Zero Knowledge and Soundness Are Symmetric*

Shien Jin Ong and Salil Vadhan

School of Engineering and Applied Sciences
Harvard University
Cambridge, Massachusetts, USA
{shienjin,salil}@eecs.harvard.edu

Abstract. We give a complexity-theoretic characterization of the class of problems in **NP** having zero-knowledge argument systems. This characterization is symmetric in its treatment of the zero knowledge and the soundness conditions, and thus we deduce that the class of problems in **NP** $\cap$ **coNP** having zero-knowledge arguments is closed under complement. Furthermore, we show that a problem in **NP** has a *statistical* zero-knowledge argument system if and only if its complement has a computational zero-knowledge *proof* system. What is novel about these results is that they are *unconditional*, i.e., do not rely on unproven complexity assumptions such as the existence of one-way functions.

Our characterization of zero-knowledge arguments also enables us to prove a variety of other unconditional results about the class of problems in **NP** having zero-knowledge arguments, such as equivalences between honest-verifier and malicious-verifier zero knowledge, private coins and public coins, inefficient provers and efficient provers, and non-black-box simulation and black-box simulation. Previously, such results were only known unconditionally for zero-knowledge *proof systems*, or under the assumption that one-way functions exist for zero-knowledge argument systems.

1 Introduction

Zero-knowledge protocols are interactive protocols whereby one party, the *prover*, convinces another party, the *verifier*, that some assertion is true with the remarkable property that the verifier "learns nothing" other than the fact that the assertion being proven is true. Since their introduction by Goldwasser, Micali, and Rackoff [GMR], zero-knowledge protocols have played a central role in the design and study of cryptographic protocols.

Zero-knowledge protocols come in several flavors, depending on how one formulates the two security conditions: (1) the zero-knowledge condition, which says that the verifier "learns nothing" other than the fact the assertion being proven is true, and (2) the soundness conditions, which says that the prover

* A preliminary version of this paper appeared in the *Electronic Colloquium on Computational Complexity* [OV]. Both the authors were supported by NSF grant CNS-0430336 and ONR grant N00014-04-1-0478.

M. Naor (Ed.): EUROCRYPT 2007, LNCS 4515, pp. 187–209, 2007.

cannot convince the verifier of a false assertion. In *statistical zero knowledge*, the zero-knowledge condition holds regardless of the computational resources the verifier invests into trying to learn something from the interaction. In *computational zero knowledge*, we only require that a probabilistic polynomial-time verifier learn nothing from the interaction.[1] Similarly, for soundness, we have *statistical soundness*, giving rise to *proof systems*, where even a computationally unbounded prover cannot convince the verifier of a false statement (except with negligible probability), and *computational soundness*, giving rise to *argument systems* [BCC], where we only require that a polynomial-time prover cannot convince the verifier of a false statement. Using a prefix of **S** or **C** to indicate whether the zero knowledge is statistical or computational and a suffix of **P** or **A** to indicate whether we have a proof system or argument system, we obtain four complexity classes corresponding to the different types of zero-knowledge protocols: **SZKP**, **CZKP**, **SZKA**, **CZKA**. More precisely, these are the classes of *decision problems* Π having the correponding type of zero-knowledge protocol. In such a protocol, the prover and verifier are given as common input an instance x of Π, and the prover is trying convince the verifier that x is a YES instance of Π.

These two security conditions seem to be of very different flavors; zero knowledge is a 'secrecy' condition, whereas soundness is more like an 'unforgeability' condition. However, in a remarkable paper, Okamoto [Oka] showed that they are actually symmetric in the case of statistical security.

Theorem 1 ([Oka, GSV][2]). *The class* **SZKP** *of problems having statistical zero-knowledge proofs is closed under complement. That is,* $\Pi \in \mathbf{SZKP}$ *if and only if* $\overline{\Pi} \in \mathbf{SZKP}$.

In a zero-knowledge protocol for proving that a string x is a YES instance of a problem Π, zero knowledge is required only when x is a YES instance (that is, when the statement being proven is true) and soundness is required only when x is a NO instance (that is, when the statement is false). Thus, by showing that **SZKP** is closed under complement, Okamoto established a symmetry between zero knowledge and soundness, in the case when both security conditions are statistical.

We ask whether an analogous theorem holds when the security conditions are *computational*, namely when considering computational zero-knowledge arguments. If we make complexity assumptions, then the answer is yes. Indeed, the classical results of Goldreich, Micali, and Wigderson [GMW], and Brassard, Chaum, and Crépeau [BCC] show that every problem in **NP** has computational

[1] More precisely, in statistical zero knowledge, we require that the verifier's view of the interaction can be efficiently simulated up to negligible statistical distance, whereas in computational zero knowledge, we only require that the simulation be computationally indistinguishable from the verifier's view.

[2] Okamoto's result was actually for the class of languages having *honest-verifier* statistical zero-knowledge proofs, but in [GSV] it was shown this is the same as the class of languages having general statistical zero-knowledge proofs.

zero-knowledge argument systems under widely believed complexity assumptions, and in fact either one of the security conditions can be made statistical. Moreover, it is known that the existence of one-way functions (OWF) suffices for the construction of computational zero-knowledge proof systems and statistical zero-knowledge argument systems for every problem in **NP** [Nao, HILL, NOV]. Thus, the existence of one-way functions implies that computational zero knowledge and computational soundness are symmetric for problems in $\mathbf{NP} \cap \mathbf{coNP}$, by implying that all problems in $\mathbf{NP} \cap \mathbf{coNP}$ and their complements have computational zero-knowledge arguments. We note that here, and throughout the paper, we usually restrict attention to problems in **NP**, because argument systems are mainly of interest when the prover can be implemented in polynomial time given a witness of membership, which only makes sense for problems in **NP**.[3]

In this paper, we establish an *unconditional* symmetry between computational zero knowledge and computational soundness.

Theorem 2 (Symmetry Theorem)

1. (**CZKA** *versus* **co**-**CZKA**) *A problem* $\Pi \in \mathbf{NP} \cap \mathbf{coNP}$ *has a computational zero-knowledge argument system if and only if* $\overline{\Pi}$ *has a computational zero-knowledge argument system.*
2. (**SZKA** *versus* **CZKP**) *A problem* $\Pi \in \mathbf{NP}$ *has a statistical zero-knowledge argument system if and only if* $\overline{\Pi}$ *has a computational zero-knowledge proof system.*

Observe how the quality of the zero-knowledge condition for Π translates to the quality of the soundness condition for $\overline{\Pi}$ and vice-versa.

1.1 The SZKP–OWF Characterization

The Symmetry Theorem is obtained by new characterizations of the classes of problems having zero-knowledge protocols, and moreover these characterizations treat zero knowledge and soundness symmetrically. These characterizations are a generalization of the "SZK/OWF Characterization Theorem" of [Vad], which says that any problem having a computational zero-knowledge *proof* system can be described as a problem having a statistical zero-knowledge proof plus a set of YES instances from which we can construct a one-way function. To characterize zero-knowledge *argument* systems, we will also allow some additional NO instances from which we can construct a one-way function.

To formalize this, we will need the notion of a *promise problem*, which is simply a decision problem with some inputs excluded. More precisely, a promise problem Π consists of two disjoint sets of strings $(\Pi_{\mathrm{Y}}, \Pi_{\mathrm{N}})$, corresponding to YES and NO instances respectively. All of the complexity classes that we consider—for

[3] Actually polynomial-time provers also make sense for problems in **MA**, which is a variant of **NP** where the verification of witnesses is probabilistic. All of our results easily extend to **MA**, but we state them for **NP** for simplicity.

instance, **SZKP**, **CZKP**, **SZKA**, and **CZKA**—generalize to promise problems in a natural way; completeness and zero knowledge are required for YES instances, and soundness is required for NO instances.

Definition 1 (SZKP–OWF CONDITION)**.** *We say that promise problem* $\Pi = (\Pi_\mathrm{Y}, \Pi_\mathrm{N})$ *satisfies the* SZKP–OWF CONDITION *if there exists a set of instances* $I \subseteq \Pi_\mathrm{Y} \cup \Pi_\mathrm{N}$ *such that the following two conditions hold:*

- *The promise problem* $(\Pi_\mathrm{Y} \setminus I, \Pi_\mathrm{N} \setminus I)$ *is in* **SZKP***.*
- *There exists a polynomial-time computable function* $f_x \colon \{0,1\}^{n(|x|)} \to \{0,1\}^{m(|x|)}$*, with* $n(\cdot)$ *and* $m(\cdot)$ *being polynomials and instance* x *given as an auxiliary input, such that for every nonuniform probabilistic polynomial-time adversary* A*, and for every constant* $c > 0$*, we have*

$$\Pr_{y \leftarrow \{0,1\}^{n(|x|)}} \left[A(f_x(y)) \in f_x^{-1}(f_x(y))\right] \leq |x|^{-c} \quad ,$$

for every sufficiently long $x \in I$*.*

We call I *the set of* OWF instances*,* $I \cap \Pi_\mathrm{Y}$ *the set of* OWF YES instances*, and* $I \cap \Pi_\mathrm{N}$ *the set of* OWF NO instances*.*

We use the SZKP–OWF CONDITION to characterize the classes of problems having zero-knowledge protocols.

Theorem 3 (SZKP–OWF Characterization of Zero Knowledge)

1. (**SZKP** [trivial]) *A problem* $\Pi \in$ **IP** *has a statistical zero-knowledge proof system if and only if* Π *satisfies the* SZKP–OWF CONDITION *without* OWF *instances, namely* $I = \emptyset$*.*
2. (**CZKP** [Vad]) *A problem* $\Pi \in$ **IP** *has a computational zero-knowledge proof system if and only if* Π *satisfies the* SZKP–OWF CONDITION *without* OWF NO *instances, namely* $I \cap \Pi_\mathrm{N} = \emptyset$*.*
3. (**SZKA** [new]) *A problem* $\Pi \in$ **NP** *has a statistical zero-knowledge argument system if and only if* Π *satisfies the* SZKP–OWF CONDITION *without* OWF YES *instances, namely* $I \cap \Pi_\mathrm{Y} = \emptyset$*.*
4. (**CZKA** [new]) *A problem* $\Pi \in$ **NP** *has a computational zero-knowledge argument system if and only if* Π *satisfies the* SZKP–OWF CONDITION*.*

Theorem 2, our Symmetry Theorem between computational zero knowledge and computational soundness, follows directly from: (i) Theorem 3 above, (ii) Okamoto's Theorem that **SZKP** is closed under complement (Theorem 1), and (iii) the symmetric role played by the set of OWF instances I in the SZKP–OWF CONDITION.

The advantage of the SZKP–OWF Characterization Theorem is that it reduces the study of the various forms of zero-knowledge protocols to the study of **SZKP** together with the study of the consequences of one-way functions, both of which are by now quite well-developed. Indeed, we also use these characterizations to prove many other unconditional theorems about the classes of problems in **NP** possessing zero-knowledge arguments, such as equivalences between honest-verifier and malicious-verifier zero knowledge, private coins and public coins, inefficient provers and efficient provers, and non-black-box simulation and black-box simulation. Previously, such results were only known unconditionally for the case of zero-knowledge *proof* systems [Oka, GSV, Vad, NV], or were known under the complexity assumptions like the existence of one-way functions for the case of zero-knowledge argument systems [GMW, Nao, HILL, NOV].

While our characterizations of **SZKA** and **CZKA** (Items 3 and 4) are similar in spirit to the **CZKP** characterization of [Vad] (Item 2), both directions of the implications require new ingredients that were not present in [Vad].

In the forward direction, going from **CZKA** or **SZKA** to an SZKP–OWF Condition, we combine the work of [Vad] with an idea of Ostrovsky [Ost] to construct a one-way function on no instances in $I \cap \Pi_{\mathrm{N}}$. Ostrovsky showed that if a *hard-on-average* problem has a statistical zero-knowledge argument system, then (standard) one-way functions exist.[4] (This was later generalized to computational zero knowledge in [OW].) We use the same construction, but with a slightly different analysis. In Ostrovsky's work, the hardness of inverting the one-way function is derived from the assumed (average-case) hardness of the problem having the zero-knowledge protocol, and it is shown to be hard to invert on yes instances. In our proof, the hardness of inverting the one-way function is instead derived from a gap between between statistical soundness and computational soundness, and it is analyzed on no instances.

In the reverse direction, going from an SZKP–OWF Condition to **CZKA** or **SZKA**, there were more fundamental obstacles in extending the work of [Vad]. First, the construction of [Vad] made use of a computationally unbounded prover in an essential way (as did the previous work on **SZKP**, such as [Oka]), whereas argument systems are rather unnatural with unbounded provers and hence are typically defined with respect to efficient provers. Second, at the time we did not know of a construction of statistical zero-knowledge arguments for **NP** from any one-way function, which is necessary to make use of the one-way functions constructed from instances in $I \cap \Pi_{\mathrm{N}}$—this is clear when trying to characterize **SZKA**, but it also turns out to be important for characterizing **CZKA**. Fortunately, both of these obstacles have been recently overcome in [NV] and [NOV], respectively.

In more detail, the way the reverse direction is proved is to show that for any problem Π satisfying the SZKP–OWF Condition, we can construct an *instance-dependent* commitment scheme,[5] and then we use the instance-dependent commitment scheme to construct a zero-knowledge protocol for Π. In the original version of this paper [OV], our instance-dependent commitment scheme inherited a certain "1-out-of-2" binding property from [NV] and [NOV]. This property is weaker and more complicated than the standard binding

[4] Ostrovky's theorem is only stated in terms of statistical zero-knowledge proofs, but it immediately extends to arguments.

[5] Informally, instance-dependent commitment schemes for a problem Π are commitment schemes where the hiding and binding properties are required to hold only on the yes and no instances of Π, respectively. A formal definition is given in Sect. 2.1.

property of commitments, but sufficed for establishing our main theorems (Theorems 2 and 3). Subsequently, the results of [NV] and [NOV] have been improved to yield standard-binding commitments, the latter by Haitner and Reingold [HR] and the former by [HORV]. Thus in this version, we use standard-binding instance-dependent commitments, as it simplifies our presentation.

2 Preliminaries

If X is a random variable taking values in a finite set $\mathcal{U}$, then we write $x \leftarrow X$ to indicate that x is selected according to X. If S is a subset of $\mathcal{U}$, then $x \leftarrow S$ means that x is selected according to the uniform distribution on S. We adopt the convention that when the same random variable occurs several times in an expression, they refer to a single sample. For example, $\Pr[f(X) = X]$ is defined to be the probability that when $x \leftarrow X$, we have $f(x) = x$. We write U_n to denote the random variable distributed uniformly over $\{0,1\}^n$.

A function $\varepsilon : \mathbb{N} \rightarrow [0,1]$ is called *negligible* if $\varepsilon(n) = n^{-\omega(1)}$. We let $\operatorname{neg}(n)$ denote an arbitrary negligible function (i.e., when we say that $f(n) < \operatorname{neg}(n)$ we mean that *there exists* a negligible function $\varepsilon(n)$ such that for every n, $f(n) < \varepsilon(n)$). Likewise, $\operatorname{poly}(n)$ denotes an arbitrary function $f(n) = n^{O(1)}$.

PPT refers to probabilistic algorithms (i.e., Turing machines) that run in *strict* polynomial time. A *nonuniform* PPT algorithm is a pair $(A, \bar{z})$, where $\bar{z} = z_1, z_2, \ldots$ is an infinite sequence of strings where $|z_n| = \operatorname{poly}(n)$, and A is a PPT algorithm that receives pairs of inputs of the form $(x, z_{|x|})$. (The string z_n is the called the *advice string* for A for inputs of length n.) Nonuniform PPT algorithms are equivalent to (nonuniform) families of polynomial-sized Boolean circuits.

Statistical Difference. The *statistical difference* (a.k.a. *variation distance*) between random variables X and Y taking values in $\mathcal{U}$ is defined to be $\Delta(X,Y) = \max_{S \subset \mathcal{U}} |\Pr[X \in S] - \Pr[Y \in S]|$. We say that X and Y are *ε-close* if $\Delta(X,Y) \leq \varepsilon$. Conversely, we say that X and Y are *ε-far* if $\Delta(X,Y) > \varepsilon$. For basic facts about this metric, see [SV, Sec 2.3].

Promise problems. Roughly speaking, a *promise problem* [ESY] is a decision problem where some inputs are excluded. Formally, a promise problem is specified by two disjoint sets of strings $\Pi = (\Pi_{\mathrm{Y}}, \Pi_{\mathrm{N}})$, where we call Π_{Y} the set of YES *instances* and Π_{N} the set of NO *instances*. Such a promise problem is associated with the following computational problem: given an input that is "promised" to lie in $\Pi_{\mathrm{Y}} \cup \Pi_{\mathrm{N}}$, decide whether it is in Π_{Y} or in Π_{N}. Note that languages are a special case of promise problems (namely, a language L over alphabet Σ corresponds to the promise problem $(L, \Sigma^* \setminus L)$). Thus working with promise problems makes our results more general. Moreover, even to prove our results just for languages, it turns out to be extremely useful to work with promise problems along the way. We refer the reader to the recent survey of Goldreich [Gol2] for more on the utility and subtleties of promise problems.

2.1 Instance-Dependent Cryptographic Primitives

It will be very useful for us to work with cryptographic primitives that may depend on an instance x of a problem $\Pi = (\Pi_{\mathrm{Y}}, \Pi_{\mathrm{N}})$, and where the security condition will hold only if x is in some particular set $I \subseteq \{0,1\}^*$. Indeed, recall that the SZKP–OWF Condition (Definition 1) refers to such a variant of of one-way functions, as captured by Definition 3 below.

Instance-Dependent One-Way Functions. To define instance-dependent one-way functions, we will need to define what it means for a function to be *instance dependent.*

Definition 2. *An* instance-dependent function *is a family* $\mathcal{F} = \{f_x \colon \{0,1\}^{n(|x|)} \to \{0,1\}^{m(|x|)}\}_{x\in\{0,1\}^*}$, *where* $n(\cdot)$ *and* $m(\cdot)$ *are polynomials. We call* $\mathcal{F}$ polynomial-time computable *if there is a deterministic polynomial-time algorithm* F *such that for every* $x \in \{0,1\}^*$ *and* $y \in \{0,1\}^{n(|x|)}$*, we have* $F(x,y) = f_x(y)$.

To simplify notation, we often write $f_x \colon \{0,1\}^{n(|x|)} \to \{0,1\}^{m(|x|)}$ to mean the family $\{f_x \colon \{0,1\}^{n(|x|)} \to \{0,1\}^{m(|x|)}\}_{x\in\{0,1\}^*}$.

Definition 3 (Instance-Dependent One-Way Function). *For any set* $I \subseteq \{0,1\}^*$*, a polynomial-time computable instance-dependent function* $f_x \colon \{0,1\}^{n(|x|)} \to \{0,1\}^{m(|x|)}$ *is an* instance-dependent one-way function *on* I *if for every nonuniform PPT adversary* A*, there exists a negligible function* ε *such that for every* $x \in I$*,*

$$\Pr_{y \leftarrow \{0,1\}^{n(|x|)}} \left[A(x, f_x(y)) \in f_x^{-1}(f_x(y))\right] \le \varepsilon(|x|) \ .$$

Next we consider an instance-dependent variant of *distributionally one-way functions*, which are functions that are hard for PPT adversaries to invert in a distributional manner—that is, given y it is hard for PPT adversaries to output a random preimage $f^{-1}(y)$. The standard definition of distributionally one-way function is given by Impagliazzo and Luby [IL]; here we give the instance-dependent analogue.

Definition 4 (Instance-Dependent Distributionally One-Way Function). *For any set* $I \subseteq \{0,1\}^*$*, a polynomial-time computable instance-dependent function* $f_x \colon \{0,1\}^{n(|x|)} \to \{0,1\}^{m(|x|)}$ *is an* instance-dependent distributionally one-way function *on* I *if there exists a polynomial* $p(\cdot)$ *such that for every nonuniform PPT adversary* A*, the random variables* $(U_{n(|x|)}, f_x(U_{n(|x|)}))$ *and* $(A(f_x(U_{n(|x|)})), f_x(U_{n(|x|)}))$ *are* $1/p(|x|)$*-far for all sufficiently long* $x \in I$*.*

Asking to invert in a distributional manner is a stronger requirement that just finding a preimage, therefore distributionally one-way functions might seem weaker than one-way functions. However, Impagliazzo and Luby [IL] proved that they are in fact equivalent. Like almost all reductions between cryptographic primitives, this result immediately extends to the instance-dependent analogue (using the same proof).

Proposition 1 (based on [IL, Lemma 1]). *For every set $I \subseteq \{0,1\}^*$, there exists an instance-dependent one-way function on I if and only if there exists an instance-dependent distributionally one-way function on I.*

Indistinguishability of Instance-Dependent Ensembles. The notions of statistical and computational indistinguishability have instance-dependent analogues. But first, we define an instance-dependent analogue of probability ensembles.

Definition 5. *An* instance-dependent probability ensemble *is a collection of random variables $\{A_x\}_{x\in\{0,1\}^*}$, where A_x takes values in $\{0,1\}^{p(|x|)}$ for some polynomial p. We call such an ensemble* samplable *if there is a probabilistic polynomial-time algorithm M such that for every x, the output $M(x)$ is distributed according to A_x.*

Definition 6. *Two instance-dependent probability ensembles $\{A_x\}_{x\in\{0,1\}^*}$ and $\{B_x\}_{x\in\{0,1\}^*}$ are* computationally indistinguishable *on $I \subseteq \{0,1\}^*$ if for every nonuniform PPT D, there exists a negligible function ε such that for all $x \in I$,*

$$|\Pr[D(x, A_x) = 1] - \Pr[D(x, B_x) = 1]| \le \varepsilon(|x|) \ .$$

Similarly, we say that $\{A_x\}_{x\in\{0,1\}^*}$ and $\{B_x\}_{x\in\{0,1\}^*}$ are statistically indistinguishable *on $I \subseteq \{0,1\}^*$ if the above is required for all functions D, instead of only nonuniform PPT ones. Equivalently, $\{A_x\}_{x\in\{0,1\}^*}$ and $\{B_x\}_{x\in\{0,1\}^*}$ are statistically indistinguishable on I iff A_x and B_x are $\varepsilon(|x|)$-close for some negligible function ε and all $x \in I$. We write $\approx_c$ and $\approx_s$ to denote computational and statistical indistinguishability, respectively.*

Often, we will informally say "A_x and B_x are computationally indistinguishable when $x \in I$" to mean the ensembles $\{A_x\}_{x\in\{0,1\}^*}$ and $\{B_x\}_{x\in\{0,1\}^*}$ are computationally indistinguishable on I.

Instance-Dependent Commitment Schemes. Recall that a (standard) commitment scheme is a two-stage protocol between a sender and a receiver. In the first stage, called the *commit stage*, the sender "commits" to a private message m. In the second stage, called the *reveal stage*, the sender reveals m and "proves" that it was the message to which she committed in the first stage. We require two properties of commitment schemes. The *hiding* property says that the receiver learns nothing about m in the commit stage. The *binding* property says that after the commit stage, the sender is bound to a particular value of m; that is, she cannot successfully open the commitment to two different bits in the reveal stage. A commitment scheme is said to be *public coin* if the all messages from the receiver to the sender are random coin tosses.

Instance dependent analogues of commitments schemes are commitments schemes that are tailored specifically to a specific problem Π. More precisely, *instance-dependent commitment schemes* receive an instance x of the problem Π as auxiliary input, and are required to be hiding when $x \in \Pi_{\mathrm{Y}}$ and be binding when $x \in \Pi_{\mathrm{N}}$. Thus, they are a relaxation of standard commitment schemes, since we do not require that the hiding and binding properties hold at the same time. Nevertheless, this relaxation is still useful in constructing zero-knowledge protocols. The reason is that zero-knowledge protocols based on commitments (for example, the protocol of [GMW]) typically use only the hiding property in proving zero knowledge (which is required only when x is a YES instance) and use only the binding property in proving soundness (which is required only when x is a NO instance).

2.2 Interactive Protocols and Zero Knowledge

In general, we follow the standard definitions of *interactive protocols*, *interactive proofs* and *arguments*, and *zero-knowledge proofs* and *arguments*, as in [Gol1]. We provide informal definitions of completeness, soundness, and public coin properties of an interactive protocol (P, V) for a promise problem $\Pi = (\Pi_Y, \Pi_N)$; the reader is referred to [Gol1] for the formal definitions.

- The *completeness error* of (P, V) is the maximum probability of V rejecting when interacting with an honest prover P on an input $x \in \Pi_Y$; we usually insist that the completeness error of an interactive protocol be bounded by 1/3. We say that (P, V) has *perfect completeness* if it has zero completeness error; in other word, V always accepts with probability 1 when interacting with the honest prover P on every input $x \in \Pi_Y$.
- The *statistical [resp., computational] soundness error* of (P, V) is the probability of V accepting when interacting with any [resp., nonuniform PPT] adversarial prover P^* on input $x \in \Pi_N$. Protocol (P, V) is said to be a *proof [resp., argument] system* if it has statistical [resp., computational] soundness error bounded by 1/3.
- We say (P, V) is *public coin* if all the messages sent by verifier V to prover P are random coin tosses.

Informally, an interactive protocol is *zero knowledge* if the verifier "learns nothing" from interacting with the prover other than the fact that the assertion being proven is true. This guarantee of "learning nothing" is formalized by exhibiting a PPT algorithm, called a *simulator*, whose output is indistinguishable from the verifier's view of the interaction with the prover. (Unlike the verifier, the simulator does not have access to the prover.) Intuitively, the verifier learns nothing because it could run the simulator instead of interacting with the prover. There are various notions of zero knowledge, referring to how rich a class of verifier strategies are considered. We informally describe them as follows:

- *Honest-verifier zero knowledge* refers to interactive protocols where there exists a PPT simulator for the verifier that follows the prescribed (honest) strategy.[6] This is the weakest formulation of zero knowledge, but it is already a nontrivial and interesting notion.

[6] This is an instantiation of what is called an "honest-but-curious adversary" or "passive adversary" in the literature on cryptographic protocols.

- *Auxiliary-input zero knowledge* or just *zero knowledge* refers to interactive protocols where for every (nonuniform PPT) verifier V^*, even one that deviates from the prescribed strategy, there exists a PPT simulator that simulates the view of V^* in the interaction with the prover.
- *Black-box zero knowledge* refers to zero knowledge protocols where the zero knowledge property is established by exhibiting a single, universal simulator that simulates an arbitrary verifier strategy V^* by using V^* as a subroutine. In other words, the simulator does not depend on or use the code of V^* (or its auxiliary input), and instead only requires black-box access to V^*.

The complexity classes that we use are defined as follows:

- **IP** denotes the class of promise problems possessing interactive proof systems.
- **HV-SZKP** and **HV-CZKP** denote the classes of promise problems having honest-verifier statistical and computational zero-knowledge proofs, respectively. Analogously, **HV-SZKA** and **HV-CZKA** denote the classes of promise problems having honest-verifier statistical and computational zero-knowledge *arguments*, respectively.
- **SZKP** and **CZKP** are the classes of promise problems possessing statistical and computational (auxiliary-input) zero-knowledge proofs, respectively. Analogously, **SZKA** and **CZKA** are the classes of promise problems possessing statistical and computational (auxiliary-input) zero-knowledge *arguments*, respectively.

We highlight the following points:

1. (Proof vs. argument systems) Interactive argument systems refer to protocols whose *soundness* condition is *computational*. That is, only nonuniform PPT cheating provers are guaranteed not to be able to convince the verifier of false statements except with probability 1/3; this is a weaker condition than proof systems, where the soundness condition is required of all cheating provers instead of just nonuniform PPT ones. Hence, we say that proof systems have *statistical soundness*.
2. (Prover complexity) In interactive proofs and interactive arguments, and in their zero-knowledge analogues, we allow the honest prover to be computationally unbounded, unless we specify *efficient prover*, which means a polynomial-time honest prover strategy given a witness for membership. It was shown in [NV] that for problems in **NP**, any zero-knowledge *proof* system with an unbounded prover can be transformed into one with an efficient prover; we will show the same for *argument* systems.

3 Unconditional Characterizations of Zero Knowledge

In this section, we provide *unconditional* characterizations of zero knowledge that would among other things allow us to establish our Symmetry Theorem between computational zero knowledge and computational soundness (Theorem 2). We first present our main characterization theorems in Sect. 3.1, which expands upon Theorem 3. The steps involved in proving these characterization theorems are outlined in Sect. 3.2, and lemmas needed to establish these theorems are given in Sects. 3.3, 3.4, and 3.5.

3.1 Our Main Characterization Theorems

In this subsection, we elaborate upon the SZKP–OWF Characterization of Zero Knowledge Theorem (Theorem 3). Specifically, we state four theorems giving a variety of equivalent characterizations of the classes **SZKP**, **CZKP**, **CZKA**, and **SZKA**. The ones for zero-knowledge arguments, namely **CZKA** and **SZKA**, are new; the other for zero-knowledge proofs, namely **CZKP** and **SZKP**, contain results from previous work, but are given for comparison. In addition to establishing Theorem 3 (and hence Theorem 2), these theorems show an equivalence between problems having only honest-verifier zero-knowledge protocols, problems satisfying the SZKP–OWF Condition, and problems with (malicious-verifier) zero-knowledge protocols having desirable properties like an efficient prover, perfect completeness, public coins, and black-box simulation. We note that these characterizations refer only to the classes of problems, and do not necessarily preserve other efficiency measures like round complexity, unless explicitly mentioned.

The following two previously known theorems give unconditional characterizations of zero-knowledge *proofs*.

Theorem 4 (SZKP Characterization Theorem [Oka, GSV, NV, HORV]). *For every problem $\Pi \in$ **IP**, the following conditions are equivalent.*

1. $\Pi \in$ **HV-SZKP**.
2. Π *satisfies the* SZKP–OWF Condition *without* OWF *instances.*
3. Π *has an instance-dependent commitment scheme that is statistically hiding on the* YES *instances and statistically binding on the* NO *instances. Moreover, the scheme is public coin.*
4. $\Pi \in$ **SZKP**, *and the statistical zero-knowledge proof system for* Π *has a black-box simulator, is public coin, and has perfect completeness. Furthermore, if* $\Pi \in$ **NP**, *the proof system has an efficient prover.*

Theorem 5 (CZKP Characterization Theorem [Vad, NV, HORV]). *For every problem $\Pi \in$ **IP**, the following conditions are equivalent.*

1. $\Pi \in$ **HV-CZKP**.
2. Π *satisfies the* SZKP–OWF Condition *without* OWF NO *instances.*
3. Π *has an instance-dependent commitment scheme that is computationally hiding on the* YES *instances and statistically binding on the* NO *instances. Moreover, the scheme is public coin.*
4. $\Pi \in$ **CZKP**, *and the computational zero-knowledge proof system for* Π *has a black-box simulator, is public coin, and has perfect completeness. Furthermore, if* $\Pi \in$ **NP**, *the proof system has an efficient prover.*

We give analogous characterizations for zero-knowledge *arguments*.

Theorem 6 (SZKA Characterization Theorem). *For every problem* $\Pi \in$ **NP***, the following conditions are equivalent.*

1. $\Pi \in$ **HV-SZKA**.
2. Π *satisfies the* SZKP–OWF Condition *without* owf yes *instances.*
3. Π *has an instance-dependent commitment scheme that is statistically hiding on the* yes *instances and computationally binding on the* no *instances. Moreover, the scheme is public coin.*
4. $\Pi \in$ **SZKA***, and the statistical zero-knowledge argument system for* Π *has a black-box simulator, is public coin, has perfect completeness, and an efficient prover.*

Theorem 7 (CZKA Characterization Theorem). *For every problem* $\Pi \in$ **NP***, the following conditions are equivalent.*

1. $\Pi \in$ **HV-CZKA**.
2. Π *satisfies the* SZKP–OWF Condition.
3. Π *has an instance-dependent commitment scheme that is computationally hiding on the* yes *instances and computationally binding on the* no *instances. Moreover, the scheme is public coin.*
4. $\Pi \in$ **CZKA***, and the computational zero-knowledge proof system for* Π *has a black-box simulator, is public coin, has perfect completeness, and an efficient prover.*

We prove Theorems 6 and 7 using lemmas established in Sections 3.3, 3.4, and 3.5. Notice that in these theorems involving zero knowledge arguments, we have restricted Π to be in **NP** in contrast to the theorems involving zero-knowledge proofs (Theorems 4 and 5), which are naturally restricted to **IP**. The reason for this is that argument systems are mainly interesting when the honest prover runs in polynomial time given a witness for membership (otherwise the protocol would not even be sound against prover strategies with the same resources as the honest prover), and such efficient provers only make sense for problems in **NP** (or actually, **MA**, to which our results generalize easily). In fact our theorems above show that for problems in **NP**, a zero-knowledge protocol without an efficient prover can be converted into one with an efficient prover (by the equivalence of Items 1 and 4 in Theorems 4 to 6 above).

3.2 Steps of Our Proof

Having stated our main characterization theorems in the previous subsection, we now provide an outline of the steps involved in establishing these characterization theorems:

1. We show that every problem Π possessing a (honest-verifier) zero-knowledge protocol satisfies the SZKP–OWF Condition. Depending on the zero knowledge and soundness guarantee, the types of SZKP–OWF Condition that Π satisfies will differ (in whether the sets of owf yes instances and owf no instances are empty or nonempty). This extends the unconditional characterization work of [Vad] for zero-knowledge proof systems to the more general zero-knowledge argument systems, and is in Section 3.3.

2. Next, we show that every problem Π satisfying the SZKP–OWF Condition yields an *instance-dependent commitment scheme* for Π. This is based on the techniques of [NOV, NV, HR, HORV], and is in Section 3.4.
3. Finally, we show that every problem $\Pi \in \mathbf{NP}$ having instance-dependent commitments allow us to construct zero-knowledge argument systems for Π with desirable properties like perfect completeness, black-box zero knowledge, public coins, and an efficient prover. This is done by substituting instance-dependent commitments for standard (non-instance-dependent) commitments used in existing zero-knowledge protocols like the Goldreich–Micali–Wigderson [GMW] zero-knowledge protocol for **NP**, and is in Section 3.5.

3.3 From Zero-Knowledge Protocols to SZKP–OWF Characterizations

In this subsection, we show that problems possessing (honest verifier) zero-knowledge arguments satisfy the SZKP–OWF Condition. Specifically, we prove that for every problem Π having a zero-knowledge argument also satisfies the SZKP–OWF Condition. This involving establishing a set of instances $I \subseteq \Pi_{\mathrm{Y}} \cup \Pi_{\mathrm{N}}$ such that $(\Pi_{\mathrm{Y}} \setminus I, \Pi_{\mathrm{N}} \setminus I) \in \mathbf{SZKP}$, and from which instance-dependent one-way functions can be constructed. The main difference from [Vad] is that [Vad] characterizes only zero-knowledge proofs and has no owf no instances, namely $I \cap \Pi_{\mathrm{N}} = \emptyset$. In other words, the characterizations of [Vad] satisfy the SZKP–OWF Condition without owf no instances.

Lemma 1. *If problem* $\Pi \in \mathbf{HV\text{-}CZKA}$*, then* Π *satisfies the* SZKP–OWF Condition*. In addition, if* $\Pi \in \mathbf{HV\text{-}SZKA}$*, then* Π *satisfies the* SZKP–OWF Condition *without* owf yes *instances, namely* $I \cap \Pi_{\mathrm{Y}} = \emptyset$*.*

Proof Idea. To show that $\Pi \in \mathbf{HV\text{-}CZKA}$ satisfies the SZKP–OWF Condition, we will need to establish a set I with $(\Pi_{\mathrm{Y}} \setminus I, \Pi_{\mathrm{N}} \setminus I) \in \mathbf{SZKP}$, and construct an instance-dependent one-way on I. We will do a separate analysis for the yes and no instances, and therefore we first show how to define sets $I_Y \subseteq \Pi_{\mathrm{Y}}$ and $I_N \subseteq \Pi_{\mathrm{N}}$ such that instance-dependent one-way functions can be constructed on these sets, and that $(\Pi_{\mathrm{Y}} \setminus I_Y, \Pi_{\mathrm{N}} \setminus I_N) \in \mathbf{SZKP}$. Having two (different) instance-dependent one-way functions f_x and g_x on I_Y and I_N, respectively, we construct a single instance-dependent one-way function on $I \stackrel{\mathrm{def}}{=} I_Y \cup I_N$ by concatenating the functions f_x and g_x.

Next, we describe, on an intuitive level, how to define the sets $I_Y \subseteq \Pi_{\mathrm{Y}}$ and $I_N \subseteq \Pi_{\mathrm{N}}$. Fix an instance x of the problem $\Pi \in \mathbf{HV\text{-}CZKA}$. From the simulator S on input x, we consider a simulation-based prover P_S and a simulation-based verifier V_S. On a high level, P_S replies with the same conditional probability as the prover in the output of S, and V_S sends its messages with the same conditional probability as the verifier in the output of S. We make the following observations:

1. The interaction between P_S and V_S is identical to the output of the simulator S, on every x.
2. By the zero-knowledge condition, we have that $\langle P_S, V_S \rangle$ is computationally indistinguishable from $\langle P, V \rangle$, when $x \in \Pi_{\mathrm{Y}}$.
3. By assuming, without loss of generality, that the simulator always outputs accepting transcripts, it holds that P_S makes V_S accepts with probability 1, on every x.

We consider a statistical measure of how "similar" V_S is to V (on instance x, when interacting with simulation-based prover P_S). Using this statistical measure (given in the full proof below), we define sets I_Y and I_N as follows:

- I_Y contains instances $x \in \Pi_{\mathrm{Y}}$ for which V_S is *statistically different* from V.
- I_N contains instances $x \in \Pi_{\mathrm{N}}$ for which V_S is *statistically similar* to V.

Now the proof that this gives a SZKP–OWF Condition proceed as follows:

1. On I_Y, we have that V_S is statistically different from V. Nevertheless, by the zero-knowledge condition (as noted above), V_S is computationally similar to V. This enables us to construct one-way functions for instances in I_Y, as shown in [Vad].
2. On I_N, we have that V_S is statistically similar to V. Combining this with the fact that P_S will always convince V_S to accept (as noted above), we conclude that P_S convinces V to accept with high probability. By computational soundness of (P, V), it must be the case that P_S is not PPT. Using techniques from Ostrovsky [Ost], this allows us to convert the simulator S into an instance-dependent distributional one-way function g_x.[7] Then by Proposition 1, due to Impagliazzo and Luby [IL], we can obtain an instance-dependent one-way function from g_x.
3. To see that $(\Pi_{\mathrm{Y}} \setminus I_Y, \Pi_{\mathrm{N}} \setminus I_N) \in \mathbf{SZKP}$, we observe the following: For those YES instances not in I_Y—that is, instances in $\Pi_{\mathrm{Y}} \setminus I_Y$—the simulated verifier V_S is statistically similar to V. And for those NO instances not in I_N—that is, instances in $\Pi_{\mathrm{N}} \setminus I_N$—the simulated verifier V_S is statistically different from V. This gap in the statistical properties allows us to reduce promise problem $(\Pi_{\mathrm{Y}} \setminus I_Y, \Pi_{\mathrm{N}} \setminus I_N)$ to one of the complete problems for **SZKP** [SV, GV, Vad].

Proof of Lemma 1. Let (P, V) be a zero-knowledge argument system for Π, with simulator S. We now proceed as in the proof of [Vad] and modify our interactive protocol (P, V) to satisfy the following (standard) additional properties.

- The completeness error $c(|x|)$ and soundness error $s(|x|)$ are both negligible. This can be achieved by standard error reduction via (sequential) repetition.

[7] If g_x is not distributionally one-way, then P_S can be made to be efficient, hence contradicting the computational soundness of (P, V). Interestingly, Ostrovsky [Ost] uses the assumption that g_x is not distributionally one-way to invert the simulator S on the YES instances, and conclude that Π is not "hard-on-average". Although we use similar techniques as [Ost], we instead invert S on the NO instances to contradict the computational soundness of (P, V).

- On every input x, the two parties exchange $2\ell(|x|)$ messages for some polynomial ℓ, with the verifier sending even-numbered messages and sending all of its $r(|x|)$ random coin tosses in the last message. (Without loss of generality, we may assume that $r(|x|) \geq |x|$.) Having the verifier send its coin tosses at the end does not affect soundness because it is after the prover's last message, and does not affect honest-verifier zero knowledge because the simulator is anyhow required to simulate the verifier's coin tosses.
- On every input x, the simulator always outputs *accepting transcripts*, where we call a sequence τ of 2ℓ messages an accepting transcript on x if all of the verifier's messages are consistent with its coin tosses (as specified in the last message), and the verifier would accept in such an interaction.

For a transcript τ, we denote by τ_i the *prefix* of τ consisting of the first i messages. For readability, we often drop the input x from the notation, for instance using $\ell = \ell(|x|)$, $\langle P, V\rangle = \langle P, V\rangle(x)$, $r = r(|x|)$, and so forth. Thus, in what follows, $\langle P, V\rangle_i$ and S_i are random variables representing prefixes of transcripts generated by the real interaction and simulator, respectively, on a specified input x.

We define the *simulation-based prover*, denoted as $P_S(x)$, as follows: Given an execution prefix τ_{2i}, for $i = 1, 2, \ldots, \ell - 1$, prover P_S responses as follows.

1. If simulator $S(x)$ outputs a transcript that begins with τ_{2i} with probability 0, then P_S replies with a dummy message.
2. Otherwise, P_S replies according with the same conditional probability as the prover in the output of the simulator. That is, it replies with a string β with probability $p_\beta = \Pr\left[S(x)_{2i+1} = \tau_{2i} \circ \beta | S(x)_{2i} = \tau_{2i}\right]$.

Following [AH, PT, GV, Vad], we consider the following quantity:

$$h(x) = \sum_{i=1}^{\ell} \left[\mathrm{H}(S(x)_{2i}) - \mathrm{H}(S(x)_{2i-1})\right] \quad , \tag{1}$$

where $\mathrm{H}(\cdot)$ denotes the *(Shannon) entropy* measure, which is given by $\mathrm{H}(X) = \mathrm{E}_{x \leftarrow X}\left[\log(1/\Pr[X = x])\right]$.

Now, we define the sets I_Y and I_N as follows:

$$I_Y = \{x \in \Pi_\mathrm{Y} : h(x) < r(|x|) - 1/q(|x|)\} \ ;$$
$$I_N = \{x \in \Pi_\mathrm{N} : h(x) > r(|x|) - 2/q(|x|)\} \ ,$$

where the polynomial $q(|x|) = 256 \cdot \ell(|x|)$.

Having defined sets I_Y and I_N, Lemma 1 is established by the following claims, where the first three are established using techniques in [Vad].

Claim 1. *Problem* $(\Pi_\mathrm{Y} \setminus I_Y, \Pi_\mathrm{N} \setminus I_N) \in$ **SZKP**.

Claim 2. *There exists an instance-dependent one-way function on* I_Y.

Claim 3. *For* $\Pi \in$ **HV-SZKA***, we can take* $I_Y = \emptyset$.

The main novelty in our analysis is the following claim.

Claim 4. *There exists an instance-dependent one-way function on I_N.*

Proof of Claim. To get an instance-dependent one-way function on I_N, we use the following idea of Ostrovsky [Ost]: If we can invert the simulator, then P_S's replies can be approximated efficiently. By the computational soundness of (P, V), this is impossible, so the simulator must be a one-way function. More precisely, we define the function g_x, whose purpose is to output messages of the simulator, as follows:

$$g_x(i, \omega) = (x, i, S(x; \omega)_{2i}) \ . \tag{2}$$

Note that g_x is polynomial-time computable because the simulator S runs in polynomial time. If g_x is *not* distributionally one-way (in the sense of Definition 4), then we can devise an efficient cheating prover strategy, call it $\widetilde{P}$, that *efficiently* "simulates" our simulation-based prover P_S upto negligible statistical error. The way to do this is to feed a given transcript prefix τ_{2i} after the verifier has responded in round $2i$, into the inversion algorithm of g_x to obtain the simulation-based prover response for round $2i+1$. In doing so, we contradict the computational soundness property of (P, V). This argument is captured by following proposition.

Proposition 2 (based on [Ost, Lemma 1]). [8] *Let g_x be as in* (2). *For every set $K \subseteq \{0,1\}^*$, if g_x is* not *an instance-dependent distributionally one-way function on K, then for every polynomial p, there exists a nonuniform PPT prover $\widetilde{P}$ such that*

$$\Delta(\langle \widetilde{P}, V\rangle(x), S(x)) \leq \ell(|x|) \cdot \left(\frac{1}{p(|x|)} + 2 \cdot \Delta(\langle P_S, V\rangle(x), S(x)) \right) \ ,$$

for infinitely many $x \in K$.

This leaves us to upper bound $\Delta(\langle P_S, V\rangle, S)$ in order to obtain an upper bound on $\Delta(\langle \widetilde{P}, V\rangle, S)$, and hence contradict the computational soundness of V (because S always outputs accepting transcripts). Recall that for every $x \in I_N$, we have $h > r - 2/q$. From [AH, PT, GV], we know that $h = r - \mathrm{KL}(\langle P_S, V\rangle, S)$, where KL is the *Kullback-Leibler* distance defined as $\mathrm{KL}(X, Y) = \mathrm{E}_{\alpha \leftarrow X}\left[\log(\Pr[X = \alpha]) - \log(\Pr[Y = \alpha])\right]$. (See [GV, Lemma 2.2].) Hence, we get $\mathrm{KL}(\langle P_S, V\rangle, S) < 2/q$. Using the fact that for any random variables X and Y, $\mathrm{KL}(X, Y) \geq (1/2) \cdot (\Delta(X, Y))^2$ [CT, Lemma 12.6.1], we get that for all $x \in I_N$,

$$\Delta(\langle P_S, V\rangle, S) < 2/\sqrt{q} = 1/(8 \cdot \ell) \ , \tag{3}$$

since $q = 256 \cdot \ell$.

[8] As pointed out to us by Lilach Bien, the statement and application of this proposition in the original version of our paper [OV] erroneously neglected the dependence on $\Delta(\langle P_S, V\rangle(x), S(x))$.

Now by Proposition 2, if g_x is not distributionally one-way on I_N, we can take $I_N = K$ and choose $p(|x|) = 4 \cdot \ell(|x|)$, to get a nonuniform PPT $\widetilde{P}$ such that

$$\begin{aligned} \Delta(\langle \widetilde{P}, V\rangle, S) &\leq \ell \cdot (1/p + 2 \cdot \Delta(\langle P_S, V\rangle, S)) \\ &= 1/4 + 2 \cdot \ell \cdot \Delta(\langle P_S, V\rangle, S) \\ &< 1/2 \ . \qquad \text{(by (3))} \end{aligned}$$

And since the simulator S always produce accepting transcripts, we have

$$\Pr[(\widetilde{P}, V)(x) = \texttt{accept}] \geq 1/2 \ ,$$

for infinitely many $x \in I_N$. This contradicts the computational soundness of (P, V). Therefore, g_x must be a distributionally one-way function on I_N. By Proposition 1 (due to Impagliazzo and Luby [IL]), g_x can be converted into an instance-dependent (standard) one-way function on I_N, as desired. □

Let us see how the above five claims establish Lemma 1. Define set $I = I_Y \cup I_N$. This means that the promise problem $(\Pi_\mathrm{Y} \setminus I, \Pi_\mathrm{N} \setminus I) = (\Pi_\mathrm{Y} \setminus I_Y, \Pi_\mathrm{N} \setminus I_N)$, and Claim 1 places this problem in **SZKP**. Claims 2 and 4 give us instance-dependent one-way functions on I_Y and I_N, respectively; to obtain a single instance-dependent one-way function on $I = I_Y \cup I_N$, we use the following claim.

Claim 5. *For any sets $J, K \subseteq \{0,1\}^*$, if there exist instance-dependent one-way functions on J and there exist instance-dependent one-way functions on K, then there exist instance-dependent one-way functions on $J \cup K$.*

Therefore, by Claim 5 above, we know that $\Pi \in$ **HV-CZKA** satisfies the SZKP–OWF Condition. Furthermore, if $\Pi \in$ **HV-SZKA**, Claim 3 tells us that $I_Y = \emptyset$, and hence $I \cap \Pi_\mathrm{Y} = I_Y = \emptyset$, giving us that Π satisfies the SZKP–OWF Condition without owf yes instances. □

3.4 From SZKP–OWF Characterization to Instance-Dependent Commitment Schemes

In this subsection, we show that every problem Π satisfying the SZKP–OWF Condition yields an instance-dependent commitment scheme for Π. This is obtained by combining statistically-binding commitments from one-way functions [Nao, HILL], statistically-hiding commitments from one-way functions [NOV, HR], and instance-dependent commitments for **SZKP** [NV, HORV]. In the original version of this paper [OV], our instance-dependent commitment scheme inherited a certain “1-out-of-2” binding property from [NV, NOV]. This property is weaker and more complicated than the standard binding property of commitments, but sufficed for establishing our main theorems (Theorems 2 and 3). Due to improvements by [HR, HORV], it is now possible to construct instance-dependent commitments with the standard binding property, and hence we use standard-binding commitments to simplify our presentation.

Lemma 2. *The following conditions hold for problems* Π *satisfying the* SZKP–OWF Condition.

- *If* Π *satisfies the* SZKP–OWF Condition *without* owf no *instances [resp., without* owf *instances], then it has an instance-dependent commitment scheme that is computationally [resp., statistically] hiding on the* yes *instances and statistically binding on the* no *instances.*
- *If* Π *satisfies the* SZKP–OWF Condition *[resp., without* owf yes *instances], then it has an instance-dependent commitment scheme that is computationally [resp., statistically] hiding on the* yes *instances and computationally binding on the* no *instances.*

Furthermore, all the above instance-dependent commitment schemes are public coin.

The proof of Lemma 2, tying together all the following propositions and claims, is given at the end of this subsection. Before stating our propositions and claims, we provide an outline of what we intend to construct in the next paragraph.

Given that problem Π satisfies the SZKP–OWF Condition, we let the set of owf yes instances be denoted as $I_Y = I \cap \Pi_{\mathrm{Y}}$, and the set of owf no instances be denoted as $I_N = I \cap \Pi_{\mathrm{N}}$. Our task of constructing an instance-dependent commitment scheme for Π is broken into following four steps: (1) construct an instance-dependent commitment scheme for the problem $(\Pi_{\mathrm{Y}} \setminus I, \Pi_{\mathrm{N}} \setminus I) \in \mathbf{SZKP}$, (2) construct an instance-dependent commitment scheme for the problem $(I_Y, \overline{I_Y})$, (3) construct an instance-dependent commitment scheme for the problem $(\overline{I_N}, I_N)$, and (4) combine all these three instance-dependent commitment schemes into a single instance-dependent commitment scheme for Π. We will explain why these four steps yield an instance-dependent commitment scheme for Π in the proof of Lemma 2, given at the end of this subsection.

Step 1: The instance-dependent commitment for the problem $(\Pi_{\mathrm{Y}} \setminus I_Y, \Pi_{\mathrm{N}} \setminus I_N) \in \mathbf{SZKP}$ follows from [HORV] (which builds on [NV]).

Proposition 3 ([HORV]). *For any problem* $\Gamma \in \mathbf{SZKP}$*, problem* Γ *has an instance-dependent commitment scheme that is statistically hiding on the* yes *instances and statistically binding on the* no *instances. Moreover, the instance-dependent commitment scheme obtained is public coin.*

Step 2: Notice that the instance-dependent commitments given by the above proposition do not guarantee hiding or binding properties on the owf instances sets I_Y and I_N. Nevertheless, we noted in [Vad], we can use the instance-dependent one-way functions on I_Y to construct instance-dependent commitment schemes that are computationally hiding on I_Y and statistically binding elsewhere, based on Naor's [Nao] commitment scheme. This is because Naor's scheme can be based on any one-way function [HILL], and the statistical binding property of the scheme does not depend on the one-way security of the function.

Proposition 4 (based on [Nao, HILL]). *For every set $K \subseteq \{0,1\}^*$, if there is an instance-dependent one-way function on K, then problem $(K, \overline{K})$ has an instance-dependent commitment scheme that is computationally hiding on the* YES *instances (namely, instances in K), and statistically binding on the* NO *instances (namely, instances in $\overline{K}$). Moreover, the instance-dependent commitment scheme obtained is public coin.*

Step 3: We construct instance-dependent commitment schemes that are computationally binding on I_N and statistically hiding elsewhere, based on the fact that statistically hiding and computationally binding commitments can be constructed from any one-way function [NOV, HR].

Proposition 5 (based on [NOV, HR]). *For every set $K \subseteq \{0,1\}^*$, if there is an instance-dependent one-way function on K, then problem $(\overline{K}, K)$ has an instance-dependent commitment that is statistically hiding on the* YES *instances (namely, instances in $\overline{K}$), and computationally binding on the* NO *instances (namely, instances in K). Moreover, the instance-dependent commitment scheme obtained is public coin.*

Step 4: Finally, we use standard methods to combine the three instance-dependent commitment schemes that we have constructed into a single instance-dependent commitment scheme for Π. The first method gives a combined scheme for the intersection of two problems.

Claim 6. *Suppose problems $\Gamma = (\Gamma_{\mathrm{Y}}, \Gamma_{\mathrm{N}})$ and $\Gamma' = (\Gamma'_{\mathrm{Y}}, \Gamma'_{\mathrm{N}})$ have instance-dependent commitment schemes Com_x and Com'_x, respectively. Then problem $\Gamma \cap \Gamma' = (\Gamma_{\mathrm{Y}} \cap \Gamma'_{\mathrm{Y}}, \Gamma_{\mathrm{N}} \cup \Gamma'_{\mathrm{N}})$ has an instance-dependent commitment scheme Com''_x with the following properties:*

– Com''_x *is statistically [resp., computationally] hiding if both* Com_x *and* Com'_x *are statistically [resp., computationally] hiding.*
– Com''_x *is statistically [resp., computationally] binding if either of* Com_x *or* Com'_x *is statistically [resp., computationally] binding.*
– Com''_x *is public coin if both* Com_x *and* Com'_x *are public coin.*

Proof. In commitment scheme Com''_x, the sender commits to b by committing to b in both schemes Com_x and Com'_x, with the execution of both schemes done in parallel. The claimed properties of Com''_x follow by inspection. □

The second method provides a combined scheme for the union of two problems.

Claim 7. *Suppose problems $\Gamma = (\Gamma_{\mathrm{Y}}, \Gamma_{\mathrm{N}})$ and $\Gamma' = (\Gamma'_{\mathrm{Y}}, \Gamma'_{\mathrm{N}})$ have instance-dependent commitment schemes Com_x and Com'_x, respectively. Then problem $\Gamma \cup \Gamma' = (\Gamma_{\mathrm{Y}} \cup \Gamma'_{\mathrm{Y}}, \Gamma_{\mathrm{N}} \cap \Gamma'_{\mathrm{N}})$ has an instance-dependent commitment scheme Com''_x with the following properties:*

- Com''_x *is statistically [resp., computationally] hiding if either of* Com_x *or* Com'_x *is statistically [resp., computationally] hiding.*
- Com''_x *is statistically [resp., computationally] binding if both* Com_x *and* Com'_x *are statistically [resp., computationally] binding.*
- Com''_x *is public coin if both* Com_x *and* Com'_x *are public coin.*

Proof. In commitment scheme Com''_x, the sender on input bit b, first secret shares b into two shares, b_1 and b_2, with the property that $b_1 \oplus b_2 = b$ and each b_i is uniform in $\{0,1\}$. (This can be done by choosing a random $b_1 \leftarrow \{0,1\}$, and setting $b_2 = b_1 \oplus b$.) The sender then commits to b by committing to bits b_1 and b_2 in schemes Com_x and Com'_x, respectively. The execution of schemes Com_x and Com'_x is done in parallel.

The hiding property follows from the fact that bit b remains hidden as long as one of the bits b_1 or b_2 remains hidden. Then binding property follows from the fact that $b = b_1 \oplus b_2$, and hence b is bounded to a fixed value if both b_1 and b_2 are bounded to fixed values. The public coin property and round complexity of Com''_x follow by inspection. □

Having established the propositions and claims that we need, we now prove Lemma 2.

Proof of Lemma 2. Given that problem Π satisfies the SZKP–OWF CONDITION, let I be the set of OWF instances, and let the OWF YES instances be $I_Y = I \cap \Pi_{\mathrm{Y}}$ and the OWF NO instances be $I_N = I \cap \Pi_{\mathrm{N}}$. By Propositions 3, 4, and 5, we have three instance-dependent commitment schemes, call them $\mathsf{Com}^{(1)}_x$, $\mathsf{Com}^{(2)}_x$, and $\mathsf{Com}^{(3)}_x$, for the problems $(\Pi_{\mathrm{Y}} \setminus I, \Pi_{\mathrm{N}} \setminus I) \in \mathbf{SZKP}$, $(I_Y, \overline{I_Y})$, and $(\overline{I_N}, I_N)$, respectively. Moreover, all three schemes are public coin.

If Π satisfies the SZKP–OWF CONDITION without OWF instances, then set $I = \emptyset$, and hence $\mathsf{Com}^{(1)}_x$ suffices to be our instance-dependent commitment scheme for Π. If Π satisfies the SZKP–OWF CONDITION without OWF NO instances, then $I_N = I \cap \Pi_{\mathrm{N}} = \emptyset$. Consequently, we do not need scheme $\mathsf{Com}^{(3)}_x$, and can just combine schemes $\mathsf{Com}^{(1)}_x$ and $\mathsf{Com}^{(2)}_x$ in a manner prescribed by Claim 7 to get an instance-dependent commitment scheme for Π.

Analogously, if Π satisfies the SZKP–OWF CONDITION without OWF YES instances, then $I_Y = I \cap \Pi_{\mathrm{Y}} = \emptyset$. Consequently, we do not need scheme $\mathsf{Com}^{(2)}_x$, and can just combine schemes $\mathsf{Com}^{(1)}_x$ and $\mathsf{Com}^{(3)}_x$ in a manner prescribed by Claim 6 to get an instance-dependent commitment scheme for Π. Finally, if Π satisfies the SZKP–OWF CONDITION, we first combine schemes $\mathsf{Com}^{(1)}_x$ and $\mathsf{Com}^{(2)}_x$ in a manner prescribed by Claim 7 to get an instance-dependent commitment scheme for $(\Pi_{\mathrm{Y}}, \Pi_{\mathrm{N}} \setminus I_N)$, and then combine this scheme with $\mathsf{Com}^{(3)}_x$ in a manner prescribed by Claim 6 to get an instance-dependent commitment scheme for Π.

The hiding, binding, and public coin properties of the instance-dependent commitment scheme for Π follow by inspection. □

3.5 From Instance-Dependent Commitment Schemes to Zero-Knowledge Protocols

Having obtained instance-dependent commitments in the previous subsection, we now use these commitments to construct unconditional zero-knowledge protocols for problems $\Pi \in \mathbf{NP}$ having these instance-dependent commitments. We observe that the existing zero-knowledge protocols for **NP** require complexity assumptions because they use standard (non-instance dependent) commitments, and standard commitments are not known to exist unconditionally. Therefore, we can remove the complexity assumptions needed by substituting standard commitments for instance-dependent commitments in these existing protocols. Specifically, we do this substitution in the Goldreich–Micali–Wigderson [GMW] zero-knowledge protocol for **NP**.

Lemma 3 (based on [GMW]). *If problem* $\Pi \in \mathbf{NP}$ *has an instance-dependent commitment scheme* Com_x*, then it has a zero-knowledge protocol* (P, V) *with the following properties:*

- (P, V) *is statistical [resp., computational] zero knowledge if* Com_x *is statistically [resp., computationally] hiding on the* YES *instances. Moreover,* (P, V) *has a black-box simulator.*
- (P, V) *is a proof [resp., argument] system if* Com_x *is statistically [resp., computationally] binding on the* NO *instances.*
- (P, V) *has perfect completeness and has an efficient prover.*
- (P, V) *is public coin if* Com_x *is public coin.*

3.6 Putting It All Together

We now show how our lemmas in Sects. 3.3, 3.4, and 3.5 imply our main characterization theorems in Sect. 3.1.

Proof of Theorems 6 and 7. The implications for both theorems are captured by the same lemmas, so we can conveniently state them together.

(1) ⇒ **(2)** is established by Lemma 1.
(2) ⇒ **(3)** is established by Lemma 2.
(3) ⇒ **(4)** is established by Lemma 3. This is the only step that requires the problem $\Pi \in \mathbf{NP}$.
(4) ⇒ **(1)** follows directly from definition. □

Acknowledgements

We are grateful to Lilach Bien for pointing out an error in the statement and use of Proposition 2 in the original version of our paper [OV]. We thank Oded Goldreich and the anonymous *EUROCRYPT 2007* reviewers for their helpful comments. We also thank Iftach Haitner and Omer Reingold for allowing us to simplify the presentation of our results with their wonderful result [HR] and the follow-up [HORV].

References

[AH] Aiello, W., and Håstad, J. Statistical zero-knowledge languages can be recognized in two rounds. *J. Comput. Syst. Sci.*, 42(3):327–345, 1991.

[BCC] Brassard, G., Chaum, D., and Crépeau, C. Minimum disclosure proofs of knowledge. *J. Comput. Syst. Sci.*, 37(2):156–189, 1988.

[CT] Cover, T. M., and Thomas, J. A. *Elements of information theory.* Wiley-Interscience, New York, NY, USA, 2 edition, 2006.

[ESY] Even, S., Selman, A., and Yacobi, Y. The complexity of promise problems with applications to public-key cryptography. *Inform. Control*, 61(2):159–173, 1984.

[GMR] Goldwasser, S., Micali, S., and Rackoff, C. The knowledge complexity of interactive proof systems. *SIAM J. Comput.*, 18(1):186–208, 1989.

[GMW] Goldreich, O., Micali, S., and Wigderson, A. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *J. ACM*, 38(1):691–729, 1991.

[Gol1] Goldreich, O. *Foundations of Cryptography: Basic Tools.* Cambridge University Press, 2001.

[Gol2] Goldreich, O. On promise problems (a survey in memory of Shimon Even [1935-2004]). Technical Report TR05–018, ECCC, 2005.

[GSV] Goldreich, O., Sahai, A., and Vadhan, S. Honest verifier statistical zero-knowledge equals general statistical zero-knowledge. In *Proc. 30th STOC*, pages 399–408, 1998.

[GV] Goldreich, O., and Vadhan, S. Comparing entropies in statistical zero-knowledge with applications to the structure of SZK. In *Proc. 14th Comput. Complex.*, pages 54–73, 1999.

[HILL] Håstad, J., Impagliazzo, R., Levin, L., and Luby, M. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[HORV] Haitner, I., Ong, S., Reingold, O., and Vadhan, S. Instance-dependent commitments for statistical zero-knowledge proofs. In preparation, March 2007.

[HR] Haitner, I., and Reingold, O. Statistically-hiding commitment from any one-way function. Technical Report 2006/436, Cryptol. ePrint Arch., 2006.

[IL] Impagliazzo, R., and Luby, M. One-way functions are essential for complexity based cryptography. In *Proc. 30th FOCS*, pages 230–235, 1989.

[Nao] Naor, M. Bit commitment using pseudorandomness. *J. Cryptol.*, 4(2):151–158, 1991.

[NOV] Nguyen, M., Ong, S., and Vadhan, S. Statistical zero-knowledge arguments for NP from any one-way function. In *Proc. 47th FOCS*, pages 3–14, 2006.

[NV] Nguyen, M., and Vadhan, S. Zero knowledge with efficient provers. In *Proc. 38th STOC*, pages 287–295, 2006.

[Oka] Okamoto, T. On relationships between statistical zero-knowledge proofs. *J. Comput. Syst. Sci.*, 60(1):47–108, 2000.

[Ost] Ostrovsky, R. One-way functions, hard on average problems, and statistical zero-knowledge proofs. In *Proc. 6th Annual Structure in Complexity Theory Conference*, pages 133–138, 1991.

[OV] Ong, S., and Vadhan, S. Zero knowledge and soundness are symmetric. Technical Report TR06-139, ECCC, 2006.

[OW] Ostrovsky, R., and Wigderson, A. One-way functions are essential for non-trivial zero-knowledge. In *Proc. 2nd Israel Symposium on Theory of Computing Systems*, pages 3–17, 1993.

[PT] Petrank, E., and Tardos, G. On the knowledge complexity of NP. *Combinatorica*, 22(1):83–121, 2002.

[SV] Sahai, A., and Vadhan, S. A complete problem for statistical zero knowledge. *J. ACM*, 50(2):196–249, 2003.

[Vad] Vadhan, S. An unconditional study of computational zero knowledge. *SIAM J. Comput.*, 36(4):1160–1214, 2006.