdots \frac{K_{\tau_{2\ell}}}{K_{\tau_{2\ell-1}}} \\
&= r_V(|x|) + \sum_{i=1}^{\ell} \log \frac{K_{\tau_{2i}}/K_{\tau_0}}{K_{\tau_{2i-1}}/K_{\tau_0}} \\
&= r_V(|x|) + \sum_{i=1}^{\ell} \log \Pr\left[\tau_{2i} \leftarrow S_{|K}(x)_{2i} \middle| \tau_{2i-1} \leftarrow S_{|K}(x)_{2i-1}\right].
\end{aligned}$$

By taking the expectation over $\tau \sim S_{|K}(x)$, we have

$$\mathrm{KL}(S_{|K}(x) || \langle P_{S|K}, V\rangle(x)) = r_V(|x|) - \sum_{i=1}^{\ell} H\left(S_{|K}(x)_{2i} \middle| S_{|K}(x)_{2i-1}\right)$$

Thus, for every $x \in I_Y$,

$$\sum_{i=1}^{\ell} H\left(S_{|K}(x)_{2i} \middle| S_{|K}(x)_{2i-1}\right) < r_V(|x|) - 3k(|x|). \tag{1}$$

By contrast, since V outputs its internal randomness as the final message, we have

$$\sum_{i=1}^{\ell} H\left(\langle P, V\rangle(x)_{2i} | \langle P, V\rangle(x)_{2i-1}\right) = r_V(|x|). \tag{2}$$

We exploit the additive gap $3k(|x|)$ between Eqs. (1) and (2) as false entropy.

For each $i \in [\ell]$, we consider the following two joint distributions:

- $(\mathcal{X}^i_x, \mathcal{Y}^i_x)$: a joint distribution selected according to $(S(x;w)_{2i}, S(x;w)_{2i-1})$ for $w \sim \{0,1\}^{r_S(|x|)}$.
- $(\bar{\mathcal{X}}^i_x, \bar{\mathcal{Y}}^i_x)$: a joint distribution selected according to the following procedure: Select $w \sim \{0,1\}^{r_S(|x|)}$. If $w \notin K$, then $(\bar{\mathcal{X}}^i_x, \bar{\mathcal{Y}}^i_x) = (S(x;w)_{2i}, S(x;w)_{2i-1})$; otherwise (if $w \in K$), $(\bar{\mathcal{X}}^i_x, \bar{\mathcal{Y}}^i_x) = (\langle P, V\rangle(x)_{2i}, \langle P, V\rangle(x)_{2i-1})$.

It is easily observed that $(\mathcal{X}^i_x, \mathcal{Y}^i_x)$ is efficiently samplable (note that $(\mathcal{X}'_x, \mathcal{Y}'_x)$ is not efficiently samplable in general). In addition, since (i) $\Pr_w[w \in K] \geq 2^{-k(|n|)} = 1/\mathsf{poly}(|x|)$ and (ii) $S_{|K}(x)$ and $\langle P, V\rangle(x)$ are computationally indistinguishable, $(\mathcal{X}^i_x, \mathcal{Y}^i_x)$ and $(\bar{\mathcal{X}}^i_x, \bar{\mathcal{Y}}^i_x)$ are also computationally indistinguishable.

We define a random variable E as an indicator for the event that $w \in K$. Then, by Lemma 4.9,

$$\begin{aligned}
H(\mathcal{X}^i | E, \mathcal{Y}^i) &= \Pr_w[w \in K] H(S_{|K}(x)_{2i} | S_{|K}(x)_{2i-1}) + \Pr_w[w \notin K] H(S_{|\bar{K}}(x)_{2i} | S_{|\bar{K}}(x)_{2i-1}); \\
H(\bar{\mathcal{X}}^i_x | E, \bar{\mathcal{Y}}^i_x) &= \Pr_w[w \in K] H(\langle P, V\rangle(x)_{2i} | \langle P, V\rangle(x)_{2i-1}) + \Pr_w[w \notin K] H(S_{|\bar{K}}(x)_{2i} | S_{|\bar{K}}(x)_{2i-1}).
\end{aligned}$$

Thus, by Eqs. (1) and (2),

$$\begin{aligned}
&\sum_{i=1}^{\ell} H(\bar{\mathcal{X}}_x^i|E,\bar{\mathcal{Y}}_x^i) - \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|E,\mathcal{Y}_x^i)\\
&\qquad = \Pr_w[w\in K]\left(\sum_{i=1}^{\ell} H(\langle P,V\rangle(x)_{2i}|\langle P,V\rangle(x)_{2i-1}) - \sum_{i=1}^{\ell} H(S_{|K}(x)_{2i}|S_{|K}(x)_{2i-1})\right)\\
&\qquad > \frac{1}{2^{k(|x|)}}\cdot 3k(|x|).
\end{aligned}$$

For now, we assume the following claim shown by a simple calculation and continue the proof.

Claim 5.25.

$$\sum_{i=1}^{\ell} H(\mathcal{X}_x^i|E,\mathcal{Y}_x^i) \geq \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|\mathcal{Y}_x^i) - H(E).$$

Claim 5.25 implies that

$$\begin{aligned}
\sum_{i=1}^{\ell} H(\bar{\mathcal{X}}_x^i|E,\bar{\mathcal{Y}}_x^i) - \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|E,\mathcal{Y}_x^i) &\leq \sum_{i=1}^{\ell} H(\bar{\mathcal{X}}_x^i|\bar{\mathcal{Y}}_x^i) - \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|E,\mathcal{Y}_x^i)\\
&\leq \sum_{i=1}^{\ell} H(\bar{\mathcal{X}}_x^i|\bar{\mathcal{Y}}_x^i) - \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|\mathcal{Y}_x^i) + H(E).
\end{aligned}$$

From the two inequalities above,

$$\begin{aligned}
\sum_{i=1}^{\ell} H(\bar{\mathcal{X}}_x^i|\bar{\mathcal{Y}}_x^i) - \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|\mathcal{Y}_x^i) &\geq \sum_{i=1}^{\ell} H(\bar{\mathcal{X}}_x^i|E,\bar{\mathcal{Y}}_x^i) - \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|E,\mathcal{Y}_x^i) - H(E)\\
&> \frac{3k(|x|)}{2^{k(|x|)}} - H(E)\\
&= \frac{3k(|x|)}{2^{k(|x|)}} - \frac{1}{2^{k(|x|)}}\log 2^{k(|x|)} - (1-\frac{1}{2^{k(|x|)}})\log\frac{1}{1-2^{-k(|x|)}}\\
&\geq \frac{3k(|x|)}{2^{k(|x|)}} - \frac{1}{2^{k(|x|)}}\log 2^{k(|x|)} - \frac{1}{2^{k(|x|)}}\log 2^{k(|x|)}\\
&= \frac{k(|x|)}{2^{k(|x|)}}.
\end{aligned}$$

where the third inequality holds because $p\log p^{-1} \geq (1-p)\log(1-p)^{-1}$ for $p\in(0,1/2]$.

For a random variable I selected according to the uniform distribution over $[\ell]$,

$$H(\bar{\mathcal{X}}_x^I|\bar{\mathcal{Y}}_x^I) - H(\mathcal{X}_x^I|\mathcal{Y}_x^I) \geq \frac{1}{\ell(|x|)}\cdot\frac{k(|x|)}{2^{k(|x|)}} \geq \frac{1}{\mathsf{poly}(|x|)},$$

where the last inequality follows from $k(|x|) = O(\log(|x|))$.

The joint distribution $(\mathcal{X}_x^I,\mathcal{Y}_x^I)$ is efficiently samplable for given x and computationally indistinguishable from $(\bar{\mathcal{X}}_x^I,\bar{\mathcal{Y}}_x^I)$ because $(\mathcal{X}_x^i,\mathcal{Y}_x^i)$ and $(\bar{\mathcal{X}}_x^i,\bar{\mathcal{Y}}_x^i)$ are computationally indistinguishable for all $i\in[\ell]$. Thus, by Lemma 4.8, we can construct a one-way function f_x from $(\mathcal{X}_x^I,\mathcal{Y}_x^I)$ (i.e., $S(x)$) which is secure almost everywhere on I_Y.

Finally, we present the proof of Claim 5.25.

Proof of Claim 5.25. The claim is verified as the following calculation:

$$\begin{aligned}
\sum_{i=1}^{\ell} H(\mathcal{X}_x^i|E,\mathcal{Y}_x^i) &= \sum_{i=1}^{\ell} H(S(x)_{2i}|E,S(x)_{2i-1}) \\
&= \sum_{i=1}^{\ell} H(S(x)_{2i}|E,S(x)_1,\dots,S(x)_{2i-1}) \\
&= H(E,S(x)_1,\dots,S(x)_{2\ell}) - H(E) - \sum_{i=1}^{\ell} H(S(x)_{2i-1}|E,S(x)_1,\dots,S(x)_{2i-2}) \\
&\geq H(S(x)_1,\dots,S(x)_{2\ell}) - H(E) - \sum_{i=1}^{\ell} H(S(x)_{2i-1}|S(x)_1,\dots,S(x)_{2i-2}) \\
&= \sum_{i=1}^{\ell} H(S(x)_{2i}|S(x)_1,\dots,S(x)_{2i-1}) - H(E) \\
&= \sum_{i=1}^{\ell} H(S(x)_{2i}|S(x)_{2i-1}) - H(E) \\
&= \sum_{i=1}^{\ell} H(\mathcal{X}_x^i|\mathcal{Y}_x^i) - H(E),
\end{aligned}$$

where the third and fourth equality follows from the chain rule. ⋄

□

We also use the following theorem.

Theorem 5.26 ([IL90; HN23])**.** *For every polynomial-time computable function $f = \{f_z : \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z\in\{0,1\}^*}$ and every polynomials q and q', there exist a polynomial-time randomized oracle machine R_{est}, a polynomial-time computable function $\tilde{f} = \{\tilde{f}_z : \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z\in\{0,1\}^*}$, and a polynomial p such that for every $z \in \{0,1\}^*$ and every oracle A, if $\Pr_w[A(\tilde{f}_z(w)) \notin \tilde{f}_z^{-1}(\tilde{f}_z(w))] \leq 1/p(|z|)$, then*

$$\Pr_{R,w}\left[-\log p_{f_z(w)} - q(|z|) \leq R_{est}^A(z, f_z(w)) \leq -\log p_{f_z(w)} + q(|z|)\right] \geq 1 - 1/q'(|z|),$$

where $p_{f_z(w)} := \Pr_{w'}[f_z(w) = f_z(w')]$.

Now, we complete the proof of Theorem 5.12.

Proof of Theorem 5.12. Let Π be a promise problem that has an interactive proof system of computational knowledge complexity at most $k(n) = O(\log n)$ and negligible soundness error, where n represents the length of an instance. By Lemma 5.24, there exists a polynomial-time-computable auxiliary-input function $f^Y = \{f_x^Y\}_{x\in\{0,1\}^*}$ secure almost everywhere on $I_Y := \{x \in \Pi_{\text{yes}} : \mathrm{KL}(S_{|K}(x)||\langle P_{S|K}, V\rangle(x)) > 3k(|x|)\}$. Let $\ell := \ell(|x|)$.

It suffices to construct a polynomial-time-computable function and a black-box reduction from solving $(\Pi_{\text{yes}} \setminus I_Y, \Pi_{\text{no}})$ to inverting a function on average.

By Lemma 5.23, for every $x \in \Pi_{\mathrm{yes}} \setminus I_Y$,

$$\Pr_{\tau \sim S(x)}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \leq 6 + 12k(n)\right] \geq \frac{1}{2} \cdot 2^{-k(n)}. \tag{3}$$

By contrast, we can observe the following.

Claim 5.27. *For every $x \in \Pi_{\mathrm{no}}$,*

$$\Pr_{\tau \sim S(x)}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} > 9 + 12k(n)\right] \geq 1 - \mathsf{negl}(n). \tag{4}$$

We assume the claim above for now and continue the proof.

Recall that for any $\tau \in \mathrm{supp}(S(x))$,

$$\begin{aligned}\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} &= \log \frac{\frac{T_{\tau_1}}{T_{\tau_0}} \cdot \frac{T_{\tau_2}}{T_{\tau_1}} \cdot \cdots \cdot \frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}}{2^{-r_V(|x|)} \cdot \frac{T_{\tau_1}}{T_{\tau_0}} \cdot \frac{T_{\tau_3}}{T_{\tau_2}} \cdot \cdots \cdot \frac{T_{\tau_{2\ell-1}}}{T_{\tau_{2\ell-2}}}} \\ &= r_V(|x|) + \log\left(\frac{T_{\tau_2}}{T_{\tau_1}} \cdots \frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}\right).\end{aligned}$$

We employ Theorem 5.26 to reduce estimating $T_{\tau_0}, \ldots, T_{\tau_{2\ell}}$ to inverting a function. We define a polynomial-time-computable function $g = \{g_x\}_x$ as follows:

$$g_x(i, w) = (i, S(x; w)_{\leq i}),$$

where $i \sim [2\ell]$, $w \sim r_S(|x|)$, and $S(x; w)_{\leq i}$ represents the first i messages in the transcript produced by $S(x; w)$. Since $\ell = \ell(n) = \mathsf{poly}(n)$ and $2^{k(n)} = \mathsf{poly}(n)$, by Theorem 5.26, there exist a polynomial-time randomized oracle machine R_{est}, a polynomial-time computable function $\tilde{g} = \{\tilde{g}_x\}_{x \in \{0,1\}^*}$, and a polynomial p such that for every $x \in \{0,1\}^*$ and every oracle A, if $\Pr_w[A(\tilde{g}_x(w)) \notin \tilde{g}_x^{-1}(\tilde{g}_x(w))] \leq 1/p(|x|)$, then

$$\Pr_{R_{est}, i, w}\left[-\log p_{g_x(i,w)} - \frac{1}{2\ell} \leq R_{est}^A(x, g_x(i,w)) \leq -\log p_{g_x(i,w)} + \frac{1}{2\ell}\right] \geq 1 - \frac{1}{16\ell} \cdot 2^{-k(n)}.$$

Recall that $p_{g_x(i,w)} := \Pr_{i',w'}[g_x(i', w') = g_x(i, w)]$. Since i is selected from $[2\ell]$ uniformly at random, it holds that

$$\Pr_{R_{est}, w}\left[\forall i \in [2\ell] \ -\log p_{g_x(i,w)} - \frac{1}{2\ell} \leq R_{est}^A(x, i, S(x;w)_{\leq i}) \leq -\log p_{g_x(i,w)} + \frac{1}{2\ell}\right] \geq 1 - \frac{1}{8} \cdot 2^{-k(n)}. \tag{5}$$

For each $\tau \in \mathrm{supp}(S(x))$ and $i \in [2\ell]$,

$$p_{g_x(i,\tau_i)} = \Pr_{i',w}[i' = i \text{ and } S(x; w)_{\leq i} = \tau_i] = \frac{1}{2\ell} \cdot \frac{T_{\tau_i}}{r_S(|x|)};$$

thus,

$$\log T_{\tau_i} = \log p_{g_x(i,\tau_i)} + \log 2\ell + \log r_S(|x|),$$

and

$$\log\left(\frac{T_{\tau_2}}{T_{\tau_1}}\cdots\frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}\right)=\sum_{i=1}^{\ell}\left((-\log p_{g_x(2i-1,\tau_{2i-1})})-(-\log p_{g_x(2i,\tau_{2i})})\right).$$

We consider the case where R_{est} and w satisfy the event in Eq. (5). Then, for

$$\Delta(x):=\sum_{i=1}^{\ell}\left(R_{est}^{A}(x,2i-1,\tau_{2i-1})-R_{est}^{A}(x,2i,\tau_{2i})\right),$$

we have

$$\begin{aligned}\Delta(x)&\leq\sum_{i=1}^{\ell}\left((-\log p_{g_x(2i-1,\tau_{2i-1})})-(-\log p_{g_x(2i,\tau_{2i})})\right)+\sum_{i=1}^{2\ell}\frac{1}{2\ell}\\&=\log\left(\frac{T_{\tau_2}}{T_{\tau_1}}\cdots\frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}\right)+1,\end{aligned}\tag{6}$$

and

$$\begin{aligned}\Delta(x)&\geq\sum_{i=1}^{\ell}\left((-\log p_{g_x(2i-1,\tau_{2i-1})})-(-\log p_{g_x(2i,\tau_{2i})})\right)-\sum_{i=1}^{2\ell}\frac{1}{2\ell}\\&=\log\left(\frac{T_{\tau_2}}{T_{\tau_1}}\cdots\frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}\right)-1.\end{aligned}\tag{7}$$

Now, we specify the black-box reduction $R^{?}$ from solving $(\Pi_{\mathrm{yes}}\setminus I_Y,\Pi_{\mathrm{no}})$ to inverting $\tilde{g}$ based on the argument above. For given oracle access to A that inverts $\tilde{g}$, the reduction $R^A(x)$ executes R_{est}^A to estimate the probability that $r_V(|x|)+\Delta(x)<7.5+12k(n)$ holds over the choice of randomness for R_{est} and $S(x)$ by empirical estimation of accuracy error $\pm(1/16)2^{-k(n)}$ and confidence error $1/3$. If the estimated probability is greater than $(7/32)2^{-k(n)}$, the reduction R outputs 1; otherwise, R outputs 0.

It is not hard to verify that R is polynomial-time computable because the empirical estimation is done by at most $O(2^{2k(n)})=\mathsf{poly}(n)$ trials. We show the correctness of R. Suppose that A inverts $\tilde{g}$ successfully with failure probability at most $1/p(|x|)$.

In the case where $x\in\Pi_{\mathrm{yes}}\setminus I_Y$, by Eqs. (5) and (6),

$$\Pr_{R_{est},w}\left[r_V(|x|)+\Delta(x)\leq r_V(|x|)+\log\left(\frac{T_{\tau_2}}{T_{\tau_1}}\cdots\frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}\right)+1\right]\geq 1-\frac{1}{8}\cdot 2^{-k(n)}.$$

Moreover, by Eq. (3),

$$\Pr_{\tau\sim S(x)}\left[r_V(|x|)+\log\left(\frac{T_{\tau_2}}{T_{\tau_1}}\cdots\frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}\right)\leq 6+12k(n)\right]\geq\frac{1}{2}\cdot 2^{-k(n)}.$$

Thus, we have

$$\Pr_{R_{est},w}\left[r_V(|x|)+\Delta(x)\leq 7+12k(n)\right]\geq\frac{3}{8}\cdot 2^{-k(n)}.$$

By contrast, in the case where $x \in \Pi_{\mathrm{no}}$, Eqs. (5) and (7) imply that

$$\Pr_{R_{est},w}\left[r_V(|x|) + \Delta(x) \geq r_V(|x|) + \log\left(\frac{T_{\tau_2}}{T_{\tau_1}} \cdots \frac{T_{\tau_{2\ell}}}{T_{\tau_{2\ell-1}}}\right) - 1\right] \geq 1 - \frac{1}{8} \cdot 2^{-k(n)}.$$

Moreover, by Eq. (4),

$$\Pr_{\tau \sim S(x)}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} > 9 + 12k(n)\right] \geq 1 - \mathsf{negl}(n).$$

Thus, by the union bound,

$$\Pr_{R_{est},w}[r_V(|x|) + \Delta(x) \geq 8 + 12k(n)] \geq 1 - \left(\mathsf{negl}(n) + \frac{1}{8} \cdot 2^{-k(n)}\right),$$

and

$$\Pr_{R_{est},w}[r_V(|x|) + \Delta(x) < 7.5 + 12k(n)] \leq \mathsf{negl}(n) + \frac{1}{8} \cdot 2^{-k(n)}.$$

Thus, by approximating the probability that $r_V(|x|) + \Delta(x) < 7.5 + 12k(n)$ with additive accuracy error $\pm 1/16 \cdot 2^{-k(n)}$, the reduction R^A can distinguish the cases where (i) $x \in \Pi_{\mathsf{yes}} \setminus I_Y$ and $x \in \Pi_{\mathrm{no}}$ with probability at least $2/3$.

In the reminder, we show Claim 5.27.

Proof of Claim 5.27. Let $B \subseteq \operatorname{supp} S(x)$ be the subset of transcripts τ that satisfies

$$\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \leq 9 + 12k(n).$$

Then, we have

$$\sum_{\tau \in B} \Pr_S[\tau \leftarrow S(x)] \leq 2^{9+12k(n)} \Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]$$

However, by the assumption that S always outputs an accepting transcript, the soundness implies that

$$\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)] \leq \mathsf{negl}(n).$$

Thus, we conclude that

$$\begin{aligned}\Pr_{\tau \sim S(x)}\left[\log \frac{\Pr_S[\tau \leftarrow S(x)]}{\Pr_{P_S,V}[\tau \leftarrow \langle P_S, V\rangle(x)]} \leq 9 + 12k(n)\right] &\leq 2^{9+12k(n)} \cdot \mathsf{negl}(n) \\ &\leq \mathsf{negl}(n),\end{aligned}$$

which implies Eq. (4). ◇

□

6 Nontrivial Savings of Computational Knowledge Complexity

In this section, we show the following theorem.

Theorem 6.1. *Let $s, k\colon \mathbb{N} \to \mathbb{N}$ be functions satisfying that $s(k(n)) = n^{\omega(1)}$. If there exists an auxiliary-input one-way function with sufficiently large security against $\mathsf{SIZE}[\mathsf{poly}(s(m))]$ with success probability at most $\mathsf{negl}(s(m))$ (where m represents the length of auxiliary input), then NP has an interactive proof system of computational knowledge complexity $k(n)$ and negligible soundness error.*

The proof of Theorem 6.1 is based on two key theorems.

The first one is the small support min-max theorem. We introduce some notions. Let M be an $r \times c$ $[-1, 1]$-valued matrix. We regard M as a description of a zero-sum game, where a row player selects a row $i \in [r]$, a column player selects a column $j \in [c]$, and then the row (resp. column) player loses (resp. wins) the reward $M_{i,j}$. We call $i \in [r]$ and $j \in [c]$ a pure strategy of the row and column player, respectively. A mixed strategy of the row (resp. column) player is a distribution over $[r]$ (resp. $[c]$). Let $\mathcal{R}$ and $\mathcal{C}$ be a set of all mixed strategies of the row and column players, respectively. Then, the standard min-max theorem states that

$$v(M) := \min_{p \in \mathcal{R}} \max_{j \in [c]} \mathop{\mathbb{E}}_{i \sim p} [M_{i,j}] = \max_{q \in \mathcal{C}} \min_{i \in [r]} \mathop{\mathbb{E}}_{i \sim q} [M_{i,j}].$$

We call $v(M)$ above the value of game M.

The small-support min-max theorem [LY94] states that the value of a game M is accomplished by a mixed strategy defined as a uniform distribution over a relatively small support with respect to the number of opponent's pure strategy. For any $k \in \mathbb{N}$, let $\mathcal{R}_k$ (resp. $\mathcal{C}_k$) be a set of uniform distributions over a multi-set $S \subseteq [r]$ (resp. $S \subseteq [c]$) with $|S| = k$. Note that $\mathcal{R}_k \subseteq \mathcal{R}$ and $\mathcal{C}_k \subseteq \mathcal{C}$. The small-support zero-sum game is stated as follows.

Theorem 6.2 (Small-support min-max theorem [LY94]). *Let M be an $r \times c$ $[-1, 1]$-valued matrix. For every $\delta > 0$, let $k_r = 10 \ln c/\delta^2$ and $k_c = 10 \ln r/\delta^2$. Then,*

$$(v(M) \leq) \min_{p \in \mathcal{R}_{k_r}} \max_{j \in [c]} \mathop{\mathbb{E}}_{i \sim p} [M_{i,j}] \leq v(M) + \delta$$

and

$$v(M) - \delta \leq \max_{q \in \mathcal{C}_{k_c}} \min_{i \in [r]} \mathop{\mathbb{E}}_{i \sim q} [M_{i,j}] \ (\leq v(M)).$$

The second one is the construction of zero-knowledge proof for NP based on one-way functions [GMW91]. Particularly, we use the previous work as a building block in the following form.

Theorem 6.3 ([GMW91]). *Let $f = \{f_z\}_{z \in \{0,1\}^*}$ be a polynomial-time-computable auxiliary-input function. For every $\Pi \in \mathsf{NP}$, there exist an interactive proof system (P_z, V_z) for Π (with negligible completeness and soundness error), a polynomial-time randomized algorithm S_z, where z is an auxiliary input, a polynomial-time randomized oracle machine $R^?$, and a constant $c > 0$ such that for every $x \in \Pi_{\mathrm{yes}}$, every $\gamma^{-1} \in \mathbb{N}$, and every $z \in \{0,1\}^*$, if there exists a nonuniform randomized polynomial-time algorithm A such that*

$$\left| \Pr_{P_z, V_z, A}[A(z, \mathsf{view}_{V_z}(\langle P_z, V_z \rangle(x))) = 1] - \Pr_{S_z, A}[A(z, S_z(x)) = 1] \right| \geq \gamma$$

then

$$\Pr_{w,R,A}[R^A(1^{\gamma^{-1}}, z, f_z(w)) \in f_z^{-1}(f_z(w))] \geq \gamma^c.$$

Now, we prove Theorem 6.1.

Proof of Theorem 6.1. Let $s, k \colon \mathbb{N} \to \mathbb{N}$ be functions satisfying that $s(k(n)) = n^{\omega(1)}$. Let $f = \{f_z\}_{z \in \{0,1\}^*}$ be a polynomial-time-computable auxiliary-input function with sufficiently large security against $\mathsf{SIZE}[\mathsf{poly}(s(|z|))]$. For $\Pi \in \mathsf{NP}$, we construct an interactive proof system of computational knowledge complexity $k(n)$ (with negligible soundness error).

By Theorem 6.3, there exist an interactive proof system (P_z, V_z) for Π (with negligible completeness and soundness error), a polynomial-time randomized algorithm S_z, and a polynomial-time randomized oracle machine $R^?$ satisfying the properties of Theorem 6.3.

Fix $n \in \mathbb{N}$ arbitrarily. Let $m := k(n)$. We define a zero-sum game M as follows: Each row is indexed by $z \in \{0,1\}^m$, each column is indexed by circuits C of size $s(m)$, and each entry of M is defined as

$$M_{z,C} := \left| \Pr_{P_z,V_z}[C(z, \mathsf{view}_{V_z}(\langle P_z, V_z \rangle(x))) = 1] - \Pr_{S_z}[C(z, S_z(x)) = 1] \right|.$$

We also define $v(m) \in [0,1]$ as

$$v(m) = \min_z \max_C \left| \Pr_{P_z,V_z}[C(z, \mathsf{view}_{V_z}(\langle P_z, V_z \rangle(x))) = 1] - \Pr_{S_z}[C(z, S_z(x)) = 1] \right|.$$

By the definition of the value of M, we have $v(M) \leq v(m)$. Note that $v(m)$ is uniformly computable in the time-unbounded setting. In addition, $v(m) \leq \mathsf{negl}(s(m))$; otherwise, by Theorem 6.3, there exists a $\mathsf{poly}(s(m))$-size circuit that inverts f_z for all $z \in \{0,1\}^m$ with success probability at least $1/\mathsf{poly}(s(m))$.

Next, we apply the small support min-max theorem (Theorem 6.2). Then, there exists a multiset $S_m \subseteq \{0,1\}^m$ of size $k_m = O(s(m)^{2\log s(m)} \log(s(m)))$ such that for every circuit C of size $s(m)$,

$$\mathop{\mathbb{E}}_{z \sim S_m}\left[\left| \Pr_{P_z,V_z,A}[A(z, \mathsf{view}_{V_z}(\langle P_z, V_z \rangle(x))) = 1] - \Pr_{S_z,A}[A(z, S_z(x)) = 1] \right|\right] \leq v(m) + s(m)^{-\log s(m)}$$

Now, we present an interactive proof system (P', V') of computational knowledge complexity $k(n)$ (with negligible soundness error) for Π. Let $x \in \{0,1\}^n$ be a common input for (P', V'), and let $m = k(n)$. First, by exhaustive search, P' finds the lexicographically first multi-set S_0 of size k_m satisfying

$$\begin{aligned}\mathop{\mathbb{E}}_{z \sim S_0}\left[\left| \Pr_{P_z,V_z,A}[A(z, \mathsf{view}_{V_z}(\langle P_z, V_z \rangle(x))) = 1] - \Pr_{S_z,A}[A(z, S_z(x)) = 1] \right|\right] &\leq v(m) + s(m)^{-\log s(m)} \\ &\leq \mathsf{negl}(s(m)). \end{aligned} \tag{8}$$

Recall that such an S_0 must exist because of the argument above. Next, P' sends $z \sim S_0$ to V'. Then, (P', V') executes (P_z, V_z) for the common input x and V' makes the same decision as V_z.

The completeness and soundness of (P', V') follows from that of (P, V). We show that the knowledge complexity of (P, V) is $k(n)$ by constructing a simulator S' for (P', V'). Let $S_0 = \{z_1, \ldots, z_{k_m}\}$, and let $\kappa : \{0,1\}^n \times \{0,1\}^{\mathsf{poly}(n)} \to \{0,1\}^{k(n)}$ be a function defined as

$$\kappa(x, w) = z_i,$$

where $i \in [k_m]$ is an index indicated by $w_{[\lceil \log k_m \rceil]}$. The simulator S', given a common input $x \in \{0,1\}^n$, a randomness $w \in \{0,1\}^{\mathsf{poly}(n)}$, and advice $z = \kappa(x, w)$, simulates $S_z(x)$ by using suffix of w not used in κ and outputs the same transcript. Suppose that there exists a nonuniform polynomial-time adversary A such that

$$\left| \Pr_{z\sim S_0, P_z, V_z, A}[A(z, \mathsf{view}_{V_z}(\langle P_z, V_z\rangle(x))) = 1] - \Pr_{z\sim S_0, S_z, A}[A(z, S_z(x)) = 1] \right|$$
$$\left| \Pr_{P', V', A}[A(z, \mathsf{view}_{V'}(\langle P', V'\rangle(x))) = 1] - \Pr_{S', A}[A(z, S'(x)) = 1] \right| \geq \frac{1}{\mathsf{poly}(n)}$$

Then,

$$\mathop{\mathbb{E}}_{z\sim S_0}\left[\left| \Pr_{P_z, V_z, A}[A(z, \mathsf{view}_{V_z}(\langle P_z, V_z\rangle(x))) = 1] - \Pr_{S_z, A}[A(z, S_z(x)) = 1] \right| \right] \geq \frac{1}{\mathsf{poly}(n)} \geq \frac{1}{s(k(n))} = \frac{1}{s(m)},$$

which contradicts Eq. (8), where we used the assumption that $s(k(n)) = n^{\omega(1)}$. Thus, S' is a computational simulator for (P', V') with κ, and its knowledge complexity is $k(n)$. □

Next, we prove Theorems 2.4 and 2.5 by choosing $k(\cdot)$ and $s(\cdot)$ above properly.

Corollary 6.4 (Theorem 2.5)**.** *If there exists an exponentially secure auxiliary-input one-way function with sufficiently large security, then for every increasing function $k(n) = \omega(\log n)$, NP has an interactive proof system of computational knowledge complexity $k(n)$ (with negligible soundness error).*

Proof. Let $k(n) = \omega(\log n)$ be an increasing function. We define an increasing function $s(n)$ as

$$s(n) := \max_{m\in\mathbb{N}: k(m)\leq n} m^{\sqrt{\frac{k(m)}{\log m}}}.$$

Note that $s(k(n)) = n^{\sqrt{\frac{k(n)}{\log n}}}$ for every $n \in \mathbb{N}$. Since $k(n) = \omega(\log n)$, we have $s(k(n)) = n^{\omega(1)}$.

We also observe that $\log s(k(n)) = \sqrt{k(n) \log n} = o(k(n))$. Thus, $s(m) = 2^{o(m)}$, and the assumption implies that there exists an auxiliary-input one-way function secure against $\mathsf{SIZE}[\mathsf{poly}(s(m))]$ with success probability at most $\mathsf{negl}(s(m))$. By Theorem 6.1, we conclude that NP has an interactive proof system of computational knowledge complexity $k(n)$ and negligible soundness error. □

Corollary 6.5 (Theorem 2.4)**.** *If there exists an auxiliary-input one-way function secure against $\mathsf{P/poly}$ with sufficiently large security, then there exists a function $k(n)$ such that NP has an interactive proof system of computational knowledge complexity $k(n)$ (with negligible soundness error) and $k(n) \leq n^{\epsilon}$ for every constant $\epsilon > 0$ and every large enough n.*

Proof. Let $f = \{f_z\}_{z\in\{0,1\}^*}$ be an auxiliary-input one-way function. We can fix an increasing function $\gamma(m) = m^{\omega(1)}$ such that every polynomial-size (family of) inverters successfully inverts f_z with probability at most $1/\gamma(|z|)$ for any large enough $z \in \{0,1\}^*$ [cf. Bel02]. We define an increase function $c(m)$ so that for each $m \in \mathbb{N}$, the minimum circuit size that inverts f_z for all $z \in \{0,1\}^{\leq m}$ with probability greater than $1/\gamma(|z|)$ is $c(m) + 1$. By the choice of γ, we have that $c(m) = m^{\omega(1)}$. Let $s(m) = \min\{m^{\sqrt{\log \gamma(m)/\log m}}, m^{\sqrt{\log c(m)/\log m}}\} = m^{\omega(1)}$. Then, we have that f is secure against $\mathsf{SIZE}[\mathsf{poly}(s(m))]$ with success probability at most $\mathsf{negl}(s(m))$ since $\mathsf{poly}(s(m)) < \gamma(m)$ and

$\mathsf{poly}(s(m)) < c(m)$ for large enough m. Without loss of generality, we assume that there exists a function $f\colon \mathbb{N} \to \mathbb{N}$ such that $s(m^a) = O(s(m)^{f(a)})$ for any $a > 0$ by selecting $\gamma(m)$ and $c(m)$ as sufficiently small superpolynomials.

We define a function $k(n)$ as

$$k(n) := \max\left\{ m \in \mathbb{N} : s(m) \leq n^{\sqrt{\frac{\log s(n)}{\log n}}} \right\}.$$

Note that $s(k(n)+1) > n^{\sqrt{\frac{\log s(n)}{\log n}}}$ for each $n \in \mathbb{N}$. Since $\log s(n) = \omega(\log n)$, we have $s(k(n)+1) = n^{\omega(1)}$. We also observe that, for every $a > 0$,

$$\log s(k(n^a)) \leq \sqrt{\log s(n^a) \log n^a} \leq \sqrt{O(\log s(n) \log n)} = o(\log s(n)),$$

where we used $s(m^a) = O(s(m)^{f(a)})$ and $\log s(m) = \omega(\log n)$. Therefore, for every constant $\epsilon > 0$, we have $k(n) < n^\epsilon$ and $k(n)+1 \leq n^\epsilon$ for large enough n; otherwise, $k(n) \geq n^\epsilon$ for large enough n, and

$$\log s(k(n^{1/\epsilon})) \geq \log s(n),$$

which contradicts the above.

By Theorem 6.1, we conclude that NP has an interactive proof system of computational knowledge complexity $k(n)+1$ and negligible soundness error, and it holds that $k(n)+1 \leq n^\epsilon$ for every constant $\epsilon > 0$ and every large enough n. □

7 Non-Triviality with Nondeterminism and Robustly-Often Security

In this section, we characterize the following primitive, which is a natural intermediate notion between (standard) one-way functions and auxiliary-input one-way functions.

Definition 7.1 (Robustly-often one-way function). *A robustly-often* $\mathsf{P/poly}$*-computable one-way function is a polynomial-time-computable function family* $f = \{f_z \colon \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z \in Z}$*, where* $Z \subseteq \{0,1\}^*$ *is an infinite set, satisfying that for every non-uniform polynomial-time algorithm* A *and for every polynomial* p*, for all enough large* $z \in Z$*,*

$$\Pr_{w,A}\left[A(z, f_z(w)) \in f_z^{-1}(f_z(w))\right] < 1/p(|z|),$$

where w *is a uniform random seed for* f_z*.*

For this purpose, we introduce the classes $\mathsf{i.o.N{\cdot}CZK}$ and $\mathsf{i.o.CZK/poly}$.

Definition 7.2 (io-N·CZK). *The class* $\mathsf{i.o.N{\cdot}CZK}$ *is defined to be the set of promise problems* Π *satisfying the following property: There exists a promise problem* $\bar{\Pi} \in \mathsf{CZK}$ *and a polynomial* p *such that for infinitely many* $n \in \mathbb{N}$*, and for every* $x \in \{0,1\}^n$*,*

1. *(Completeness)* $x \in \Pi_{\mathsf{yes}} \Longrightarrow \exists z \in \{0,1\}^{p(n)}$ *s.t.* $(x,z) \in \bar{\Pi}_{\mathsf{yes}}$*.*
2. *(Soundness)* $x \in \Pi_{\mathsf{no}} \Longrightarrow \forall z \in \{0,1\}^{p(n)}$ $(x,z) \in \bar{\Pi}_{\mathsf{no}}$*.*

Definition 7.3 (io-CZK/poly)**.** *The class* $\mathsf{i.o.CZK/poly}$ *is defined to be the set of promise problems that have a computational zero-knowledge interactive proof system* (P, V) *in the same manner as* CZK *except that (i) the verifier* V *is a nonuniform polynomial-time algorithm (and the nonuniform advice is also given to the simulator), and (ii) the correctness, soundness, computational zero-knowledgeness holds simultaneously only for infinitely many lengths of an instance.*

The main theorem we show in this section is stated as follows.

Theorem 7.4. *For every complexity class* $\mathfrak{D}$ *satisfying* $\mathsf{MA} \subseteq \mathfrak{D}$ *and* $\mathsf{PSPACE} \not\subseteq \mathfrak{D}$*, the following are equivalent:*

1. *There exists a robustly-often* $\mathsf{P/poly}$*-computable one-way function.*
2. $\mathsf{PSPACE} \subseteq \mathsf{i.o.CZK/poly}$
3. $\mathsf{PSPACE} \subseteq \mathsf{i.o.N{\cdot}CZK}$
4. $\mathsf{i.o.CZK} \not\subseteq \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$
5. $\mathsf{i.o.N{\cdot}CZK} \not\subseteq \mathsf{i.o.}\mathfrak{D}$
6. $\mathsf{i.o.CZK} \not\subseteq \mathsf{i.o.NP/poly}$
7. $\mathsf{i.o.N{\cdot}CZK} \not\subseteq \mathsf{i.o.MA}$

Proof. Item 1 $\Longrightarrow$ Item 2 follows from the black-box construction of a commitment scheme that has computational hiding and statistical binding based on one-way functions [Nao91], and the construction of a zero-knowledge proof system for PSPACE problems [GMW91; BGGHKMR88]. More precisely, the nonuniform advice is used to specify the indices of the robustly-often $\mathsf{P/poly}$-computable one-way function.

The high-level idea to show Item 1 $\Longrightarrow$ Item 3 is the same as above, but we need some additional observations. Let $\Pi \in \mathsf{PSPACE}$ $(= \mathsf{IP})$, and let $Z \subseteq \{0,1\}^*$ be an index set for a robustly-often $\mathsf{P/poly}$-computable one-way function. Then we define a promise problem $\bar{\Pi} = (\bar{\Pi}_{\text{yes}}, \bar{\Pi}_{\text{no}})$ as follows:

$$\bar{\Pi}_{\text{yes}} := \{(x, z) : x \in \Pi_{\text{yes}} \text{ and } z \in Z\}$$
$$\bar{\Pi}_{\text{no}} := \{(x, z) : x \in \Pi_{\text{no}}\}.$$

Since $\Pi \in \mathsf{IP}$, the problem Π has a public coin interactive proof system (P, V). It is not hard to verify that (P, V) also recognizes $\bar{\Pi}$. We construct a zero-knowledge proof system for $\bar{\Pi}$ based on (P, V), as presented in [BGGHKMR88] (roughly speaking, the prover first simulates the original protocol (P, V) but sends the commitment of proper's messages, and then the prover proves in zero-knowledge that there is a transcript that V accepts and is consistent with the commitments at the first stage), where we use the second element z in the given instance as a candidate for an index of a robustly-often $\mathsf{P/poly}$-computable one-way function and employ the commitment scheme presented in [Nao91] base on the function indexed by z. For every $(x, z) \in \bar{\Pi}_{\text{yes}}$, since $z \in Z$, the modified prover demonstrates the protocol on a secure one-way function; thus, the computational zero-knowledgeness follows from that of the original protocol. The completeness is also follows from that of the original protocol and the syntax of the reveal phase of the commitment scheme. To see soundness, we observe that the soundness of the modified interactive proof system (based on

[GMW91; BGGHKMR88]) is derived only from the statistical biding of the commitment scheme. Moreover, the statistical biding property of the commitment scheme presented in [Nao91] does not depend on the one-wayness (and it follows only from the stretch of underlying generators). Thus, for every $(x, z) \in \bar{\Pi}_{\text{no}}$ (recall that z may not be contained in Z), the soundness follows from that of the original interactive proof and the statistical biding (that does not depend on z). Therefore, $\bar{\Pi} \in \mathsf{CZK}$ and $\Pi \in \mathsf{i.o.N{\cdot}CZK}$, where i.o. is due to the fact that Z is guaranteed only to be infinite.

Item 3 $\Longrightarrow$ Item 5 follows from $\mathsf{PSPACE} \not\subseteq \mathfrak{D}$. We also prove Item 2 $\Longrightarrow$ Item 4. By $\mathsf{PSPACE} \not\subseteq \mathfrak{D}$ and Item 2, we have $\mathsf{i.o.CZK/poly} \not\subseteq \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$. Thus, it suffices to observe that $\mathsf{i.o.CZK} \subseteq \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$ implies that $\mathsf{i.o.CZK/poly} \subseteq \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$. For every $\Pi \in \mathsf{i.o.CZK/poly}$, there exist another promise problem $\tilde{\Pi} \in \mathsf{CZK}$ and an infinite set $\mathcal{N} \subseteq \mathbb{N}$ such that for each $n \in \mathcal{N}$, there exists $z_n \in \{0,1\}^{\mathsf{poly}(n)}$ such that (i) $x \in \Pi_{\text{yes}} \cap \{0,1\}^n \Rightarrow (x, z_n) \in \tilde{\Pi}_{\text{yes}}$ and (ii) $x \in \Pi_{\text{no}} \cap \{0,1\}^n \Rightarrow (x, z_n) \in \tilde{\Pi}_{\text{no}}$. Let $\ell(n)$ be the length of the encoding for (x, z_n) for each $n \in \mathbb{N}$. We define a promise problem Γ as follows: for each $m \in \mathbb{N}$,

$$\Gamma \cap \{0,1\}^m = \begin{cases} (\tilde{\Pi}_{\text{yes}}, \tilde{\Pi}_{\text{no}}) & \text{if } m = \ell(n) \text{ for some } n \in \mathcal{N} \\ L_{hard} \cap \{0,1\}^m & \text{otherwise,} \end{cases}$$

where L_{hard} is an arbitrary language such that $L_{hard} \notin \mathsf{i.o.}\mathfrak{D}$. Since $\tilde{\Pi} \in \mathsf{CZK}$ and $\mathcal{N}$ is infinite, $\Gamma \in \mathsf{i.o.CZK}$. Thus, if $\mathsf{i.o.CZK} \subseteq \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$, we have $\Gamma \in \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$. Since $L_{hard} \notin \mathsf{i.o.}\mathfrak{D}$, the algorithm in $\mathfrak{D}\mathsf{/poly}$ must solve $\tilde{\Pi}$ for infinitely many $n \in \mathcal{N}$. Thus, we conclude that $\Pi \in \mathsf{i.o.}\mathfrak{D}\mathsf{/poly}$.

Both Item 4 $\Longrightarrow$ Item 6 and Item 5 $\Longrightarrow$ Item 7 follows from $\mathsf{MA} \subseteq \mathfrak{D}$. We prove Item 6 $\Longrightarrow$ Item 1 and Item 7 $\Longrightarrow$ Item 1 in Section 7.1. □

7.1 Extended Non-Triviality of ZK Implies Robustly Often One-Way Functions

In this subsection, we show Item 6 $\Longrightarrow$ Item 1 and Item 7 $\Longrightarrow$ Item 1.

Theorem 7.5. *If there is no robustly-often* $\mathsf{P/poly}$*-computable one-way function, then* $\mathsf{i.o.N{\cdot}CZK} = \mathsf{i.o.MA}$.

Theorem 7.6. *If there is no robustly-often* $\mathsf{P/poly}$*-computable one-way function, then* $\mathsf{i.o.CZK/poly} \subseteq \mathsf{i.o.NP/poly}$.

First, we introduce a slight modification of a robustly-often $\mathsf{P/poly}$-computable one-way function.

Definition 7.7 (Semi-robustly-often one-way)**.** *Let $Z \subseteq \{0,1\}^*$ be an infinite subset. A polynomial-time-computable function $f = \{f_z \colon \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z \in Z}$ is semi-robustly-often one-way if for every polynomials $\sigma(\cdot)$ and $p(\cdot)$, there exists an infinite subset $Z_{\sigma,p} \subseteq Z$ such that for every $z \in Z_{\sigma,p}$ and every circuit A of size at most $\sigma(|z|)$,*

$$\Pr_{A,w}\left[A(z, f_z(w)) \in f_z^{-1}(f_z(w))\right] < 1/p(|z|),$$

where w is a uniform random seed for f_z.

In fact, the existence of semi-robustly-often one-way is equivalent to that of a robustly-often $\mathsf{P/poly}$-computable one-way function.

Lemma 7.8. *A semi-robustly-often one-way function exists if and only if a robustly-often* $\mathsf{P/poly}$*-computable one-way function exists.*

Proof. It is not hard to verify that any robustly-often $\mathsf{P/poly}$-computable one-way function is also a semi-robustly-often one-way function by the definitions, where the infinite subset $Z_{\sigma,p}$ for the security of the semi-robustly-often one-way function is a subset obtained from Z by removing a finite number of small indices. Thus, we will only show the converse.

Let $f = \{f_z \colon \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z \in Z}$ be a secure semi-robustly-often one-way function defined on the infinite index set $Z \subseteq \{0,1\}^*$. From the security condition, for each $c \in \mathbb{N}$, there exists an infinite subset $Z_c \subseteq Z$ such that for every $z \in Z_c$ and every circuit A of size $|z|^c$,

$$\Pr_{A,w}\left[A(z, f_z(w)) \in f_z^{-1}(f_z(w))\right] \leq |z|^{-c}.$$

Without loss of generality, we assume that Z_c does not contain two strings of the same length.

We construct an infinitely-often one-way function. For each $n = \langle m, c\rangle$ (where $\langle , \rangle$ represents the standard one-to-one pairing function), we define $z'_n := z_{m,c} \circ 10^{n-m-1}$ if there exists $z_{m,c} \in Z_c$ such that $|z_{m,c}| = m$; otherwise, $z'_n = 0^n$. Let $Z' = \{z'_n : n \in \mathbb{N}\}$ and define $g = \{g_z \colon \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_Z$ as follows

$$g_{z'_n}(w) = \begin{cases} f_{z_{m,c}}(w) & \text{if } z'_n = z_{m,c} \circ 10^{n-m-1} \\ 0^n & \text{otherwise.} \end{cases}$$

Since $|z'_n| = n$ and f is polynomial-time computable, the function g is also polynomial-time computable.

We show the security of g. Let $A = \{A_n\}_{n \in \mathbb{N}}$ be a circuit family of size $\sigma(n)$, where σ is an arbitrary polynomial. In addition, let $p(n)$ be an arbitrary polynomial. We select a large enough constant c so that for all large enough n, it holds that $p(n) \leq n^c$ and $\sigma(n) \leq n^c$. By the property of Z_c, for every large enough $z \in Z_c$ and $n = \langle |z|, c\rangle$, the circuit A_n succeeds in inverting f_z only with probability at most $|z|^{-c} \leq 1/p(n)$. Since every index in $z \in Z_c$ is used in g on the security parameter $n = \langle |z|, c\rangle$, the circuit A fails to invert g on infinitely many indices corresponding to Z_c. □

The purpose of introducing semi-robustly-often one-way functions is to obtain the following lemma.

Lemma 7.9. *If there is no robustly-often* $\mathsf{P/poly}$*-computable one-way function, then for every polynomial-time auxiliary-input function* $f = \{f_z \colon \{0,1\}^{\mathsf{poly}(|z|)} \to \{0,1\}^{\mathsf{poly}(|z|)}\}_{z \in \{0,1\}^*}$*, every infinite set* $Z \subseteq \{0,1\}^*$*, and every polynomials* $p(\cdot)$ *and* $q(\cdot)$*, there exist a polynomial-time randomized algorithm* M*, a polynomial-time randomized oracle machine* R*, and a polynomial* $\sigma(\cdot)$ *such that for every* $z \in Z$ *and every circuit* I *of size* $\sigma(|z|)$ *such that if*

$$\Pr[M(I) = 1] \geq 1/q(|z|),$$

then

$$\Delta_{\mathsf{TV}}\left(R_z^I(f_z(\mathcal{U}_{\mathsf{poly}(|z|)})), \mathsf{UnifInv}_{f_z}(f_z(\mathcal{U}_{\mathsf{poly}(|z|)}))\right) \leq 1/p(|z|),$$

where $R_z(\cdot) := R(z, \cdot)$*.*

Moreover, for every $z \in Z$*, there exists a circuit* I_z *of size* $\sigma(|z|)$ *such that*

$$\Pr[M(I_z) = 1] \geq 1 - 1/q(|z|).$$

Proof. The polynomial-time randomized oracle machine R is the efficient black-box reduction from distributional inverting f to (standard) inverting $\tilde{f}$ [IL89], where $\tilde{f}$ is a polynomial-time auxiliary-input function on the same index set Z. The polynomial-time algorithm M is the polynomial-time randomized test based on the testability of inverting $\tilde{f}$, which is the same argument as the proof of Lemma 5.2. The first property follows from the testability of inverting and the correctness of the reduction R. Note that the first property holds for every polynomial σ.

We show the existence of I_z by contraposition. If for every polynomial σ, there exist infinitely many $z_1, z_2, \dots \in Z$ such that for every $i \in \mathbb{N}$ and every $\sigma(|z_i|)$-size circuit I, the circuit I fails to invert $\tilde{f}_{z_i}$ with probability at least $1 - 1/\mathsf{poly}(|z_i|)$, then $\tilde{f}$ is a semi-robustly-often variant of a weak one-way function, which implies the existence of a semi-robustly-often one-way function based on the proof of equivalence between a (standard) weak one-way function and a (strong) one-way function [Yao82]. By Lemma 7.8, there also exists a secure robustly-often $\mathsf{P/poly}$-computable one-way function, which contradicts the assumption of the non-existence. Thus, there exists a polynomial σ such that for every $z \in Z$, there exists a circuit I_z of size $\sigma(|z|)$ that inverts $\tilde{f}_z$ successfully. The randomized test M accepts such an I_z with high probability. □

We also use the following theorem presented in [Vad06].

Theorem 7.10 ([Vad06, Theorem 7.4])**.** *If there is no robustly-often* $\mathsf{P/poly}$*-computable one-way function, then* $\mathsf{HV\text{-}CZK} = \mathsf{HV\text{-}SZK}$*.*

Now, we prove Theorem 7.5.

Proof of Theorem 7.5. Fix $\Pi \in \mathsf{i.o.N{\cdot}CZK}$ arbitrarily. For contraposition, we assume the non-existence of robustly-often $\mathsf{P/poly}$-computable one-way functions and show that $\Pi \in \mathsf{i.o.MA}$.

Since $\Pi \in \mathsf{i.o.N{\cdot}CZK}$, there exist a promise problem $\bar{\Pi} \in \mathsf{CZK}$, a polynomial p, and an infinite subset $\mathcal{N} \subseteq \mathbb{N}$ such that for every $n \in \mathcal{N}$ and every $x \in \{0,1\}^n$, if $x \in \Pi_{\text{yes}}$, then $\exists z \in \{0,1\}^{p(n)}$ such that $(x,z) \in \bar{\Pi}_{\text{yes}}$; and if $x \in \Pi_{\text{no}}$, then $\forall z \in \{0,1\}^{p(n)}$ $(x,z) \in \bar{\Pi}_{\text{no}}$. Below, we fix $n \in \mathcal{N}$ arbitrarily.

Under the assumption that there is no robustly-often $\mathsf{P/poly}$-computable one-way functions, $\bar{\Pi} \in \mathsf{CZK} \subseteq \mathsf{HV\text{-}CZK} = \mathsf{HV\text{-}SZK}$ by Theorem 7.10. Let (P, V) denote the honest-verifier statistical zero-knowledge interactive proof system for $\bar{\Pi}$, where the completeness error $c(\cdot)$ and the soundness error $s(\cdot)$ satisfy $(1 - c(n)) - s(n) \geq 1/\gamma(n)$ for a polynomial $\gamma(n)$, and let S be the honest-verifier statistical simulator. Note that for every $x \in \Pi_{\text{yes}} \cap \{0,1\}^n$ and every $w \in \{0,1\}^{p(n)}$, the statistical difference between the verifier's view on the transaction $\langle P, V\rangle(x,z)$ and the outcome of $S(x,z)$ is at most $\mathsf{negl}(n)$.

We define a polynomial-time-computable auxiliary-input function $f = \{f_{x,z} \colon \{0,1\}^{\mathsf{poly}(|x|)} \to \{0,1\}^{\mathsf{poly}(|x|)}\}_{x \in \Pi_{\text{yes}} \cup \Pi_{\text{no}}, z \in \{0,1\}^{p(|x|)}}$ as, for each (x,z),

$$f_{x,z}(i, w) = (i, S(x, z; w)_{\leq i}),$$

where $i \sim [2\ell(|x|)]$, $w \sim \{0,1\}^{r_S(|x|)}$, and $S(x,z;w)_{\leq i}$ represents the first i messages in the transcript produced by $S(x,z;w)$.

We assume that $\bigcup_{n' \in \mathcal{N}} (\Pi_{\text{yes}} \cup \Pi_{\text{no}}) \cap \{0,1\}^{n'}$ is infinite: otherwise, Π is solvable in $\mathsf{i.o.P}$ (on $\mathcal{N}$). Thus, by Lemma 7.9, there exist a polynomial-time oracle machine R and a polynomial $\sigma(\cdot)$ such that for every $x \in (\Pi_{\text{yes}} \cup \Pi_{\text{no}}) \cap \{0,1\}^n$ and every $z \in \{0,1\}^{p(n)}$, there exists a randomized circuit $I_{x,z}$ of size $\sigma(n)$ such that

$$\Delta_{\mathsf{TV}}((f_{x,z}(i,w), R_{x,z}^{I_{x,z}}(x,z,f_{x,z}(i,w))), (f_{x,z}(i,w), \mathsf{UnifInv}_{f_{x,z}}(f_{x,z}(i,w))) \leq \frac{1}{2\ell(|x|)q(|x|)}, \tag{9}$$

where $R_{x,z}(\cdot) := R(x,z,\cdot)$, and q is a large enough polynomial we will specify later.

Now, we consider a randomized algorithm A that is given input $x \in \{0,1\}^*$ and attempts to determine $x \in \Pi_{\text{yes}}$ with a nondeterministic guess of (z, I) as follows, where $z \in \{0,1\}^{p(|x|)}$, and I is a description of $\sigma(|x|)$-size circuit: the algorithm A first checks whether the given I satisfies Eq. (9) by the test algorithm M in Lemma 7.9 with failure probability at most $1/(4\gamma(n))$. If I does not pass the test, A outputs 0. If I passes the test, A simulates the interaction with the simulator-based prover and the honest verifier on (x,z) by using I. More precisely, A constructs a simulator-based prover P_S from I as follows: On input (x,z), a given $2i$-th massage m_{2i} sent by a verifier, and the previous transaction $(m_1, \ldots, m_{2i-1})$, the simulator-based prover P_S sends back $S(x,z;R_{x,z}^{I_{x,z}}(x,z,2i,(m_1,\ldots,m_{2i-1},m_{2i})))_{=2i+1}$ as the $(2i+1)$-th message, where the notation $S(x,z;w)_{=i}$ represents the i-th message in the transcript produced by $S(x,z;w)$. Then, A executes $\langle P_S, V\rangle(x,z)$ and outputs 1 if V accepts at the end; otherwise, outputs 0.

Let $\mathcal{C}_\sigma \subseteq \{0,1\}^{\leq \mathsf{poly}(\sigma)}$ be a set of valid descriptions of all σ-size circuits. We show that for every large enough $x \in \{0,1\}^n$ (recall that $n \in \mathcal{N}$),

- if $x \in \Pi_{\text{yes}}$, then there exist $z \in \{0,1\}^{p(n)}$ and $I \in \mathcal{C}_{\sigma(n)}$ (in fact, $I = I_{x,z}$) such that
$$\Pr_A[A(x,z,I) = 1] \geq 1 - c(n) - \frac{1}{2\gamma(n)};$$
- if $x \in \Pi_{\text{no}}$, then for all $z \in \{0,1\}^{p(n)}$ and all $I \in \mathcal{C}_{\sigma(n)}$
$$\Pr_A[A(x,z,I) = 1] \leq 1 - c(n) - \gamma(n),$$

which implies $\Pi \in \mathsf{i.o.MA}$.

First, we consider the cases of $x \in \Pi_{\text{no}}$. In the case, for all $z \in \{0,1\}^{p(n)}$, $(x,z) \in \bar{\Pi}_{\text{no}}$. For each choice of $I \in \mathcal{C}_{\sigma(n)}$, there are two cases: (i) I does not pass the test, and (ii) I passes the test. In the former case, A outputs 0. By contrast, in the latter case, A constructs the simulation-based prover P_S based on I and outputs the decision of $\langle P_S, V\rangle(x,z)$. By the soundness property of (P,V), the verifier outputs 1 with probability at most $s(n) \leq 1 - c(n) - \gamma(n)$. Thus, in any case, $A(x,z,I)$ outputs 1 with probability at most $1 - c(n) - \gamma(n)$.

Next, we consider the cases of $x \in \Pi_{\text{yes}}$. In the case, there exists $z := z_x \in \{0,1\}^{p(n)}$ such that $(x, z_x) \in \bar{\Pi}_{\text{yes}}$. For $I := I_{x,z_x}$, there are two cases; (i) I does not pass the test in A, or (ii) I passes the test in A. Recall that the probability that case (i) occurs is at most $1/(4\gamma(n))$.

We consider the case (ii). Recall that I satisfies Eq. (9) in this case. Since the random input $i \sim [2\ell(n)]$ for $f_{x,z}$ is selected uniformly at random, it is not hard to verify that for all $i \in [2\ell(n)]$,

$$\Delta_{\mathsf{TV}}((S(x,z;w)_{\leq i}, R_{x,z}^{I}(x,z,i,S(x,z;w)_{\leq i})), (S(x,z;w)_{\leq i}, \mathsf{UnifInv}_{f_{x,z}}(f_{x,z}(i,w))) \leq \frac{1}{q(n)}. \tag{10}$$

Let P_S^* be the ideal simulation-based prover, which is that same as P_S except that P_S^* uses the ideal distributional inverting $\mathsf{UnifInv}_{f_{x,z}}$(-) instead of $R_{x,z}^I$. Let V_S^* be the ideal simulation-based verifier, which is a verifier that returns

$$S(x,z;\mathsf{UnifInv}_{f_{x,z}}(x,z,2(i-1)-1,(m_1,\ldots,m_{2(i-1)},m_{2i-1})))_{=2i}$$

as the $2i$-th message for an input (x, z), given $(2i-1)$-th massage m_{2i-1} from a prover, and the previous transaction $(m_1, \ldots, m_{2(i-1)})$. Moreover, we introduce the following inequality proved in [OV07] (where we used Eq. (10)).

Claim 7.11 ([proved in OV07, Proposition 3.11])**.**

$$\Delta_{\mathsf{TV}}(\langle P_S, V\rangle(x,z), S(x,z)) \leq \ell(n)\left(\frac{1}{q(n)} + 2\cdot\Delta_{\mathsf{TV}}(\langle P_S^*, V\rangle(x,z), S(x,z))\right).$$

Recall that S simulates the verifier's private coin, and let R be the distribution of the verifier's private coin obtained by $S(x,z) = \langle P_S^*, V_S^*\rangle(x,z)$. Then, $\Delta_{\mathsf{TV}}(R, \mathcal{U}_{r_S(n)}) = \mathsf{negl}(n)$ since S simulates $\mathsf{view}_V(\langle P, V\rangle(x,z))$ with a negligible statistical error. Since S always outputs valid transcripts, the statistical difference between the verifier's messages sent by V_S^* and V is at most $\mathsf{negl}(n)$ when they interact with P_S^*. Thus, we have $\Delta_{\mathsf{TV}}(\langle P_S^*, V\rangle(x,z), S(x,z)) = \Delta_{\mathsf{TV}}(\langle P_S^*, V\rangle(x,z), \langle P_S^*, V_S^*\rangle(x,z)) = \mathsf{negl}(n)$ and by Claim 7.11,

$$\Delta_{\mathsf{TV}}(\langle P_S, V\rangle(x,z), S(x,z)) \leq \frac{\ell(n)}{q(n)} + \mathsf{negl}(n).$$

This implies that

$$\begin{aligned}
\Pr[\langle P_S, V\rangle(x,z) = \text{“}\mathsf{accept}\text{”}] &\geq \Pr[S(x,z) \text{ is an accepting script}] - \frac{\ell(n)}{q(n)} - \mathsf{negl}(n)\\
&\geq \Pr[\langle P, V\rangle(x,z) = \text{“}\mathsf{accept}\text{”}] - \mathsf{negl}(n) - \frac{\ell(n)}{q(n)} - \mathsf{negl}(n)\\
&\geq 1 - c(n) - \frac{\ell(n)}{q(n)} - \mathsf{negl}(n),
\end{aligned}$$

where the second inequality follows from the property of the honest-verifier simulator, and the last inequality follows from the completeness of (P, V).

Now, we select a polynomial q sufficiently large so that

$$\frac{\ell(n)}{q(n)} + \mathsf{negl}(n) \leq \frac{1}{4\gamma(n)}.$$

Then, under the condition that I passes the test (case (ii)), A outputs 1 with probability at least $1 - c(n) - (1/4\gamma(n))$ Thus, by the union bound, we have

$$\Pr[A(x,z,I) = 1] \geq 1 - c(n) - \frac{1}{4\gamma(n)} - \frac{1}{4\gamma(n)} = 1 - c(n) - \frac{1}{2\gamma(n)}.$$

□

We also present the proof of Theorem 7.6, which is almost the same as that of Theorem 7.5; so we only highlight the differences.

Proof of Theorem 7.6. For contraposition, we assume the non-existence of robustly-often $\mathsf{P/poly}$-computable one-way functions and derive $\mathsf{i.o.CZK/poly} \subseteq \mathsf{i.o.MA/poly}$ ($= \mathsf{i.o.NP/poly}$ [cf. FF93]). For this, we fix $\Pi \in \mathsf{i.o.N{\cdot}CZK}$ arbitrarily and will show that $\Pi \in \mathsf{i.o.MA/poly}$.

The construction of the function family $\{f_{x,z}\}$ and the nondeterministic algorithm A for solving Π is the same as that of Theorem 7.5 except that z is now nonuniform advice for the verifier and is passed to A as nonuniform advice (namely, A uses its nondeterminism only on the inverter I). By the same argument as Theorem 7.5, we can show that A solves Π for infinitely many instance sizes. Thus, we conclude that $\Pi \in \mathsf{i.o.MA/poly}$. □

8 NP hardness of Meta-Complexity and Zero-Knowledge

In this section, we show that the NP-hardness of the meta-computational problem called $\mathsf{GapMdKP}$ via a BPP-reduction (with plausible properties) yields $\mathsf{NP} \in \mathsf{SZKA}$, which simplifies Hirahara's reduction presented in [Hir23].

First, we introduce the problem $\mathsf{GapMdKP}$.

Definition 8.1 (Distributional Kolmogorov complexity)**.** *For a string $x \in \{0,1\}^*$, a time bound $t \in \mathbb{N}$, and a parameter $\lambda \in (0,1]$, a randomized oracle A, and a distribution over $\mathcal{D}$ over $\{0,1\}^*$, the A-oracle t-time-bounded distributional Kolmogorov complexity of x given $\mathcal{D}$ is defined as*

$$\mathsf{dK}^{t,A}_{\lambda}(x|\mathcal{D}) := \min\left\{p \in \mathbb{N} \mid \exists \Pi \in \{0,1\}^p \text{ such that } \Pr_{A, y\sim\mathcal{D}}\left[U^A(\Pi, y) \text{ outputs } x \text{ in time } t\right] \geq \lambda\right\}.$$

For a function $\tau \colon \mathbb{N} \to \mathbb{N}$, we define

$$\mathsf{dK}^{\tau,A}_{\lambda}(x|\mathcal{D}) := \mathsf{dK}^{\tau(n+m),A}_{\lambda}(x|\mathcal{D}),$$

where $n = |x|$ and $m = \max\{|y| : y \in \mathrm{supp}(\mathcal{D})\}$. We omit the superscript A if $A = \emptyset$.

Definition 8.2 (GapMdKP)**.** *For a polynomial $\tau \colon \mathbb{N} \to \mathbb{N}$, a constant $\epsilon > 0$, and an oracle A, we define $\mathsf{Gap}_{\tau,\epsilon}\mathsf{MdKP}^A = (\Pi^A_{\mathrm{yes}}, \Pi^A_{\mathrm{no}})$ as*

$$\Pi^A_{\mathrm{yes}} := \left\{(x, \mathcal{D}, 1^s) \mid \mathsf{dK}^{\tau,A}_{1-\mathsf{negl}(|x|)}(x|\mathcal{D}) \leq s\right\}$$
$$\Pi^A_{\mathrm{no}} := \left\{(x, \mathcal{D}, 1^s) \mid \mathsf{dK}^{\tau,A}_{1-1/|x|}(x|\mathcal{D}) > (1+\epsilon)s\right\}.$$

In this work, we fix the parameter $\epsilon > 0$ arbitrarily and omit the subscript ϵ from $\mathsf{Gap}_{\tau,\epsilon}\mathsf{MdKP}^A$.

For a promise problem Π and a set of promise problems $\mathcal{S}$, we use the expression that $\Pi \leq_m \mathcal{S}$ via BPP-reduction whose error probability is negligible to refer to the existence of a many-one reduction from Π to $\mathcal{S}$ in the sense that there exists a polynomial-time randomized algorithm R such that for every $x \in \Pi_{\mathrm{yes}}$ (resp. $x \in \Pi_{\mathrm{no}}$) and for every $\Gamma \in \mathcal{S}$, it holds that $R(x) \in \Gamma_{\mathrm{yes}}$ (resp. $R(x) \in \Gamma_{\mathrm{no}}$) with probability $1 - \mathsf{negl}(|x|)$ over the choice of the randomness for R. Particularly we use the notation $\mathsf{NP} \leq_m \mathcal{S}$ to refer to the statement that $\Pi \leq_m \mathcal{S}$ for all $\Pi \in \mathsf{NP}$.

Previously, Hirahara [Hir23] proved the equivalence between the NP-hardness of GapMdKP and the existence of one-way functions, one of which is stated as follows.

Theorem 8.3 (See [Hir23, Section 6])**.** *If a one-way function secure against $\mathsf{P/poly}$ exists, then $\mathsf{NP} \leq_m \{\mathsf{Gap}_{\tau}\mathsf{MdKP}^A : \tau \in \mathsf{poly}$ and $A \in \mathsf{P/poly}\}$ via a nonadaptive parametric-honest BPP-reduction whose error probability is negligible over the randomness for the reduction.*

We show that every NP-hard problem Π reducible GapMdKP$^{\mathsf{P/poly}}$ via a nonadaptive BPP-reduction (with negligible error probability) indeed admits a statistically zero-knowledge interactive argument system.

Theorem 8.4. *If an NP-hard problem Π satisfies $\Pi \leq_m \{\mathsf{Gap}_{\tau}\mathsf{MdKP}^A : \tau \in \mathsf{poly}$ and $A \in \mathsf{P/poly}\}$ via a nonadaptive parametric-honest BPP-reduction whose error probability is negligible over the randomness for the reduction, then $\Pi \in \mathsf{HV\text{-}SZKA}$.*

As a corollary, we obtain the converse of Theorem 8.3, which simplifies the original proof presented in [Hir23].

Corollary 8.5. *If* $\mathsf{NP} \leq_m \{\mathsf{Gap}_\tau\mathsf{MdKP}^A : \tau \in \mathsf{poly}$ *and* $A \in \mathsf{P/poly}\}$ *via a nonadaptive* BPP-*reduction satisfying the conditions of Theorem 8.4, then* $\mathsf{NP} \notin \mathsf{i.o.P/poly}$ *implies the existence of one-way functions secure against* P/poly.

Proof. Since $\mathsf{SAT} \leq_m \{\mathsf{Gap}_\tau\mathsf{MdKP}^A : \tau \in \mathsf{poly}$ and $A \in \mathsf{P/poly}\}$ via a nonadaptive BPP-reduction satisfying the conditions of Theorem 8.4, it holds that $\mathsf{SAT} \in \mathsf{HV\text{-}SZKA}$. Since HV-SZKA is closed by many-one reductions, $\mathsf{NP} \subseteq \mathsf{HV\text{-}SZKA}$. Thus, by Theorems 2.1 and 4.3, $\mathsf{NP} \notin \mathsf{i.o.P/poly}$ implies the existence of one-way functions secure against P/poly. □

Now, we formally prove Theorem 8.4, where we use the following direct product generator DP and its property.

Theorem 8.6 ([Hir23, Theorem 7.6 and Lemma 8.15])**.** *For any* $k \in \mathbb{N}$ *and any* $x, z \in \{0,1\}^*$ *with* $|z| = k|x|$, *let* $\mathsf{DP}_k(x; z) := z \circ \langle x, z^1\rangle_{\mathbb{F}_2} \circ \cdots \circ \langle x, z^k\rangle_{\mathbb{F}_2} \in \{0,1\}^{|z|+k}$, *where* $z = z^1 \circ \cdots \circ z^k$, $|z^i| = |x|$ *for each* i, *and* $\langle , \rangle_{\mathbb{F}_2}$ *represents the inner product in* $\mathbb{F}_2$.

There exists $B \in \mathsf{P/poly}$ *such that for every* $\epsilon \in (0,1]$, *every* $k \in \mathbb{N}$, *every* $x \sim \{0,1\}^*$, *every distribution* $\mathcal{D}$, *and every distinguisher* $D \in \mathsf{P/poly}$ *satisfying*

$$\Pr_{z\sim\{0,1\}^{k|x|}, y\sim\mathcal{D}}[D(\mathsf{DP}_k(x;z), y) = 1] - \Pr_{w\sim\{0,1\}^{k|x|+k}}[D(w,y) = 1] \geq \epsilon,$$

it holds that

$$\mathsf{dK}^{\mathsf{poly}(|x|/\epsilon), B, D}(x|\mathcal{D}) \leq k + O(\log(|x|/\epsilon)).$$

Proof of Theorem 8.4. Let τ be an arbitrary polynomial. Let $\sigma(n) = \omega(\log n)$ be a polynomial-time-computable function satisfying $\sigma(n) \leq 2n$, and let R be the nonadaptive parametric-honest BPP-reduction from a promise problem Π to $\{\mathsf{Gap}_{\tau,\sigma}\mathsf{MdKP}^A : \tau \in \mathsf{poly}$ and $A \in \mathsf{P/poly}\}$ in the assumption. Let $\xi > 0$ be the constant such that $R(z)$ always outputs $(x, \mathcal{D}, 1^s)$ with $s \geq |z|^\xi$. In addition, we can assume that $\xi \geq 2$ when Π is NP-hard [cf. Hir23, Proposition 11.2].

We construct an honest-verifier statistical zero-knowledge argument system (P, V) for Π as follows:

Common input: $z \in \{0,1\}^*$.

Verifier 1: V selects (x, y, s, w) according to $(x, \mathcal{D}, 1^s) \sim R(z)$, $y \sim \mathcal{D}$, and $w \sim \{0,1\}^{|x|(s+\sigma(|z|^\xi)/2)}$. Next, V selects a secret bit $b \sim \{0,1\}$. If $b = 0$, then V sends (y, w, w'), where $w' \sim \{0,1\}^{s+\sigma(|z|^\xi)/2}$, to the prover. If $b = 1$, then V computes $w \circ w' = \mathsf{DP}_{s+\sigma(|z|^\xi)/2}(x; w)$ (where $|w'| = s + \sigma(|z|^\xi)/2$) and sends (y, w, w', s) to the prover.

Prover 1: P is given (y, w, w', s), where $y, w, w' \in \{0,1\}^*$ and then examines whether $\mathrm{K}^{\tau'}(w'|y, w, z, s) \leq s + \sigma(|z|^\xi)/4$ by exhaustive search (recall that s is contained in the conditional string y by the assumption), where τ' is a large enough polynomial. If so, P sends 1 to the verifier; otherwise, P sends 0.

Verifier 2: V is given $b' \in \{0,1\}$ and then checks whether $b = b'$. If so, V outputs "accept"; otherwise, V outputs "reject".

It is easily verified that V's messages are computable in polynomial time in the input size since the size of the distribution $\mathcal{D}$ (encoded as a circuit) and s are bounded by the running time of the polynomial-time reduction R, and $\sigma(n)(\leq 2n)$ is polynomial-time-computable.

For the honest-verifier V above, we construct a statistical simulator S as follows:

Common input: $z \in \{0,1\}^*$.

Simulator: S executes V's first step and obtains the first message m and the internal randomness r for V used for generating m. Note that the internal secret bit $b \in \{0,1\}$ is contained in r. Then S produces the following transcript with the verifier's randomness r:

$$m, b, \text{“}\mathsf{accept}\text{”}.$$

We claim the completeness, soundness, and zero-knowledgeness as follows.

Claim 8.7 (Completeness)**.** *For every large enough polynomial polynomial τ' and for every $z \in \Pi_{\mathrm{yes}}$,*

$$\Pr_V[\langle P, V\rangle(z) = \text{“}\mathsf{accept}\text{”}] \geq 1 - \mathsf{negl}(|z|).$$

Claim 8.8 (Soundness)**.** *For every nonuniform polynomial-time prover P', for all but finite $z \in \Pi_{\mathrm{no}}$,*

$$\Pr_V[\langle P', V\rangle(z) = \text{“}\mathsf{accept}\text{”}] \leq 1 - 1/\mathsf{poly}(|z|).$$

Claim 8.9 (Zero-knowledgeness)**.** *For every large enough polynomial τ' and for every $z \in \Pi_{\mathrm{yes}}$,*

$$\Delta_{\mathsf{TV}}(S(z), \mathsf{view}_V(\langle P, V\rangle(z))) = \mathsf{negl}(|z|).$$

Proof of Claim 8.7. For the completeness, we verify the event $b' \neq b$ occurs only with negligible probability when $z \in \Pi_{\mathrm{yes}}$ is given. Since z is a yes instance, $R(z)$ produces $(x, \mathcal{D}, 1^s)$ such that $\mathsf{dK}^{\tau'/2}_{1-\mathsf{negl}(|x|)}(x|\mathcal{D}) \leq s$ with probability at least $1 - \mathsf{negl}(|z|)$. Thus, (x, y, s, w) selected by V (at the first round) satisfies that $\mathrm{K}^{\tau}(x|y) \leq s$ with probability at least $1 - \mathsf{negl}(|z|)$. Below we consider only this case.

We show the correctness by case analysis on the choice of the secret bit b.

(i) The case of $b = 0$. In this case, recall that V selects w' uniformly at random and sends it. Thus, the probability that $\mathrm{K}(w'|y, w, z, s) \leq s + \sigma(|z|^{\xi})/4 = |w'| - \sigma(|z|^{\xi})/4$ is at most $2^{-(\sigma(|z|^{\xi})/4)+1} = \mathsf{negl}(|z|)$ since $\sigma(n) = \omega(\log n)$. Thus, P returns $b' = 1$ ($\neq b$) with probability at most $\mathsf{negl}(n)$.

(ii) The case of $b = 1$. In this case, V selects $w \circ w' = \mathsf{DP}_{s+\sigma(|z|^{\xi})/2}(x; w)$ and sends it. Thus, we can observe that

$$\mathrm{K}^{\tau'}(w'|y, w, z, s) \leq \mathrm{K}^{\tau'/2}(x|y) + O(1) \leq s + O(1) < s + \sigma(|z|^{\xi})/4.$$

Thus, P returns $b' = 1$ $(= b)$ with probability at least $1 - \mathsf{negl}(|z|)$. ◇

Proof of Claim 8.8. For the soundness, we show that the distribution of w produced by the verifier when $b = 1$ is computationally indistinguishable from the uniform distribution (i.e., the distribution of w produced by the verifier when $b = 0$). Below we consider the case where $b = 1$.

Since z is a no instance, for every polynomial τ and every $A \in \mathsf{P/poly}$, the reduction $R(z)$ produces $(x, \mathcal{D}, 1^s)$ such that $\mathsf{dK}^{\tau,A}_{1-1/|x|}(x|\mathcal{D}) > (1+\epsilon)s \geq s + \epsilon|z|^2$ with probability at least $1 -$

$\mathsf{negl}(|z|)$. Under this condition, we observe that $w \circ w' = \mathsf{DP}_{s+\sigma(|x|)/2}(x; w)$ (given $y \sim \mathcal{D}$) is computationally indistinguishable from a uniformly random string (for nonuniform polynomial-time distinguishers); otherwise, by the reconstruction property (Theorem 8.6), we have

$$\mathsf{dK}^{\tau',B,D}(x|\mathcal{D}) \leq \mathsf{dK}^{\tau',B,D}(x|\mathcal{D}, z, s) + O(|z|) \leq s + \sigma(|x|)/2 + O(\log |z|) + O(|z|) < s + \epsilon|z|^2$$

for a sufficiently large polynomial τ', a $\mathsf{P/poly}$-computable distinguisher D, and $B \in \mathsf{P/poly}$ in Theorem 8.6. ◇

Proof of Claim 8.9. The proof is based on that of Claim 8.7, where we proved that, for every $z \in \Pi_{\mathrm{yes}}$, the prover's message b' is equal to the verifier's random bit b with probability at least $1 - \mathsf{negl}(n)$. Since the simulator S follows the protocol except that S outputs b instead of b', the produced view is statistically equivalent to that of the actual interaction only with negligible statistical error. ◇

□

Acknowledgements

We thank anonymous reviewers for helpful comments. Shuichi Hirahara was supported by JST, PRESTO Grant Number JPMJPR2024, Japan.

References

[Adl78] Leonard M. Adleman. “Two Theorems on Random Polynomial Time”. In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1978, pp. 75–83. DOI: 10.1109/SFCS.1978.37.

[AH19] Eric Allender and Shuichi Hirahara. “New Insights on the (Non-)Hardness of Circuit Minimization and Related Problems”. In: *TOCT* 11.4 (2019), 27:1–27:27. DOI: 10.1145/3349616.

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

[BDRV18] Itay Berman, Akshay Degwekar, Ron D. Rothblum, and Prashant Nalini Vasudevan. “From Laconic Zero-Knowledge to Public-Key Cryptography - Extended Abstract”. In: *Proceedings of the International Cryptology Conference (CRYPTO)*. 2018, pp. 674–697. DOI: 10.1007/978-3-319-96878-0_23.

[Bel02] Mihir Bellare. “A Note on Negligible Functions”. In: *J. Cryptol.* 15.4 (2002), pp. 271–284. DOI: 10.1007/s00145-002-0116-x.

[BFKL93] Avrim Blum, Merrick L. Furst, Michael J. Kearns, and Richard J. Lipton. “Cryptographic Primitives Based on Hard Learning Problems”. In: *Proceedings of the International Cryptology Conference (CRYPTO)*. 1993, pp. 278–291. DOI: 10.1007/3-540-48329-2_24.

[BGGHKMR88] Michael Ben-Or, Oded Goldreich, Shafi Goldwasser, Johan Håstad, Joe Kilian, Silvio Micali, and Phillip Rogaway. "Everything Provable is Provable in Zero-Knowledge". In: *Proceedings of the International Cryptology Conference (CRYPTO)*. 1988, pp. 37–56. DOI: 10.1007/0-387-34799-2_4.

[DH76] Whitfield Diffie and Martin E. Hellman. "New directions in cryptography". In: *IEEE Trans. Information Theory* 22.6 (1976), pp. 644–654. DOI: 10.1109/TIT.1976.1055638.

[FF93] Joan Feigenbaum and Lance Fortnow. "Random-Self-Reducibility of Complete Sets". In: *SIAM J. Comput.* 22.5 (1993), pp. 994–1005. DOI: 10.1137/0222061.

[For89] Lance Fortnow. "The Complexity of Perfect Zero-Knowledge". In: *Advances in Computing Research* 5 (1989), pp. 327–343.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. "How to construct random functions". In: *J. ACM* 33.4 (1986), pp. 792–807. DOI: 10.1145/6490.6503.

[GH98] Oded Goldreich and Johan Håstad. "On the Complexity of Interactive Proofs with Bounded Communication". In: *Inf. Process. Lett.* 67.4 (1998), pp. 205–214. DOI: 10.1016/S0020-0190(98)00116-1.

[GM84] Shafi Goldwasser and Silvio Micali. "Probabilistic Encryption". In: *J. Comput. Syst. Sci.* 28.2 (1984), pp. 270–299. DOI: 10.1016/0022-0000(84)90070-9.

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. "The Knowledge Complexity of Interactive Proof Systems". In: *SIAM J. Comput.* 18.1 (1989), pp. 186–208. DOI: 10.1137/0218012.

[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". In: *J. ACM* 38.3 (1991), pp. 691–729. DOI: 10.1145/116825.116852.

[GP99] Oded Goldreich and Erez Petrank. "Quantifying Knowledge Complexity". In: *Comput. Complex.* 8.1 (1999), pp. 50–98. DOI: 10.1007/S000370050019.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. "A Pseudorandom Generator from any One-way Function". In: *SIAM J. Comput.* 28.4 (1999), pp. 1364–1396. DOI: 10.1137/S0097539793244708.

[Hir18] Shuichi Hirahara. "Non-Black-Box Worst-Case to Average-Case Reductions within NP". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2018, pp. 247–258. DOI: 10.1109/FOCS.2018.00032.

[Hir22] Shuichi Hirahara. "NP-Hardness of Learning Programs and Partial MCSP". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2022, pp. 968–979. DOI: 10.1109/FOCS54457.2022.00095.

[Hir23] Shuichi Hirahara. "Capturing One-Way Functions via NP-Hardness of Meta-Complexity". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2023, pp. 1027–1038. DOI: 10.1145/3564246.3585130.

[HN23] Shuichi Hirahara and Mikito Nanashima. "Learning in Pessiland via Inductive Inference". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2023, pp. 447–457. DOI: 10.1109/FOCS57990.2023.00033.

[HS17] Shuichi Hirahara and Rahul Santhanam. "On the Average-Case Complexity of MCSP and Its Variants". In: *Proceedings of the Computational Complexity Conference (CCC)*. 2017, 7:1–7:20. DOI: 10.4230/LIPIcs.CCC.2017.7.

[HS22] Shuichi Hirahara and Rahul Santhanam. "Errorless versus Error-prone Average-case Complexity". In: *Proceedings of the Innovations in Theoretical Computer Science Conference (ITCS)*. 2022, 38:1–38:23.

[IL89] Russell Impagliazzo and Michael Luby. "One-way Functions are Essential for Complexity Based Cryptography (Extended Abstract)". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1989, pp. 230–235. DOI: 10.1109/SFCS.1989.63483.

[IL90] Russell Impagliazzo and Leonid A. Levin. "No Better Ways to Generate Hard NP Instances than Picking Uniformly at Random". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1990, pp. 812–821. DOI: 10.1109/FSCS.1990.89604.

[Ila23] Rahul Ilango. "SAT Reduces to the Minimum Circuit Size Problem with a Random Oracle". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2023, pp. 733–742. DOI: 10.1109/FOCS57990.2023.00048.

[Imp95] Russell Impagliazzo. "A Personal View of Average-Case Complexity". In: *Proceedings of the Structure in Complexity Theory Conference*. 1995, pp. 134–147. DOI: 10.1109/SCT.1995.514853.

[IRS22] Rahul Ilango, Hanlin Ren, and Rahul Santhanam. "Robustness of average-case meta-complexity via pseudorandomness". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 2022, pp. 1575–1583. DOI: 10.1145/3519935.3520051.

[KMNPRY14] Ilan Komargodski, Tal Moran, Moni Naor, Rafael Pass, Alon Rosen, and Eylon Yogev. "One-Way Functions and (Im)Perfect Obfuscation". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2014, pp. 374–383. DOI: 10.1109/FOCS.2014.47.

[LP20] Yanyi Liu and Rafael Pass. "On One-way Functions and Kolmogorov Complexity". In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2020, pp. 1243–1254. DOI: 10.1109/FOCS46700.2020.00118.

[LP23] Yanyi Liu and Rafael Pass. "On One-way Functions and the Worst-case Hardness of Time-Bounded Kolmogorov Complexity". In: *Electron. Colloquium Comput. Complex.* TR23-103 (2023). ECCC: TR23-103.

[LY94] Richard J. Lipton and Neal E. Young. "Simple strategies for large zero-sum games with applications to complexity theory". In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 1994, pp. 734–740. DOI: 10.1145/195058.195447.

[Nan21] Mikito Nanashima. "On Basing Auxiliary-Input Cryptography on NP-Hardness via Nonadaptive Black-Box Reductions". In: *Proceedings of the Innovations in Theoretical Computer Science Conference (ITCS)*. 2021, 29:1–29:15. DOI: 10.4230/LIPIcs.ITCS.2021.29.

[Nao91] Moni Naor. “Bit Commitment Using Pseudorandomness”. In: *J. Cryptol.* 4.2 (1991), pp. 151–158. DOI: 10.1007/BF00196774.

[NOV06] Minh-Huyen Nguyen, Shien Jin Ong, and Salil P. Vadhan. “Statistical Zero-Knowledge Arguments for NP from Any One-Way Function”. In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 2006, pp. 3–14. DOI: 10.1109/FOCS.2006.71.

[NR06] Moni Naor and Guy N. Rothblum. “Learning to impersonate”. In: *Proceedings of the International Conference on Machine Learning (ICML)*. 2006, pp. 649–656. DOI: 10.1145/1143844.1143926.

[Ost91] Rafail Ostrovsky. “One-Way Functions, Hard on Average Problems, and Statistical Zero-Knowledge Proofs”. In: *Proceedings of the Structure in Complexity Theory Conference*. 1991, pp. 133–138. DOI: 10.1109/SCT.1991.160253.

[OV07] Shien Jin Ong and Salil P. Vadhan. “Zero Knowledge and Soundness Are Symmetric”. In: *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*. 2007, pp. 187–209. DOI: 10.1007/978-3-540-72540-4_11.

[OW93] Rafail Ostrovsky and Avi Wigderson. “One-Way Fuctions are Essential for Non-Trivial Zero-Knowledge”. In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 1993, pp. 3–17. DOI: 10.1109/ISTCS.1993.253489.

[PT02] Erez Petrank and Gábor Tardos. “On the Knowledge Complexity of *NP*”. In: *Comb.* 22.1 (2002), pp. 83–121. DOI: 10.1007/s004930200005.

[Rom90] John Rompel. “One-Way Functions are Necessary and Sufficient for Secure Signatures”. In: *Proceedings of the Symposium on Theory of Computing (STOC)*. 1990, pp. 387–394. DOI: 10.1145/100216.100269.

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

[Vad06] Salil P. Vadhan. “An Unconditional Study of Computational Zero Knowledge”. In: *SIAM J. Comput.* 36.4 (2006), pp. 1160–1214. DOI: 10.1137/S0097539705447207.

[Yao82] Andrew Chi-Chih Yao. “Theory and Applications of Trapdoor Functions (Extended Abstract)”. In: *Proceedings of the Symposium on Foundations of Computer Science (FOCS)*. 1982, pp. 80–91. DOI: 10.1109/SFCS.1982.45.