Can Statistical Zero Knowledge be made Non-Interactive? or On the Relationship of $\mathcal{SZK}$ and $\mathcal{NISZK}$

Oded Goldreich* Amit Sahai† Salil Vadhan‡

October 7, 1998

Abstract

We further extend the study, recently initiated by De-Santis *et. al.* (ICALP98) of non-interactive statistical zero-knowledge proofs. Our main focus is to compare the class $\mathcal{NISZK}$ of problems possessing such *non-interactive* proofs to the class $\mathcal{SZK}$ of problems possessing *interactive* statistical zero-knowledge proofs. Along these lines, we first show that if statistical zero-knowledge is non-trivial then so is non-interactive statistical zero-knowledge, where by non-trivial we mean that the class includes problems which are *not* solvable in probabilistic polynomial-time. (The hypothesis holds under various assumptions, such as the intractability of the Discrete Logarithm Problem.) Furthermore, we show that if $\mathcal{NISZK}$ is closed under complementation, then in fact $\mathcal{SZK} = \mathcal{NISZK}$, i.e. all statistical zero-knowledge proofs can be made non-interactive.

The main tools in our analysis are two promise problems that are natural restrictions of promise problems known to be complete for $\mathcal{SZK}$. We show that these restricted problems are in fact complete for $\mathcal{NISZK}$, and using this relationship we derive our results comparing the two classes. The two problems refer to the statistical difference, and difference in entropy, respectively, of a given distribution from the uniform one. We also consider a weak form of $\mathcal{NISZK}$, in which only requires that for every inverse polynomial $1/p(n)$, there exists a simulator which achieves simulator deviation $1/p(n)$, and show that this weak form of $\mathcal{NISZK}$ actually equals $\mathcal{NISZK}$.

Keywords: Complexity and Cryptography.

*Department of Computer Science, Weizmann Institute of Science, Rehovot, ISRAEL. E-mail: `oded@wisdom.weizmann.ac.il`. Work done while visiting LCS, MIT. Supported by DARPA grant DABT63-96-C-0018.

†Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA02139. E-mail: `amits@theory.lcs.mit.edu`. Supported by DOD/NDSEG fellowship and DARPA grant DABT63-96-C-0018.

‡Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA02139. E-mail: `salil@math.mit.edu`. Supported by DOD/NDSEG fellowship and in part by DARPA grant DABT63-96-C-0018.

1 Introduction

Zero-Knowledge proofs, introduced by Goldwasser, Micali and Rackoff [23], are fascinating and extremely useful constructs. Their fascinating nature is due to their seemingly contradictory nature; they are both convincing and yet yield nothing beyond the validity of the assertion being proven. Their applicability in the domain of cryptography is vast; they are typically used to force malicious parties to behave according to a predetermined protocol (which requires parties to provide proofs of the correctness of their secret-based actions without revealing these secrets). Zero-knowledge proofs come in many flavors, and in this paper we focus on two parameters: The first parameter is the underlying *communication model*, and the second is the *type of the zero-knowledge guarantee*.

The communication model. When Goldwasser, Micali, and Rackoff proposed the definition of zero-knowledge proofs, it seemed that interaction was crucial to achieving zero-knowledge – that the possibility of zero-knowledge arose through the power of interaction. Indeed, it was not unexpected when [19] showed zero-knowledge to be trivial (i.e., only exists for proofs of $\mathcal{BPP}$ statements) in the most straightforward non-interactive models. Surprisingly, however, Blum, Feldman, and Micali [5], showed that by changing the model slightly, it is possible to achieve zero-knowledge in a non-interactive setting (i.e. where only unidirectional communication can occur). Specifically, they assume that both Prover and Verifier have access to a shared truly random string, called the *reference string*. Aside from this assumption, all communication consists of one message, the "proof," which is generated by the Prover (based on the assertion being proved and the reference string) and sent from the Prover to the Verifier.

Non-interactive zero-knowledge proofs, on top of being more communication-efficient by definition, have several applications not offered by orndinary interactive zero-knowledge proofs. They have been used, among other things, to build digital signature schemes secure against adaptive chosen message attack [3], public-key cryptosystems secure against chosen-ciphertext attack [28, 13], and non-malleable cryptosystems [13].

The zero-knowledge guarantee. For ordinary *interactive* zero-knowledge proofs, the zero-knoweldege requirement is formulated by saying that the transcript of the Verifier's interaction with the Prover can be *simulated* by the Verifier itself. Similarly, for the *non-interactive* setting described above, the zero-knowledge condition is formulated by requiring that one can produce, knowing only the statement of the assertion, a random reference string *along with* a "proof" that works for the reference string. More precisely, we require that there exists an efficient procedure that on input a valid assertion produces a distribution which is "similar" to the joint distribution of random reference strings and proofs generated by the Prover. The key parameter is the interpretation of "similarity." Two notions have been commonly considered in the literature (cf., [23, 18, 16, 6, 4]). *Statistical zero-knowledge* requires that these distributions be statistically close (i.e., the statistical difference between them is negligible). *Computational zero-knowledge* instead requires that these distributions are computationally indistinguishable (cf., [22, 35]). In this work, we focus on the stronger security requirement of statistical zero-knowledge.

Until recently, most work on non-interactive zero-knowledge has focused on the computational type (cf., [5, 6, 15, 25]). The study of non-interactive *statistical* zero-knowledge has been recently initiated by De-Santis *et. al.* [11].[1] Their main result is the existence of a complete promise problem for the class of problems possessing non-interactive statistical zero-knowledge proofs (hereafter denoted $\mathcal{NISZK}$). This was similar to the work of [32], where a complete promise problem was given for the class of problems possessing *interactive* statistical zero-knowledge proofs (denoted $\mathcal{SZK}$).

[1] Actually, [6] did define non-interactive *perfect* zero-knowledge proofs (which is a slightly stricter notion that statistical zero-knowledge) and prove that a variant of `Quadratic Residuosity` has such proofs, and non-interactive statistical zero-knowledge was considered in [4], but later works all focused on the computational version.

Our Contribution.

In this work, we seek to understand what, if any, additional power interaction gives in the context of statistical zero-knowledge. Thus, we continue the investigation of $\mathcal{NISZK}$, focusing on the relationship between the interactive and non-interactive variants of statistical zero-knowledge. Our first result is that the non-triviality of $\mathcal{SZK}$ implies non-triviality of $\mathcal{NISZK}$, where by non-trivial we mean that a class includes problems which are *not* solvable in probabilistic polynomial-time. The hypothesis holds under various assumptions, such as the intractability of Discrete Logarithm Problem [17] (or Quadratic Residuosity [23] or Graph Isomorphism [18]), but variants of these last two problems are already known to be in $\mathcal{NISZK}$ [6, 4]).

Furthermore, we show that if $\mathcal{NISZK}$ is closed under complementation, then in fact $\mathcal{SZK} = \mathcal{NISZK}$ — i.e., all statistical zero-knowledge proofs can be made non-interactive. We note that [11] does in fact claim that $\mathcal{NISZK}$ is closed under complementation; however, we were not able to verify this claim.

We also show the equivalence of a weakened form of $\mathcal{NISZK}$ and $\mathcal{NISZK}$.

Complete Problems. Central to our methodology is the use of simple and natural complete problems to understand classes with rather complicated definitions, such as $\mathcal{SZK}$ and $\mathcal{NISZK}$. In particular, we exhibit two natural promise problems and prove that they are complete for $\mathcal{NISZK}$. The two problems refer to the "distance" (in two different senses) of a given distribution from the uniform one. These two problems are natural restrictions of two promise problems shown complete for $\mathcal{SZK}$, in [32] and [21], respectively. Indeed, our results about the relationship between $\mathcal{SZK}$ and $\mathcal{NISZK}$ come from relating the corresponding complete problems. This general theme of using completeness to simplify the study of a class, rather than as evidence for computational intractability (as is the traditional use of $\mathcal{NP}$-completeness) has been evidenced in a number of recent works (cf., [18, 27, 34, 1, 2]) and has been particularly useful in understanding statistical zero-knowledge (cf., [32, 33, 11, 21]).

1.1 The non-interactive model

Let us recall the definition of a non-interactive statistical zero-knowledge proof system from [6].[2] We will adapt the definition to promise problems. Note that our definition will capture what [6] call a *bounded proof system*, in that each shared reference string can only be used once. In contrast to non-interactive *computational* zero-knowledge (cf., [6, 15]), it is unknown whether *any* problem that has such a (bounded) non-interactive *statistical* zero-knowledge proof system also has one in which the shared reference string can be used an unbounded (polynomial) number of times.

A *non-interactive statistical zero-knowledge* proof system for a promise problem Π is defined by a polynomial $r(n)$, which will give the size of the random *reference string* σ, and a triple of probabilistic machines P, S, and V, where V and S are polynomial-time, such that:

1. (Completeness:) For all $x \in \Pi_{\text{YES}}$, the probability that $V(\sigma, x, P(x, \sigma))$ accepts is at least $2/3$.
2. (Soundness:) For all $x \in \Pi_{\text{NO}}$, the probability that $V(\sigma, x, P(x, \sigma))$ accepts is at most $1/3$.
3. (Zero-Knowledge:) For all $x \in \Pi_{\text{YES}}$, the statistical deviation between the following two distributions is at most $\beta(|x|)$:

$$\begin{array}{ll} (A) & (\sigma, p) : \sigma \leftarrow \{0,1\}^{r(|x|)};\ \ p \leftarrow P(x, \sigma) \\ (B) & S(x) \end{array}$$

where $\beta(n)$ is a negligible function,[3] termed the *simulator deviation*, and the probabilities in Conditions 1 and 2 are taken over the random coins of V and P, and the choice of σ uniformly from $\{0,1\}^{r(n)}$. Note that non-interactive statistical zero-knowledge is closed under parallel repetition, so the completeness and soundness

[2] Actually, only non-interactive *perfect* and computational zero-knowledge proofs were defined in [6]. The definition we are using, previously given in [4, 11], is the natural non-interactive analogue of (interactive) statistical zero-knowledge [23].

[3] Recall that a function is *negligible* if it is eventually less than $1/g(n)$ for any polynomial g.

errors (i.e. the probability of rejection (resp., acceptance) for YES (resp., NO) instances) can be made exponentially small in $|x|$.

We also define a weaker notion of zero-knowledge, known as a *weak non-interactive statistical zero-knowledge proof system,* where we ask only that for every polynomial $g(n)$, there exists a probabilistic polynomial-time simulator S_g (whose running time may depend on g), such that the simulator deviation as defined above is at most $1/g(|x|)$. This is the natural analogue of a notion defined in the interactive setting for statistical zero-knowledge [12] as well as concurrent zero-knowledge [14].

The class of promise problems that possess non-interactive statistical zero-knowledge proof systems is denoted $\mathcal{NISZK}$, and we denote by $\mathit{weak}\text{-}\mathcal{NISZK}$ the class of promise problems that possess weak non-interactive statistical zero-knowledge proof systems. Note that by definition, $\mathcal{NISZK} \subset \mathit{weak}\text{-}\mathcal{NISZK}$. De Santis *et. al.* [11] recently began investigating $\mathcal{NISZK}$. They introduced a promise problem, called `Image Density`, and claimed that is complete for $\mathcal{NISZK}$ and that the latter class is closed under OR and complementation. We were able to verify that some variants of `Image Density` are $\mathcal{NISZK}$-complete, and indeed the ideas used towards this goal are important to our work. However, we were not able to verify the claim that $\mathcal{NISZK}$ is closed under OR and/or complementation, and for this reason, do not rely on this claim in our work.

In this paper, in addition to examining $\mathcal{NISZK}$ on its own, we also consider the relationship non-interactive statistical zero-knowledge proofs have with *interactive* statistical zero-knowledge proofs. In the context of interactive zero-knowledge proofs, another issue that arises in the zero-knowledge condition is the behavior of the verifier. The general definition of zero-knowledge requires that the zero-knowledge requirement hold for any probabilistic polynomial-time verifier. A weaker requirement, called *honest verifier zero-knowledge,* requires the zero-knowledge condition to hold only if the verifier behaves honestly. However, it is known that these two conditions are equivalent for statistical zero-knowledge, in the sense that every statistical zero-knowledge proof against the honest verifier can be transformed into one that is statistical zero-knowledge against any verifier [20]. Thus, we write $\mathcal{SZK}$ for the class of promise problems possessing statistical zero-knowledge proofs (against any polynomial-time verifier or, equivalently, against just the honest verifier).

Note that in the case of non-interactive zero-knowledge, the issue of honest verifiers does not arise since the verifier does not interact with the prover. Also, note that we can always transform a non-interactive zero-knowledge proof into an honest verifier zero-knowledge proof, since we could have the honest verifier supply a random string which can replace the common reference string required for non-interactive zero-knowledge. That is, $\mathcal{NISZK} \subset \mathcal{SZK}$ (recalling the equivalence of $\mathcal{SZK}$ with honest-verifier $\mathcal{SZK}$).

1.2 Our Results

The primary tools we use in our investigation are promise problems that are complete for $\mathcal{SZK}$ or $\mathcal{NISZK}$. In [32], a promise problem called Statistical Difference (SD) was introduced and proved complete for $\mathcal{SZK}$, providing the first completeness result for $\mathcal{SZK}$. Recently, it was shown in [21] that another natural problem, called Entropy Difference (ED), is complete for $\mathcal{SZK}$ as well. In this work, we show that "one-sided" versions of these problems, which we call Statistical Difference from Uniform (SDU) and Entropy Approximation (EA), are complete for $\mathcal{NISZK}$. To define these problems more precisely, we first recall that that *statistical difference* between two random variables X and Y on a finite set D, denoted $\Delta(X\,,\,Y)$, is defined to be

$$\Delta(X\,,\,Y) \stackrel{\text{def}}{=} \max_{S \subset D} |\Pr[X \in S] - \Pr[Y \in S]| = \frac{1}{2} \cdot \sum_{\alpha} |\Pr[X = \alpha] - \Pr[Y = \alpha]|.$$

All the promise problems we consider involve distributions which are encoded by circuits which sample from them. That is, if X is a circuit mapping $\{0,1\}^m$ to $\{0,1\}^n$, we identify X with the probability distribution induced on $\{0,1\}^n$ by feeding X the uniform distribution on $\{0,1\}^m$.

Definition 1.1 (Problems involving statistical difference): *The promise problem* Statistical Difference*, de-*

noted $\mathtt{SD} = (\mathtt{SD}_{\mathrm{YES}}, \mathtt{SD}_{\mathrm{NO}})$*, consists of*

$$\begin{aligned} \mathtt{SD}_{\mathrm{YES}} &\stackrel{\mathrm{def}}{=} \{(X,Y) : \Delta(X\,,\,Y) < 1/3\} \\ \mathtt{SD}_{\mathrm{NO}} &\stackrel{\mathrm{def}}{=} \{(X,Y) : \Delta(X\,,\,Y) > 2/3\} \end{aligned}$$

where X and Y are distributions encoded as circuits which sample from them. The promise problem Statistical Difference from Uniform*, denoted* $\mathtt{SDU} = (\mathtt{SDU}_{\mathrm{YES}}, \mathtt{SDU}_{\mathrm{NO}})$*, consists of*

$$\begin{aligned} \mathtt{SDU}_{\mathrm{YES}} &\stackrel{\mathrm{def}}{=} \{X : \Delta(X\,,\,U) < 1 - 1/n\} \\ \mathtt{SDU}_{\mathrm{NO}} &\stackrel{\mathrm{def}}{=} \{X : \Delta(X\,,\,U) > 1/n\} \end{aligned}$$

where X is a distribution encoded as a circuit outputing n bits, and U is the uniform distribution on n bits.

For the two problems related to entropy, we recall that the (Shannon) entropy of a random variable X, denoted $\mathrm{H}(X)$, is defined as

$$\mathrm{H}(X) \stackrel{\mathrm{def}}{=} \sum_{\alpha} \Pr[X = \alpha] \cdot \log_2(1/\Pr[X = \alpha])$$

Definition 1.2 (Problems involving entropy): *The promise problem* Entropy Difference*, denoted* $\mathtt{ED} = (\mathtt{ED}_{\mathrm{YES}}, \mathtt{ED}_{\mathrm{NO}})$*, consists of*

$$\begin{aligned} \mathtt{ED}_{\mathrm{YES}} &\stackrel{\mathrm{def}}{=} \{(X,Y) : \mathrm{H}(X) > \mathrm{H}(Y) + 1\} \\ \mathtt{ED}_{\mathrm{NO}} &\stackrel{\mathrm{def}}{=} \{(X,Y) : \mathrm{H}(Y) > \mathrm{H}(X) + 1\} \end{aligned}$$

The promise problem Entropy Approximation*, denoted* $\mathtt{EA} = (\mathtt{EA}_{\mathrm{YES}}, \mathtt{EA}_{\mathrm{NO}})$*, consists of*

$$\begin{aligned} \mathtt{EA}_{\mathrm{YES}} &\stackrel{\mathrm{def}}{=} \{(X,k) : \mathrm{H}(X) > k + 1\} \\ \mathtt{EA}_{\mathrm{NO}} &\stackrel{\mathrm{def}}{=} \{(X,k) : \mathrm{H}(X) < k - 1\} \end{aligned}$$

In these problems, k is a positive integer and X and Y are distributions encoded as circuits which sample from them.

Our first theorem, which is the starting point for our other results, is:

Theorem 1.3 ($\mathtt{EA}$ and $\mathtt{SDU}$ are $\mathcal{NISZK}$-complete) *The promise problems* $\mathtt{EA}$ *and* $\mathtt{SDU}$ *are complete for* $\mathcal{NISZK}$*. That is,* $\mathtt{EA}, \mathtt{SDU} \in \mathcal{NISZK}$ *and for every promise problem* $\Pi \in \mathcal{NISZK}$*, there is a polynomial time many-to-one reduction from* Π *to* $\mathtt{EA}$ *and another from* Π *to* $\mathtt{SDU}$*.*

From the proof of this theorem, we also deduce the equivalence of $\mathcal{NISZK}$ with its weakened form.

Theorem 1.4 *weak*-$\mathcal{NISZK}$ = $\mathcal{NISZK}$.

Armed with our complete problems, we then begin the work of comparing $\mathcal{SZK}$ and $\mathcal{NISZK}$. First we show that the non-triviality of $\mathcal{NISZK}$ is equivalant to the non-triviality of $\mathcal{SZK}$. This is shown by giving a Cook reduction from $\mathtt{ED}$ to $\mathtt{EA}$.

Theorem 1.5 (non-triviality of $\mathcal{NISZK}$) $\mathcal{SZK} \neq \mathcal{BPP} \iff \mathcal{NISZK} \neq \mathcal{BPP}$.

In this theorem (and throughout the paper), $\mathcal{BPP}$ denotes the class of *promise problems* solvable in probabilitic polynomial time.

In fact, it turns out that the type of Cook reduction we use is a special one, and by examining it further, we are able to shed more light on the $\mathcal{SZK}$ vs. $\mathcal{NISZK}$ question. Specifically, we observe that the reduction we give from ED to EA is an $\mathcal{AC}^0$ *truth-table reduction.* That is, it is a nonadaptive Cook reduction in which the postprocessing is done in $\mathcal{AC}^0$. (Formal definitions are given in Section 4.2.) Further, we can prove that if $\mathcal{NISZK}$ is closed under complementation, then $\mathcal{NISZK}$ is closed under $\mathcal{AC}^0$ truth-table reductions. Thus we deduce that $\mathcal{NISZK}$ being closed under complementation implies that $\mathcal{NISZK} = \mathcal{SZK}$. In fact, we can show that closure under complementation and a number of other natural conditions are equivalent to $\mathcal{SZK} = \mathcal{NISZK}$:

Theorem 1.6 (conditions for $\mathcal{SZK} = \mathcal{NISZK}$) *The following are equivalent:*

1. $\mathcal{SZK} = \mathcal{NISZK}$.
2. $\mathcal{NISZK}$ *is closed under complementation.*
3. $\mathcal{NISZK}$ *is closed under* $\mathcal{NC}^1$ *truth-table reductions.*
4. ED *(resp.,* SD*) Karp-reduces to* EA *(resp.,* SDU*).* *("general versions reduce to one-sided ones")*
5. EA *(resp.,* SDU*) Karp-reduces to its complement.* *("one-sided versions reduce to their complements")*

Theorem 1.6 can be interpreted as saying that if $\mathcal{NISZK}$ has a relatively weak closure property (closure under complementation), then the class is surprisingly rich (equals $\mathcal{SZK}$) and has a much stronger closure property (closure under $\mathcal{NC}^1$ truth-table reductions.) Moreover, the last two conditions in Theorem 1.6 show that these questions about non-interactive versus interactive statistical zero-knowledge proofs are actually equivalent to basic, intriguing questions about relationships between natural computational problems whose definitions have no *a priori* relationship to zero-knowledge proofs. Recall that [11] claim that the second item above holds, and consequently if this claim is valid, then all items above hold. However, as stated above, we were not able to verify this claim of [11].

The equality of $\mathcal{SZK}$ and $\mathcal{NISZK}$ has interesting consequences not just for $\mathcal{NISZK}$, but also for $\mathcal{SZK}$. Currently, the best known generic protocol for $\mathcal{SZK}$ requires a polynomial number of rounds [29, 21, 20]. For $\mathcal{NISZK}$, however, by [10, 20], it is known that every problem in $\mathcal{NISZK}$ has a *constant round* statistical zero-knowledge proof system (against general, cheating verifiers) with inverse polynomial soundness error. Whether every problem in $\mathcal{SZK}$ has such a proof system is still an open question, which would be resolved in the positive if $\mathcal{SZK} = \mathcal{NISZK}$.

1.3 A wider perspective

The study of non-interactive *statistical* (rather than *computational*) zero-knowledge proofs may be of interest for two reasons. Firstly, *statistical* zero-knowledge proofs provide an almost absolute level of security, whereas *computational* zero-knowledge proofs only provide security relative to computational abilities (and typically under complexity theoretic assumptions). Secondly, by analogy from the study of zero-knowledge *interactive* proofs, we believe that techniques developed for the "cleaner" statistical model can be applied or augmented to yield results for computational zero-knowledge: The proof that one-way functions are necessary for $\mathcal{SZK}$ to be non-trivial [30] was later generalized to $\mathcal{CZK}$ [31]. More recently, the transformations of honest-verifier zero-knowledge to general zero-knowledge, presented in [8, 10, 9, 20], apply both to statistical and computational zero-knowledge (whereas the original motivation was the study of statistical zero-knowledge). It is our hope that the current study of $\mathcal{NISZK}$ will eventually lead to a better understanding of $\mathcal{NICZK}$, where there are still important open questions such as the conditions under which $\mathcal{NP}$ has $\mathcal{NICZK}$ proofs.

2 $\texttt{EA}$ is in $\mathcal{NISZK}$

In this section, we show that $\texttt{EA}$ has a non-interactive statistical zero-knowledge proof system. Our proof essentially follows the line of reasoning used by [11] to show that `Image Density` is in $\mathcal{NISZK}$.

Lemma 2.1 *$\texttt{EA} \in \mathcal{NISZK}$. Moreover, there is a non-interactive statistical zero-knowledge proof system for $\texttt{EA}$ in which the completeness error, soundness error, and simulator deviation are all exponentially vanishing.*

The transformation given by the following lemma will be applied at the start of the proof system:

Lemma 2.2 *There is a polynomial-time computable function that takes an instance (X, k) of $\texttt{EA}$ and a parameter s (in unary) and produces a distribution Z on $\{0,1\}^\ell$ (encoded by a circuit which samples from it) such that*

1. *If $\mathrm{H}(X) > k+1$, then Z has statistical difference at most $2^{-\Omega(s)}$ from the uniform distribution on $\{0,1\}^\ell$, and*
2. *If $\mathrm{H}(X) < k-1$, then the support of Z is at most a $2^{-\Omega(s)}$ fraction of $\{0,1\}^\ell$.*

The proof of Lemma 2.2, though somewhat technical, uses standard techniques which are implicit in many works. For this reason, the proof is deferred to Appendix C. Given this transformation, it is straightforward to give a noninteractive statistical zero-knowledge proof system for $\texttt{EA}$:

Non-interactive proof system for $\texttt{EA}$, on input (X, k)

1. Let Z be the distribution on $\{0,1\}^\ell$ obtained from (X, k) as in Lemma 2.2 taking s to be the total description length of (X, k) in bits. Let $\sigma \in \{0,1\}^\ell$ be the reference string.
2. P selects r uniformly among $\{r' \colon Z(r') = \sigma\}$ and sends r to V.
3. V accept if $Z(r) = \sigma$ and rejects otherwise.

It is immediate from Lemma 2.2 that the completeness error and soundness error of this proof system are $2^{-\Omega(s)}$. For zero-knowledgeness, we consider the following probabilistic polynomial-time simulator:

Simulator for $\texttt{EA}$ proof system, on input (X, k)

1. Let Z be obtained from (X, k) as in the proof system.
2. Select an input r to Z uniformly at random and let $\sigma = Z(r)$.
3. Output (σ, r).

It follows from Part 1 of Lemma 2.2 that this simulator has statistical difference at most $2^{-\Omega(s)}$ from the distribution of transcripts of (P, V). Thus, assuming Lemma 2.2, we have established Lemma 2.1. In fact, we need not require that s be the length of (X, k). Instead, s can be taken to be an arbitrary security parameter, and the completeness, soundness, and simulation error will be exponentially small in s, while the running time of the protocol only depends polynomially on s. We can use this to prove the following, which will be useful to us later.

Proposition 1 *If any promise problem Π reduces to $\texttt{EA}$ by a Karp (i.e. many-one) reduction (even if it is length-reducing), then $\Pi \in \mathcal{NISZK}$.*

Proof: A noninteractive statistical zero-knowledge proof system for Π can be given as follows: On an instance x of Π, both parties compute the image (X, k) of x under the reduction $\Pi \leq_{\texttt{Karp}} \texttt{EA}$ and execute the proof system for $\texttt{EA}$ on (X, k), taking s to be the length of $|x|$. Hence, the completeness and soundness errors and simulator deviation of this proof system are exponentially small in $|x|$ (rather than $|(X, k)|$ which could be shorter than x). $\blacksquare$

3 EA and SDU are $\mathcal{NISZK}$-complete

In this section, we complete the proof of Theorem 1.3. First, we establish that $\mathtt{SDU} \in \mathcal{NISZK}$ by showing:

Lemma 3.1 $\mathtt{SDU} \leq_{\mathtt{Karp}} \mathtt{EA}$.

Proof: Let X be an instance of SDU. We assume that $\log(n) > 5$, where n is the output length of the circuit X (otherwise, once can decide in probabilistic polynomial time whether X is a YES or NO instance of SDU by random sampling). Let U denote the uniform distribution on n bits. We claim the map $X \mapsto (X, n-3)$ is the reduction required by the lemma.

If $X \in \mathtt{SDU}_{\mathrm{YES}}$, then $\delta = \Delta(X, U) < 1/n$, so X is very close to the uniform distribution, which has entropy n. An argument given in Appendix B allows us to bound difference in entropy in terms of statistical difference. Applying Fact B.1, we immediately conclude that $\mathrm{H}(X) > n-2$.

If $X \in \mathtt{SDU}_{\mathrm{NO}}$, then $\Delta(X, U) \geq 1 - 1/n$. By the definiton of statistical difference, this implies the existence of a set $S \subset \{0,1\}^n$ such that $\Pr[X \in S] - \Pr[U \in S] > 1 - 1/n$. This implies that

$$\Pr[X \in S] > 1 - 1/n \quad \text{and} \quad \Pr[U \in S] < 1/n.$$

Thus, $\mathrm{H}(X) \leq \Pr[X \in S] \cdot \log(|S|) + \Pr[X \notin S] \cdot n < 1 \cdot (n - \log n) + (1/n) \cdot n < n - 4$, and we have that $(X, n-3) \in \mathtt{EA}_{\mathrm{NO}}$. ■

Now, we establish both Theorem 1.3 and Theorem 1.4 by showing that all promise problems in weak-$\mathcal{NISZK}$ (and hence all promise problems in $\mathcal{NISZK}$) are reducible to SDU (and hence by the previous lemma to EA).

Lemma 3.2 *For all promise problems* $\Pi \in$ *weak-*$\mathcal{NISZK}$*, we have that* $\Pi \leq_{\mathtt{Karp}} \mathtt{SDU}$.

Proof: Let Π be any promise problem in *weak-*$\mathcal{NISZK}$. As *weak-*$\mathcal{NISZK}$ is preserved under parallel repetition, we may assume that Π has a *weak-*$\mathcal{NISZK}$ proof system (P, V) with completeness and soundness errors at most 2^{-n} on inputs of length n. Let $r(n) = \mathrm{poly}(n)$ be the length of the random reference string in (P, V), and let S be a randomized polynomial-time simulator S such that the statistical difference between the output distribution of S and the distribution of true transcripts of P is at most $1/(3r(n))$. (Such an S is guaranteed by the *weak-*$\mathcal{NISZK}$ property.) Let U denote the uniform distribution on $r(n)$ bits.

Let x be an instance of Π. Define M_x to be a circuit which does the following on input s:
$M_x(s)$: Simulate $S(x)$ with randomness s to obtain a transcript (σ, p). If $V(\sigma, p)$ accepts, then output σ, else output $0^{r(n)}$.

We claim that the map $x \mapsto M_x$ is the reduction required by the lemma. Suppose $x \in \Pi_{\mathrm{YES}}$. In this case, we know that the random reference string σ in the output of S has statistical difference less than $1/3r(n)$ from U. In addition, since the completeness error of protocol P is at most 2^{-n}, $S(x)$ can output rejecting transcripts with probability at most $1/(3r(n)) + 2^{-n} \leq 2/(3r(n))$. Hence, $\Delta(M_x, U) < 2/(3r(n)) + 1/(3r(n)) \leq 1/r(n)$, and $M_x \in \mathtt{SDU}_{\mathrm{YES}}$.

Suppose $x \in \Pi_{\mathrm{NO}}$. Since the soundness error of protocol P is bounded by 2^{-n}, for at most a 2^{-n} fraction of reference strings σ does there exist an accepting transcript (σ, p). Since M_x only outputs reference strings corresponding to accepting transcripts or $0^{r(n)}$, $\Delta(M_x, U) \geq 1 - (2^{-n} + 1/2^{r(n)}) > 1 - 1/r(n)$. Thus, $M_x \in \mathtt{SDU}_{\mathrm{NO}}$. ■

Clearly, Lemmas 2.1, 3.1, and 3.2 combine to prove Theorem 1.3. Lemmas 3.2 and 3.1 show that any promise problem Π in weak-$\mathcal{NISZK}$ reduces to EA; by Proposition 1, this implies that $\Pi \in \mathcal{NISZK}$ and establishes Theorem 1.4.

4 Comparing $\mathcal{NISZK}$ and $\mathcal{SZK}$

Now that we are armed with $\mathcal{NISZK}$-completeness results for promise problems so closely related to problems known to be complete for $\mathcal{SZK}$, we can quickly begin relating the two classes.

4.1 Nontriviality of $\mathcal{NISZK}$

First, we establish Theorem 1.5 by giving a Cook reduction from Entropy Difference (ED), complete for $\mathcal{SZK}$, to Entropy Approximation (EA), complete for $\mathcal{NISZK}$.

Lemma 4.1 *Suppose (X, Y) is an instance of* ED. *Let $X' = \otimes^4 X$ (resp., $Y' = \otimes^4 Y$) consist of 4 independent copies of X (resp., Y), and let n denote the maximum of the output sizes of X' and Y'. Then,*

$$(X,Y) \in \mathtt{ED}_{\mathrm{YES}} \implies \bigvee_{k=1}^{n} \left[((X',k) \in \mathtt{EA}_{\mathrm{YES}}) \wedge ((Y',k) \in \mathtt{EA}_{\mathrm{NO}}) \right]$$

$$(X,Y) \in \mathtt{ED}_{\mathrm{NO}} \implies \bigwedge_{k=1}^{n} \left[((X',k) \in \mathtt{EA}_{\mathrm{NO}}) \vee ((Y',k) \in \mathtt{EA}_{\mathrm{YES}}) \right]$$

Proof: Suppose $(X,Y) \in \mathtt{ED}_{\mathrm{YES}}$, so that $H(X') > H(Y') + 4$. Let $k = \lfloor H(X') \rfloor - 2$. Hence, $H(X') > k+1$. But $k+3 > H(X') > H(Y')+4$, and hence $H(Y') < k-1$. Suppose instead $(X,Y) \in \mathtt{ED}_{\mathrm{NO}}$, so that $H(Y') > H(X')+4$. Then for all $k > \lceil H(X') \rceil + 1$, we have $H(X') < k-1$. But for all $k \leq \lceil H(X') \rceil + 1$, we have $k+1 < H(X') + 3 < H(Y')$. ■

From this reduction, we conclude that $\mathcal{SZK} \neq \mathcal{BPP} \iff \mathcal{NISZK} \neq \mathcal{BPP}$, which is Theorem 1.5. Again, by $\mathcal{BPP}$ we mean the class of *promise problems* solvable in probabilistic polynomial time.

Proof of Theorem 1.5. By definition, $\mathcal{NISZK} \subset \mathcal{SZK}$ (recall that $\mathcal{SZK}$ equals honest-verifier $\mathcal{SZK}$ [20]). Hence if $\mathcal{SZK} = \mathcal{BPP}$, then $\mathcal{NISZK} = \mathcal{BPP}$.

Now suppose $\mathcal{NISZK} = \mathcal{BPP}$, so in particular there is a probabilistic polynomial-time machine M which decides EA (with exponentially small error probability). To show $\mathcal{SZK} = \mathcal{BPP}$, it suffices to show that $\mathtt{ED} \in \mathcal{BPP}$ since ED is $\mathcal{SZK}$-complete. We now describe how to decide instances of ED: Let (X, Y) be an instance of ED. Letting X' and Y' be as stated in Lemma 4.1, we run $M(X', k)$ and $M(Y', k)$ for all $k \in [1, n]$. If for some k, we see that $M(X', k) = 1$ and $M(Y', k) = 0$, we output 1. Otherwise, we output 0. By Lemma 4.1, this is a correct $\mathcal{BPP}$ algorithm for deciding ED. ■.

4.2 Conditions under which $\mathcal{NISZK} = \mathcal{SZK}$

Although the reduction given in Lemma 4.1 is a Cook reduction, it is a very special type of Cook reduction, which we call an $\mathcal{AC}^0$ truth-table reduction. We use the special properties of this reduction to show that if $\mathcal{NISZK}$ is closed under complement, then in fact $\mathcal{NISZK} = \mathcal{SZK}$. We now precisely define the types of reductions we are using, taking care how we define them for promise problems.

Definition 4.2 (truth-table reduction [26]): *We say a promise problem* Π truth-table reduces *to a promise problem Γ, written $\Pi \leq_{\mathtt{tt}} \Gamma$, if there exists a (deterministic) polynomial-time computable function f, which on input x produces a tuple $(x_1, x_2, \ldots, x_k)$ and a circuit C, such that*

1. *If $x \in \Pi_{\mathrm{YES}}$ then for all valid settings of $b_1, b_2, \ldots, b_k$, $C(b_1, b_2, \ldots, b_k) = 1$, and*
2. *If $x \in \Pi_{\mathrm{NO}}$ then for all valid settings of $b_1, b_2, \ldots, b_k$, $C(b_1, b_2, \ldots, b_k) = 0$.*

where a setting for b_i is considered valid when $b_i = 1$ if $x_i \in \Gamma_{\mathrm{YES}}$ and $b_i = 0$ if $x_i \in \Gamma_{\mathrm{NO}}$ (and b_i is unrestricted when x_i violates the promise).

In other words, a truth-table reduction for promise problems is a non-adaptive Cook reduction which is allowed to make queries which violate the promise, but must be able to tolerate both yes and no answers in response to queries that violate the promise. We further consider the case where we restrict the complexity of computing the output of the reduction from the queries:

Definition 4.3 ($\mathcal{AC}^0$ and $\mathcal{NC}^1$ truth-table reductions): *A truth-table reduction f between promise problems is an $\mathcal{AC}^0$* (resp., $\mathcal{NC}^1$) truth-table reduction *if the circuit C produced by the reduction on input x has depth bounded by a constant c_f independent of x (resp., has depth bounded by $c_f \log |x|$). If there is an $\mathcal{AC}^0$ (resp., $\mathcal{NC}^1$) truth-table reduction from Π to Γ, we write $\Pi \leq_{\mathcal{AC}^0\texttt{-tt}} \Gamma$ (resp., $\Pi \leq_{\mathcal{NC}^1\texttt{-tt}} \Gamma$).*

With this definition, we observe that Lemma 4.1 in fact shows that $\texttt{ED} \leq_{\mathcal{AC}^0\texttt{-tt}} \texttt{EA}$, since the formula given in the lemma can be expressed as an $\mathcal{AC}^0$ circuit, and the statement of the lemma shows that the reduction has the robustness properties against promise violations that are required in Definition 4.3.

We say that a class $\mathcal{C}$ of promise problems is closed under a class of reductions $\leq_*$ if $\Pi \leq_* \Gamma$ and $\Gamma \in \mathcal{C}$ implies that $\Pi \in \mathcal{C}$. By the above, if $\mathcal{NISZK}$ is closed under $\mathcal{AC}^0$ truth-table reductions, then $\texttt{ED} \in \mathcal{NISZK}$ and hence $\mathcal{NISZK} = \mathcal{SZK}$. Thus, we would like to capture the minimal conditions necessary for a promise class to be closed under $\mathcal{AC}^0$ truth-table reductions. Here, care must be taken to because of the possibility of promise violations. Keeping this in mind, we define the following operator on promise problems to capture the notion of an unbounded fan-in AND gate for promise problems:

Definition 4.4 *(unbounded AND): For any promise problem Π, we define $\texttt{AND}(\Pi)$ to be the promise problem:*

$$\begin{aligned} \texttt{AND}_{\text{YES}}(\Pi) &\stackrel{\text{def}}{=} \{(x_1, x_2, \ldots, x_k) : k \geq 0, \forall i \in [1,k] x_i \in \Pi_{\text{YES}}\} \\ \texttt{AND}_{\text{NO}}(\Pi) &\stackrel{\text{def}}{=} \{(x_1, x_2, \ldots, x_k) : k \geq 0, \exists i \in [1,k] x_i \in \Pi_{\text{NO}}\} \end{aligned}$$

We say a class of promise problems $\mathcal{C}$ is closed under unbounded AND *if for all $\Pi \in \mathcal{C}$, one has $\texttt{AND}(\Pi) \in \mathcal{C}$.*

We have defined $\texttt{AND}$ so that it has the weakest promise condition possible to remain well-defined. In particular, we see that $\texttt{AND}_{\text{NO}}(\Pi)$ is defined to include x_i's that violate Π's promise, as long as just *one* of them is in Π_{NO}. $\Pi \in \mathcal{C}$, $\texttt{AND}(\Pi) \in \mathcal{C}$. We also need a way of combining two promise problems:

Definition 4.5 *(disjoint union): For any pair of promise problems Π and Γ, we define the* disjoint union *of Π and Γ to be the promise problem $\texttt{DisjUn}(\Pi, \Gamma)$ defined as follows:*

$$\begin{aligned} \texttt{DisjUn}_{\text{YES}}(\Pi, \Gamma) &\stackrel{\text{def}}{=} \{0\} \times \Pi_{\text{YES}} \cup \{1\} \times \Gamma_{\text{YES}} \\ \texttt{DisjUn}_{\text{NO}}(\Pi, \Gamma) &\stackrel{\text{def}}{=} \{0\} \times \Pi_{\text{NO}} \cup \{1\} \times \Gamma_{\text{NO}} \end{aligned}$$

We say a class of promise problems $\mathcal{C}$ is closed under selection *if for all $\Pi, \Gamma \in \mathcal{C}$, one has $\texttt{DisjUn}(\Pi, \Gamma) \in \mathcal{C}$.*

With these definitions, we can give the following lemma which gives some conditions sufficient to give closure under AC^0 truth-table reductions.

Lemma 4.6 *A promise class $\mathcal{C}$ is closed under $\mathcal{AC}^0$ truth-table reductions if the following conditions hold:*

1. *$\mathcal{C}$ is closed under Karp (i.e., many-one) reductions.*
2. *$\mathcal{C}$ is closed under unbounded AND.*
3. *$\mathcal{C}$ is closed under selection.*
4. *$\mathcal{C}$ is closed under complementation.*

Lemma 4.6 can be proven by a straightforward induction on the depth of the circuits. Details are in Appendix E. Which of the conditions of Lemma 4.6 does $\mathcal{NISZK}$ satisfy? We argue that Conditions 1, 2, and 3 are satisfied by $\mathcal{NISZK}$:

Lemma 4.7 *$\mathcal{NISZK}$ is closed under Karp reductions.*

Proof: Suppose $\Gamma \in \mathcal{NISZK}$, and $\Pi \leq_{\texttt{Karp}} \Gamma$. Since $\texttt{EA}$ is complete for $\mathcal{NISZK}$, we have $\Gamma \leq_{\texttt{Karp}} \texttt{EA}$. By composing reductions, we see that $\Pi \leq_{\texttt{Karp}} \texttt{EA}$. By Lemma 2.1 and Proposition 1, $\Pi \in \mathcal{NISZK}$. ■

Lemma 4.8 *$\mathcal{NISZK}$ is closed under unbounded AND.*

Proof: First, we argue that $\mathtt{AND}(\mathtt{EA}) \in \mathcal{NISZK}$ by describing a $\mathcal{NISZK}$ proof system for $\mathtt{AND}(\mathtt{EA})$: Let $((X_1, k_1), \ldots, (X_m, k_m))$ be an instance of $\mathtt{AND}(\mathtt{EA})$, and say ℓ is the total length of the instance. Artificially pad each circuit X_i to be of description size ℓ (by adding unused gates) and let Y_i be the resulting circuit. Now execute the $\mathcal{NISZK}$ proof system for $\mathtt{EA}$ given by Lemma 2.1 on each pair (Y_i, k_i) in parallel, and have the $\mathtt{AND}(\mathtt{EA})$-verifier accept if the $\mathtt{EA}$-verifier would have accepted on each pair.

If every pair (X_i, k_i) is a YES instance of $\mathtt{EA}$, the $\mathtt{AND}(\mathtt{EA})$ verifier will accept with probability at least $1 - m \cdot 2^{-\Omega(\ell)} = 1 - 2^{-\Omega(\ell)}$, as the completeness error of the $\mathtt{EA}$ proof system is at most $2^{-\Omega(\ell)}$. Similarly, running the simulator for the $\mathtt{EA}$ proof system m times independently will give a simulation for the $\mathtt{AND}(\mathtt{EA})$ proof system with simulator deviation $m \cdot 2^{-\Omega(\ell)} = 2^{-\Omega(\ell)}$. Finally, if just one pair (X_i, k_i) is a NO instance of $\mathtt{EA}$ (even if the others violate the promise), the verifier will accept with probability at most $2^{-\Omega(\ell)}$ in the i'th execution of the $\mathtt{EA}$ protocol, and so the $\mathtt{AND}(\mathtt{EA})$ verifier will accept with probability at most $2^{-\Omega(\ell)}$.

This shows that $\mathtt{AND}(\mathtt{EA}) \in \mathcal{NISZK}$. Now let Π be any promise problem in $\mathcal{NISZK}$. Since $\mathtt{EA}$ is complete for $\mathcal{NISZK}$, there is a Karp reduction f from Π to $\mathtt{EA}$. This induces a Karp reduction from $\mathtt{AND}(\Pi)$ to $\mathtt{AND}(\mathtt{EA})$ in the obvious way (i.e. $(x_1, \ldots, x_k) \mapsto (f(x_1), \ldots, f(x_k))$). As $\mathtt{AND}(\mathtt{EA})$ is in $\mathcal{NISZK}$ and $\mathcal{NISZK}$ is closed under Karp reductions, $\mathtt{AND}(\Pi) \in \mathcal{NISZK}$. ■

Lemma 4.9 *$\mathcal{NISZK}$ is closed under disjoint union.*

Proof: Clearly, $\mathtt{DisjUn}(\mathtt{EA}, \mathtt{EA}) \leq_{\mathtt{Karp}} \mathtt{EA}$ (by dropping the extra bit.) Now, for any two promise problem Π and Γ in $\mathcal{NISZK}$, the Karp reductions f_0 from Π to $\mathtt{EA}$ and f_1 from Γ to $\mathtt{EA}$ induce a Karp reduction from $\mathtt{DisjUn}(\Pi, \Gamma)$ to $\mathtt{DisjUn}(\mathtt{EA}, \mathtt{EA})$ given by $(\sigma, x) \mapsto f_\sigma(x)$. Using $\mathtt{DisjUn}(\Pi, \Gamma) \leq_{\mathtt{Karp}} \mathtt{DisjUn}(\mathtt{EA}, \mathtt{EA}) \leq_{\mathtt{Karp}} \mathtt{EA} \in \mathcal{NISZK}$, and applying closure under Karp reductions, we get $\mathtt{DisjUn}(\Pi, \Gamma) \in \mathcal{NISZK}$. ■

Combining everything, we can give a condition under which $\mathcal{SZK} = \mathcal{NISZK}$.

Proposition 2 *If $\mathcal{NISZK}$ is closed under complementation, then $\mathcal{SZK} = \mathcal{NISZK}$.*

Proof: Suppose $\mathcal{NISZK}$ is closed under complementation. Combining this with Lemmas 4.6, 4.7, 4.8, and 4.9, it follows that $\mathcal{NISZK}$ is closed under $\mathcal{AC}^0$ truth-table reductions. Applying Lemma 4.1 and Lemma 2.1, we conclude that $\mathtt{ED} \in \mathcal{NISZK}$. Since $\mathtt{ED}$ is complete for $\mathcal{SZK}$ [21] and $\mathcal{NISZK}$ is closed under Karp reductions, we have $\mathcal{SZK} \subset \mathcal{NISZK}$. As $\mathcal{NISZK} \subset \mathcal{SZK}$ is true from the definition of $\mathcal{NISZK}$, we conclude that $\mathcal{NISZK} = \mathcal{SZK}$. ■

Finally, we deduce Theorem 1.6, which gives a number of conditions equivalent to $\mathcal{NISZK} = \mathcal{SZK}$.

Proof of Theorem 1.6:
1 ⇒ 3. This follows from the result of [33] that $\mathcal{SZK}$ is closed under $\mathcal{NC}^1$ truth-table reductions.
3 ⇒ 2 ⇒ 1. The first is trivial and the second is Proposition 2.
1 ⇔ 4. This follows from Theorem 1.3 (which asserts that that $\mathtt{EA}$ and $\mathtt{SDU}$ are complete for $\mathcal{NISZK}$), the fact that $\mathtt{ED}$ and $\mathtt{SD}$ are complete for $\mathcal{SZK}$ [32, 21], and Lemma 4.7 (that $\mathcal{NISZK}$ is closed under Karp reductions).
1 ⇔ 5. This follows from Theorem 1.3 (that $\mathtt{EA}$ and $\mathtt{SDU}$ are complete for $\mathcal{NISZK}$) and Lemma 4.7 (that $\mathcal{NISZK}$ is closed under Karp reductions).

References

[1] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and hardness of approximation problems. In *Proceedings of the Thirty Third Annual Symposium on Foundations of Computer Science*, pages 14–23, 1992.

[2] Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs. In *Proceedings of the Thirty Third Annual Symposium on Foundations of Computer Science*, pages 2–13, 1992.

[3] Mihir Bellare and Shafi Goldwasser. New paradigms for digital signatures and message authentication based on non-interactive zero knowledge proofs. In G. Brassard, editor, *Advances in Cryptology—CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*, pages 194–211. Springer-Verlag, 1990, 20–24 August 1989.

[4] Mihir Bellare and Phillip Rogaway. Non-interactive perfect zero-knowledge. Unpublished manuscript, June 1990.

[5] Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications (extended abstract). In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pages 103–112, Chicago, Illinois, 2–4 May 1988.

[6] Manuel Blum, Alfredo De Santis, Silvio Micali, and Giuseppe Persiano. Noninteractive zero-knowledge. *SIAM Journal on Computing*, 20(6):1084–1118, December 1991.

[7] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications. John Wiley & Sons, Inc., 2nd edition, 1991.

[8] Ivan Damgård. Interactive hashing can simplify zero-knowledge protocol design. In *Proceedings of Crypto '95, Lecture Notes in Computer Science*, volume 403, pages 100–109. Springer-Verlag, 1994.

[9] Ivan Damgård, Oded Goldreich, Tatsuaki Okamoto, and Avi Wigderson. Honest verifier vs. dishonest verifier in public coin zero-knowledge proofs. In *Proceedings of Crypto '95, Lecture Notes in Computer Science*, volume 403. Springer-Verlag, 1995.

[10] Ivan Damgård, Oded Goldreich, and Avi Wigderson. Hashing functions can simplify zero-knowledge protocol design (too). Technical Report RS-94–39, BRICS, November 1994. See Part 1 of [9].

[11] Alfredo De Santis, Giovanni Di Crescenzo, Giuseppe Persiano, and Moti Yung. Image Density is complete for non-interactive-SZK. In *Automata, Languages and Programming, 25th International Colloquium*, Lecture Notes in Computer Science, pages 784–795, Aalborg, Denmark, 13–17 July 1998. Springer-Verlag.

[12] Giovanni Di Crescenzo, Tatsuaki Okamoto, and Moti Yung. Keeping the SZK-verifier honest unconditionally. In *Advances in Cryptology—CRYPTO '97*, pages 31–45, 1997.

[13] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography (extended abstract). In *Proceedings of the Twenty Third Annual ACM Symposium on Theory of Computing*, pages 542–552, New Orleans, Louisiana, 6–8 May 1991.

[14] Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing*, pages 409–418, 1998.

[15] Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *31st Annual Symposium on Foundations of Computer Science*, volume I, pages 308–317, St. Louis, Missouri, 22–24 October 1990. IEEE.

[16] Lance Fortnow. The complexity of perfect zero-knowledge. In Silvio Micali, editor, *Advances in Computing Research*, volume 5, pages 327–343. JAC Press, Inc., 1989.

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

[18] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the Association for Computing Machinery*, 38(1):691–729, 1991.

[19] Oded Goldreich and Yair Oren. Definitions and properties of zero-knowledge proof systems. *Journal of Cryptology*, 7(1):1–32, Winter 1994.

[20] Oded Goldreich, Amit Sahai, and Salil Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing*, pages 399–408, 1998.

[21] Oded Goldreich and Salil Vadhan. Comparing entropies in statistical zero-knowledge with applications to the structure of SZK. Available from `http://www-math.mit.edu/~salil`, October 1998.

[22] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[23] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, February 1989.

[24] Russell Impagliazzo, Leonid A. Levin, and Michael Luby. Pseudo-random generation from one-way functions (extended abstracts). In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, pages 12–24, Seattle, Washington, 15–17 May 1989.

[25] Joe Kilian and Erez Petrank. An efficient noninteractive zero-knowledge proof system for NP with general assumptions. *Journal of Cryptology*, 11(1):1–27, Winter 1998.

[26] R. E. Ladner, N. A. Lynch, and A. L. Selman. A comparison of polynomial time reducibilities. *Theoretical Computer Science*, 1(2):103–123, December 1975.

[27] Carsten Lund, Lance Fortnow, Howard Karloff, and Noam Nisan. Algebraic methods for interactive proofs. In *Proceedings of the Thirty First Annual Symposium on Foundations of Computer Science*, pages 1–10, 1990.

[28] Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *Proceedings of the Twenty Second Annual ACM Symposium on Theory of Computing*, pages 427–437, Baltimore, Maryland, 14–16 May 1990.

[29] Tatsuaki Okamoto. On relationships between statistical zero-knowledge proofs. In *Proceedings of the Twenty Eighth Annual ACM Symposium on the Theory of Computing*, 1996. See also preprint of full version, Oct. 1997.

[30] Rafail Ostrovsky. One-way functions, hard on average problems, and statistical zero-knowledge proofs. In *Proceedings of the Thirty Second Annual Symposium on Foundations of Computer Science*, pages 133–138, 1991.

[31] Rafail Ostrovsky and Avi Wigderson. One-way functions are essential for non-trivial zero-knowledge. In *Proceedings of the Second Israel Symposium on Theory of Computing and Systems*, 1993.

[32] Amit Sahai and Salil Vadhan. A complete promise problem for statistical zero-knowledge. In *Proceedings of the Thirty Eighth Annual Symposium on Foundations of Computer Science*, pages 448–457, 1997.

[33] Amit Sahai and Salil Vadhan. Manipulating statistical difference. In Panos Pardalos, Sanguthevar Rajaseekaran, and Jose Rolim, editors, *Proceedings of the DIMACS Workshop on Randomization Methods in Algorithm Design*, Princeton, NJ, 1998. American Mathematical Society. To appear. Available from `http://www-math.mit.edu/~salil/`.

[34] Adi Shamir. IP=PSPACE. In *Proceedings of the Thirty First Annual Symposium on Foundations of Computer Science*, pages 11–15, 1990.

[35] Andrew C. Yao. Theory and application of trapdoor functions. In *Proceedings of the Twenty Third Annual Symposium on Foundations of Computer Science*, pages 80–91, 1982.

A Definitions

Following [17], we extend the standard definition of interactive proof systems to promise problems –

Definition A.1 (Interactive Proof systems – IP [23]): *Let* $c, s : \mathsf{N} \mapsto [0,1]$ *be polynomial-time computable functions so that for some positive polynomial* p *and all positive integers* n*'s,* $c(n) + s(n) < 1 - (1/p(n))$*. An* interactive proof system with two-sided error (c, s) for a promise problem $\Pi = (\Pi_{\text{YES}}, \Pi_{\text{NO}})$ *is a two-party game, between a* verifier *executing a probabilistic polynomial-time strategy* (denoted V) *and a* prover *which executes a computationally unbounded strategy* (denoted P)*, satisfying*

- Completeness*: For every* $x \in \Pi_{\text{YES}}$*, the verifier* V *with probability at least* $1 - c(|x|)$ *accepts after interacting with the prover* P *on common input* x*.*
- Soundness*: For every* $x \in \Pi_{\text{NO}}$ *and every potential strategy* P^**, the verifier* V *accepts with probability at most* $s(|x|)$*, after interacting with* P^* *on common input* x*.*

In such a case, we say that the proof system has completeness error c *and* soundness error s*. The* error *of the proof system is defined as* $\max\{c, s\}$*.*

We are mainly concerned with interactive proof systems having the following zero-knowledge property [23]:

Definition A.2 (Statistical zero-knowledge — $\mathcal{SZK}$):

- *The view of an interactive machine consists of the common input, its internal coin tosses, and all messages it has received. We denote by* $\langle P, V\rangle(x)$ *the* view *of the verifier* V *while interacting with* P *on common input* x*.*
- *A function* $\mu : \mathsf{N} \mapsto [0,1]$ *is called* negligible *if for every positive polynomial* p *and all sufficiently large* $n \in \mathsf{N}$*,* $\mu(n) < 1/p(n)$*.*
- *An interactive proof system* (P, V) *for a promise problem* $\Pi = (\Pi_{\text{YES}}, \Pi_{\text{NO}})$ *is* (general) statistical zero-knowledge *if for every probabilistic polynomial-time* V^**, there exists a probabilistic polynomial-time machine* (called a simulator)*,* S*, and a negligible function* $\mu : \mathsf{N} \mapsto [0,1]$ (called the simulator deviation) *so that for every* $x \in \Pi_{\text{YES}}$ *the statistical difference between* $S(x)$ *and* $\langle P, V^*\rangle(x)$ *is at most* $\mu(|x|)$*.*
- $\mathcal{SZK}$ *denotes the class of promise problems having statistical zero-knowledge interactive proof systems.*

Honest-verifier statistical zero-knowledge proof systems are such where the zero-knowledge requirement is only required to hold for the prescribed/honest verifier V, rather than for every polynomial-time computable V^*. every honest-verifier statistical zero-knowledge proof system can be transformed into a general statistical zero-knowledge proof system (actually meeting an even stronger zero-knowledge requirement) [20].

B Statistical Inequalities

Fact B.1 *For any two random variables, X and Y, ranging over a domain D it holds that*

$$|\mathrm{H}(X) - \mathrm{H}(Y)| \;\; \leq \;\; \log(|D|-1)\cdot\delta \; + \; \mathrm{H}_2(\delta)$$

where $\delta \stackrel{\mathrm{def}}{=} \Delta(X\,,\,Y)$.

This fact can be inferred from Fano's Inequality (cf., [7, Thm. 2.11.1]). A more direct proof follows.

Proof: Assume $\delta > 0$ or else the claim is obvious. Let $p(x) \stackrel{\mathrm{def}}{=} \Pr[X = x]$ and $q(x) \stackrel{\mathrm{def}}{=} \Pr[X = x]$. Define $m(x) \stackrel{\mathrm{def}}{=} \min\{p(x), q(x)\}$. Then $\sum_{x\in D} m(x) = 1-\delta$. Define random variables Z', X' and Y' so that

$$\begin{aligned}
\Pr[Z' = x] &= m'(x) \stackrel{\mathrm{def}}{=} \frac{1}{1-\delta}\cdot m(x)\\
\Pr[X' = x] &= p'(x) \stackrel{\mathrm{def}}{=} \frac{1}{\delta}\cdot (p(x) - m(x))\\
\Pr[Y' = x] &= q'(x) \stackrel{\mathrm{def}}{=} \frac{1}{\delta}\cdot (q(x) - m(x))
\end{aligned}$$

Think of X (resp., Y) as being generated by picking Z' with probability $1-\delta$ and X' (resp., Y') otherwise. Then

$$\begin{aligned}
\mathrm{H}(X) &\leq (1-\delta)\cdot\mathrm{H}(Z') + \delta\cdot\mathrm{H}(X') + \mathrm{H}_2(\delta)\\
\mathrm{H}(Y) &\geq (1-\delta)\cdot\mathrm{H}(Z')
\end{aligned}$$

Observing that $\Pr[X' = x] = 0$ on at least one $x \in D$, it follows that $\mathrm{H}(X') \leq \log(|D|-1)$, and the fact follows. ■

Comment: The above bound is tight. Let $e \in D$ and consider X which is identically e, and Y which with probability $1-\delta$ equals e and otherwise is uniform over $D \setminus \{e\}$. Clearly, $\Delta((\,,X), Y) = \delta$ and $\mathrm{H}(Y) - \mathrm{H}(X) = \delta\log(|D|-1) + \mathrm{H}_2(\delta) - 0$.

C Proof of Lemma 2.2

C.1 Flat distributions and the Leftover Hash Lemma

Here, we discuss some standard notions and techniques that will be useful in the proof of Lemma 2.2. We use the clean formulations of these tools given in [21].

A distribution X is called *flat* if all strings in the support of X have the same probability. Notice that if X is flat, then by the definition of entropy, $\Pr[X = x] = 2^{-\mathrm{H}(X)}$ for every x in the support of X. We quantify deviation from flatness as follows:

Definition C.1 (heavy, light and typical elements): *Let X be a distribution, x an element possibly in its support, and Δ a positive real number. We say that x is* Δ-heavy (resp., Δ-light) *if* $\Pr[X = x] \geq 2^{\Delta}\cdot 2^{-\mathrm{H}(X)}$ (resp., $\Pr[X = x] \leq 2^{-\Delta}\cdot 2^{-\mathrm{H}(X)}$). *Otherwise, we say that x is* Δ-typical.

A natural relaxed definition of flatness follows. The definition links the amount of slackness allowed in "typical" elements with the probability mass assigned to non-typical elements.

Definition C.2 (flat distributions): *A distribution X is called* Δ-flat *if for every $t > 0$, the probability that an element chosen from X is $t \cdot \Delta$-typical is at least $1 - 2^{-t^2+1}$.*

By straightforward application of Hoefding Inequality (cf., Appendix D), we have

Lemma C.3 (flattening lemma): *Let X be a distribution, k a positive integer, and $\otimes^k X$ denote the distribution composed of k independent copies of X. Suppose that for all x in the support of X it holds that $\Pr[X = x] \geq 2^{-m}$. Then $\otimes^k X$ is $\sqrt{k} \cdot m$-flat.*

The key point is that the entropy of $\otimes^k X$ grows linearly with k, whereas its deviation from flatness grows significantly slower (i.e., linear in $\sqrt{k}$) as a function of k. The other main tool we will use is:

Lemma C.4 (Leftover Hash Lemma [24]) *Let $\mathcal{H}$ be a 2-universal family of hash functions mapping a domain D to a range R. Suppose X is a distribution on D such that with probability at least $1 - \delta$ over x selected from X, $\Pr[X = x] \leq \varepsilon/|R|$. Then the statistical difference between the following two distributions is at most $O(\delta + \varepsilon^{1/3})$:*

(A) Choose h uniformly from $\mathcal{H}$ and x according to X. Output $(h, h(x))$.

(B) Choose h uniformly from $\mathcal{H}$ and y uniformly from R. Output (h, y).

In particular, notice that if X is a Δ-flat distribution, then for any parameters $s, t > 0$, X satisfies the hypothesis of the Leftover Hash Lemma with $|R| = 2^{\mathrm{H}(X) - t\Delta - s}$, $\delta = 2^{-t^2+1}$, and $\varepsilon = 2^{-s}$. As we will be applying Lemma C.4 to sets of strings, we define, for any pair of positive integers ℓ and k, $\mathcal{H}_{\ell,k}$ to be one of the standard 2-universal families of hash functions mapping $\{0,1\}^\ell$ to $\{0,1\}^k$ (e.g., affine $\mathrm{GF}(2)$-linear transformations).

C.2 Overview of the transformation

The transformation proceeds in four stages, which are roughly described below:

1. Let X' consist of many copies of X so that the entropy gap between YES and NO instances increases, and the distribution becomes quite flat relative to its entropy.
2. Hash X' so that YES instances become close to the uniform distribution while NO instances have much smaller entropy than the uniform distribution. That is, let Y be of the form $(h, h(X'))$, where h is uniformly distributed in a 2-universal family with appropriate parameters.
3. Let Y' consist of many copies of Y so that for NO instances, the entropy deficiency (as compared to the uniform distribution) becomes large and yet Y' becomes quite flat relative to its entropy; while YES instances remain close to uniform.
4. Hash the *inputs* to Y' so that NO instances have small support (rather than just small entropy), while keeping YES instances close to uniform. That is, let Z be of the form $(Y'(r), h, h(r))$ where h is uniformly distributed in a 2-universal family with appropriate parameters.

C.3 The formal construction and proof

Let (X, k) be an instance of `EA`, let m (resp., n) denote the number of input and output gates to X, and let s be the extra parameter in the transformation. By increasing s if necessary, we may assume that s is greater than the total description length of (X, k). Thus, all the intermediate circuits we build will be of size $\mathrm{poly}(s)$.

Many copies I. The first step is to take many copies of each distribution; this has the effect of increasing the entropy gap between YES and NO instances relative to X's deviation from flatness. Namely, let $q = 4sm^2$ and let $X' = \otimes^q X$. Then $\mathrm{H}(X') = q \cdot \mathrm{H}(X)$ and, by Lemma C.3, X' is Δ-flat for $\Delta = 2\sqrt{s} \cdot m^2$. In particular, we have established

Claim C.5

1. *If* $\mathrm{H}(X) > k + 1$, *then* $\mathrm{H}(X') > qk + q \geq qk + \sqrt{s}\Delta + s$.
2. *If* $\mathrm{H}(X) < k - 1$, *then* $\mathrm{H}(X') < qk - q < qk$.

Hashing I. Now consider the distribution Y on pairs $(h, h(x))$ induced by choosing h uniformly from $\mathcal{H}_{qn,qk+1}$ and x according to X'. Say that elements of $\mathcal{H}_{qn,qk+1}$ take $\ell \leq \mathrm{poly}(qn, qk) \leq \mathrm{poly}(s)$ bits to represent. Then Y has inputs (resp., outputs) of length $m' = \ell + qm$ (resp., $n' = \ell + qk + 1$). Y satisfies the following properties.

Claim C.6

1. *If* $\mathrm{H}(X) > k + 1$, *then Y has statistical difference at most* $2^{-\Omega(s)}$ *from the uniform distribution on* $\{0,1\}^{n'}$.
2. *If* $\mathrm{H}(X) < k - 1$, *then the entropy of Y is less than* $n' - 1$.

Proof: Part 1 follows from the Δ-flatness of X and the Leftover Hash Lemma. Part 2 follows from the fact that the entropy of Y is at most the entropy of X' (which is less than qk) plus the entropy of the uniform distribution on $\mathcal{H}$ (which is ℓ). ■

Many copies II. We now take many copies of Y, so that the entropy deficiency of NO instances becomes large relative to the flatness while YES instances remain close to uniform. Specifically, let $q' = 4s \cdot (m')^2$ and let $Y' = \otimes^{q'} Y$, so that Y' has $M = m'q'$ input gates, $N = n'q'$ output gates, and Y' is Δ'-flat for $\Delta' = 2\sqrt{s} \cdot (m')^2$. Then we immediately have the following:

Claim C.7

1. *If* $\mathrm{H}(X) > k + 1$, *then Y' has statistical difference at most* $q' \cdot 2^{-\Omega(s)} = 2^{-\Omega(s)}$ *from the uniform distribution on* $\{0,1\}^N$.
2. *If* $\mathrm{H}(X) < k - 1$, *then* $\mathrm{H}(Y') < N - q' \leq N - \sqrt{3s} \cdot \Delta' - s$.

Hashing II. The final step is to make a distribution which, for NO instances, has small support (rather than just low entropy) in the case of NO instances, while YES instances remain close to uniform.

Consider a distribution Z which takes as input $r \in \{0,1\}^M$ and a hash function $h \in \mathcal{H}_{M,M-N-s}$ and outputs $(Y'(r), h, h(r))$. Then,

Claim C.8 *Z satisfies the requirements of Lemma 2.2.*

The intuition for this is the following: In the case of YES instances, Y' is close to the uniform distribution on $\{0,1\}^N$, so for almost all $y \in \{0,1\}^N$, there will be about 2^{M-N} values of r such that $Y'(r) = y$. Thus, hashing r down to $M - N - s$ bits will still result in a uniform distribution.

In the case of a NO instance, Y' has large entropy deficiency and is nearly flat. From this, we know that Y' lands in some small subset T of the domain with very high probability. Thus, points $y \notin T$ must have very low probability under Y', i.e. there are very few inputs r such that $Y'(r) = y$. So, for each $y \notin T$, the pairs $(h, h(r))$ will only hit a small subset of the possible values. Therefore, $(Y'(r), h, h(r))$ has small support, because either the first component lands in a small set (namely T) or the last two components land in a small set.

Proof: Suppose $\mathrm{H}(X) > k+1$. From the fact that Y' has statistical difference at most $2^{-\Omega(s)}$ from uniform it follows that with probability at least $1 - 2^{-\Omega(s)}$ over y selected according to Y',

$$\Pr\left[Y' = y\right] \geq \frac{1}{2} \cdot \frac{1}{2^N}. \tag{1}$$

Fix any y satisfying Inequality 1. Conditioned on $Y'(r) = y$, r is selected uniformly from $\{r : Y'(r) = y\}$, which by Equation 1 is a set of size at least 2^{M-N-1}. Thus, by the Leftover Hash Lemma, conditioned on $Y'(r) = y$, the distribution of $(h, h(r))$ has statistical difference at most $2^{-\Omega(s)}$ from uniform. Therefore the total statistical difference of Z from uniform is $2^{-\Omega(s)}$.

Now suppose $\mathrm{H}(X) < k - 1$. We want to show that the support S of Z is a small fraction of $D = \{0,1\}^N \times \mathcal{H} \times \{0,1\}^{M-N-s}$. To do this, we divide S into three parts, depending on the probability mass given to the y component by Y'. Recall that a "typical" y for Y' has probability mass $\approx 2^{-\mathrm{H}(Y')} \geq 2^{-N+\sqrt{3s}\cdot\Delta+s}$.

$$\begin{array}{lcll} S_1 & = & \{(y,h,z) \in S \colon \Pr\left[Y' = y\right] \leq 2^{-N-2s}\} & \text{(“much too light”)} \\ S_2 & = & \{(y,h,z) \in S \colon 2^{-N-2s} < \Pr\left[Y' = y\right] \leq 2^{-N+s}\} & \text{(“too light, but not much too light”)} \\ S_3 & = & \{(y,h,z) \in S \colon 2^{-N+s} < \Pr\left[Y' = y\right]\} & \text{(“not too light”)} \end{array}$$

Clearly, $S = S_1 \cup S_2 \cup S_3$. We will show that $|S_i|/|D| \leq 2^{-s}$ for $i = 1, 2, 3$, and so $|S|/|D| \leq 3 \cdot 2^{-s} = 2^{-\Omega(s)}$.

First we bound $|S_1|$. For any y such that $\Pr\left[Y' = y\right] \leq 2^{-N-2s}$, there are at most 2^{M-N-2s} values of r such that $Y'(r) = y$. Thus, for any such y and any h, the set of z such that $(y, h, z) \in S_1$ is of size at most 2^{M-N-2s} (i.e., is at most a 2^{-s} fraction of $\{0,1\}^{M-N-s}$). This implies that S_1 is at most a 2^{-s} fraction of D.

Now we bound $|S_2|$. We show that the set A of y such that $2^{-N-2s} < \Pr\left[Y' = y\right] \leq 2^{-N+s}$ is at most a 2^{-s} fraction of $\{0,1\}^N$. From this, it follows that S_2 is at most a 2^{-s} fraction of D. Every $y \in A$ is $\sqrt{3s}\cdot\Delta'$-heavy (since Y' has entropy at most $N - s - \sqrt{3s}\cdot\Delta'$). By the Δ'-flatness of Y', $\Pr\left[Y' \in A\right]$ is at most 2^{-3s+1}. Since every y in A has probability mass at least 2^{-N-2s} under Y', $|A|$ is at most $2^{-3s+1}/2^{-N-2s} = 2^{N-s}$, as desired.

Finally, we bound $|S_3|$. Clearly, there can be at most 2^{N-s} values of y such that $\Pr\left[Y' = y\right] \geq 2^{-N+s}$. From this it follows that $|S_3|/|D| \leq 2^{-s}$. ■

D Proof of the Flattening Lemma

For every x in the support of X, we let $w(x) = -\log \Pr\left[X = x\right]$. Then w maps the support of X, denoted D, to $[0, m]$. Let $X_1, ..., X_k$ be identical and independent copies of X. The lemma asserts that for every t

$$\Pr\left[\left|\sum_{i=1}^{k} w(X_i) - k \cdot \mathrm{H}(X)\right| > t \cdot m\sqrt{k}\right] \; < \; 2^{-t^2}$$

Observe that $E(w(X_i)) = \sum_x \Pr\left[X = x\right] w(x) = \mathrm{H}(X)$, for every i. Thus, the lemma follows by a straightforward application of Hoefding Inequality: Specifically, define random variables $\xi_i = w(X_i)$, let $\mu = E(\xi_i)$ and $\delta = tm/\sqrt{k}$, and use

$$\begin{aligned} \Pr\left[\left|\frac{\sum_{i=1}^{k} \xi_i}{k} - \mu\right| > \delta\right] \quad &< \quad 2 \cdot \exp\left(-\frac{2\delta^2}{m^2} \cdot k\right) \\ &= \quad 2 \cdot \exp\left(-2t^2\right) \end{aligned}$$

The lemma follows. ■

E Proof of Lemma 4.6

First note that any unbounded fan-in circuit can be efficiently converted into a circuit with only unbounded fan-in NAND gates (allowing also unary NAND gates), with only a constant factor blowup in depth. to such a circuit with only a constant factor blowup in depth. So, as a first step, we observe that $\mathcal{C}$ is closed under unbounded NAND. That is, for any promise problem Π, $\mathtt{NAND}(\Pi) \stackrel{\text{def}}{=} \overline{\mathtt{AND}(\Pi)} \in \mathcal{C}$, by closure under unbounded AND and complementation. To generalize this to constant depth circuits with unbounded fan-in NAND gates, we define

Definition E.1 *For any promise problem Π, and for all natural numbers $d \geq 0$ we define $\mathtt{Depth}^d(\Pi)$ to be the promise problem whose instances are tuples $(C, (x_1, x_2, \ldots, x_k))$, where C is a circuit of depth at most d (using unbounded fan-in NAND gates only). The* YES *instances are those such that for all valid settings of $b_1, b_2, \ldots, b_k$, $C(b_1, b_2, \ldots, b_m) = 1$; whereas the* NO *instances are those tuples such that for all valid settings of $b_1, b_2, \ldots, b_k$, $C(b_1, b_2, \ldots, b_k) = 0$. Here, a setting for b_i is considered valid when $b_i = 1$ if $x_i \in \Pi_{\text{YES}}$ and $b_i = 0$ if $x_i \in \Pi_{\text{NO}}$ (and b_i is unrestricted when x_i violates the promise).*

Using the fact that every $\mathcal{AC}^0$ circuit can be efficiently transformed into one with only NAND gates, we see that $\Pi \leq_{\mathcal{AC}^0\mathtt{-tt}} \Gamma$ means that there exists some d such that $\Pi \leq \mathtt{Depth}^d(\Gamma)$ under a Karp reduction. Hence if we can show that for all $d \geq 0$ and promise problems Π, $\mathtt{Depth}^d(\Pi) \in \mathcal{C}$, the lemma will be established. We will prove this by induction.

First, observe that a depth 0 circuit is simply a variable (we can ignore constants as trivial). Hence, $\mathtt{Depth}^0(\Pi) = \Pi \in \mathcal{C}$. Now assume that $\mathtt{Depth}^d(\Pi) \in \mathcal{C}$. Observe that a depth $d+1$ circuit is simply a NAND of some number of depth d circuits. Using this observation, we will argue that that

$$\mathtt{Depth}^{d+1}(\Pi) \leq \mathtt{DisjUn}(\mathtt{Depth}^d(\Pi), \mathtt{NAND}(\mathtt{Depth}^d(\Pi)))$$

; by the closure properties of $\mathcal{C}$, this implies that $\mathtt{Depth}^{d+1}(\Pi) \in \mathcal{C}$. The reduction works as follows. The input to the reduction is a tuple $(C, \bar{x})$ where $\bar{x} = (x_1, x_2, \ldots x_k)$. If C is actually a depth d circuit, then it simply outputs $(0, (C, \bar{x}))$. If not, then it extracts from C the circuits $C_1, C_2, \ldots, C_s$ that provide input to the topmost NAND gate. Then the reduction outputs $(1, (m, (C_1, \bar{x}), (C_2, \bar{x}), \ldots, (C_s, \bar{x})))$. It is clear that map gives a Karp reduction from $\mathtt{Depth}^{d+1}(\Pi) \leq \mathtt{DisjUn}(\mathtt{Depth}^d(\Pi), \mathtt{NAND}(\mathtt{Depth}^d(\Pi)))$, completing the induction step and the proof.