Electronic Colloquium on Computational Complexity, Report No. 68 (2008)

Instance-Dependent Commitment Schemes and the Round Complexity of Perfect Zero-Knowledge Proofs

Lior Malka
Department of Computer Science
University of Victoria, BC, Canada
liorma@cs.uvic.ca

Abstract. We study the question whether the number of rounds in public-coin perfect zero-knowledge (**PZK**) proofs can be collapsed to a constant. Despite extensive research into the round complexity of interactive and zero-knowledge protocols, there is no indication how to address this question. Furthermore, the main tool to tackle this question is instance-dependent commitments, but currently such schemes are only statistically hiding, whereas we need perfectly hiding schemes.

We give the first *perfectly* hiding instance-dependent commitment scheme. This scheme can be constructed from any problem that has a **PZK** proof. We then show that obtaining such a scheme that is also *constant-round* is not only sufficient, but also necessary to collapse the number of rounds in **PZK** proofs. Hence, we show an equivalence between the tasks of obtaining the commitment, and collapsing the rounds. Our idea also yields an elegant equivalence between zero-knowledge and commitments.

In the second part of the paper we construct a non-interactive, perfectly hiding scheme whose binding property holds on all but an exponentially small fraction of the inputs. Informally, this shows that the rounds in public-coin **PZK** proofs can be collapsed if we can guarantee that the prover is not choosing its randomness from a small set. We formalize this condition using a preamble, which we then apply to some simple cases. An interesting consequence of independent interest is that we use the circuits from the study of **NIPZK** in the commitment scheme of Naor [39], and this leads to a new perfectly-hiding instance-dependent commitment for **NIPZK** problems with a small soundness error.

Key words: constant-round, perfect zero-knowledge, instance-dependent commitment schemes.

1 Introduction

Perfect zero-knowledge protocols allow a *prover* to prove an assertion to a *verifier*, but without leaking any information to the verifier but the validity of the assertion [27]. Unlike other variants of zero-knowledge protocols, which allow the prover to leak information, these protocols guarantee the highest level of privacy to the prover. Many problems that admit perfect zero-knowledge proofs, like QUADRATIC-RESIDUOUSITY [27] and DISCRETE-LOG [50], play a central role in cryptography, and they are used in key agreement, encryption schemes, digital signatures, and identification schemes (c.f., [15, 17, 19]). In addition to their value to cryptography, perfect zero-knowledge protocols are intriguing also from a complexity theoretic perspective. This is so because all the *known* problems admitting perfect zero-knowledge proofs, like GRAPH-ISOMORPHISM [22], are in **NP**, but not known to be **NP**-complete or in **P**. Furthermore, all of

ISSN 1433-8092

these problems admit 3-round public-coin **PZK** proofs, and have a very simple combinatorial characterization [29]. This raises fascinating questions. For example, is **PZK** contained in **NP**? Can we show that all the problems in **PZK** admit perfect zero-knowledge proofs with a constant-number of rounds?

Although many problems admit perfect zero-knowledge proofs, most of the research so far focused on statistical and computational zero-knowledge proofs (**SZK** and **CZK**, respectively). Unlike perfect zero-knowledge, these variants allow the prover to leak a small amount of information to the verifier. Thus, they can be studied with a wide variety of tools, even if these tools have a side effect that they cause the prover to leak information. Indeed, such tools were used to prove general results about statistical and computational zero-knowledge proofs, including complete problems, equivalence between public and private coins, equivalence between honest and malicious verifier, and the more recent result of Ong and Vadhan, which shows that **SZK** proofs have a constant number of rounds ([47, 52, 43, 25, 23, 45]). Unfortunately, these results do not apply to the perfect setting, and the lack of tools in the study of perfect zero-knowledge proofs makes them very difficult to study. Consequently, many basic questions about **PZK** remain open.

1.1 Motivation

In this paper we are interested in the question whether the number of rounds in perfect zero-knowledge proofs can be collapsed to a constant. That is, if a problem admits a public-coin **PZK** proof, does it also admit a constant-round **PZK** proof? This question is important because the number of rounds in a protocol is perhaps one of the its most expensive resources. Also, recall that all the *known* **PZK** proofs are public-coin and require 3-rounds. Thus, studying the round complexity of perfect zero-knowledge proofs may explain this phenomenon, or alternatively, lead to the discovery of a problem that admits a **PZK** proof with more than 3 rounds. Finally, recall that **SZK** proofs have a constant number of rounds [45]. Hence, if it turns out that **PZK** proofs do not have a constant number of rounds, then **PZK** $\neq$ **SZK**, which would resolve a long standing open question.

A reasonable starting point is to consider the popular tool of *commitment schemes*. Such schemes enable a *sender* to commit to a bit b such that the *receiver* cannot learn b from the commitment (this property is called *hiding*), and at the same time the sender cannot change the commitment to another value (this property is called *binding*). Goldreich, Micali, and Wigderson [22] were the first to show that if bit commitments exist, then any **NP** language has a **CZK** proof. Ben-Or *et. al* [8] extended this to any language admitting a public-coin proof (equivalently, any language in **IP** [26]). These results apply also to problems.[1]

It is not clear whether commitment schemes can be useful for the study of perfect zero-knowledge proofs. This is so because they cannot be both perfectly hiding and binding against a computationally unbounded sender. However, this limitation does not apply to *instance-dependent* commitment schemes [6, 28]. Such schemes take an instance x of a problem as an input, and the hiding and the binding properties depend on whether x is a YES or a NO instance. For example, GRAPH-ISOMORPHISM has such a scheme [22]: given a pair of graphs $x = \langle G_0, G_1 \rangle$, a commitment to a bit b is a random isomorphic copy of G_b (this hides b when the graphs are isomorphic, and binds to b otherwise).

Instance-dependent commitment-schemes were formalized by Itoh, Ohta and Shizuya [28], who observed that such schemes can replace the bit commitment-scheme in the protocol of [22] for **NP**. This observation was later extended by Vadhan [52] to the protocol of [8], and it turned out to have far reaching consequences ([28, 36, 35, 42, 41, 29, 44, 45, 11]). As was shown by Vadhan [52], what makes instance-dependent

[1] A problem is a pair $\langle \Pi_Y, \Pi_N \rangle$ of disjoint sets, where Π_Y contains the YES instances, and Π_N contains the NO instances [18]. Any language can be defined as a problem $\langle L, \overline{L} \rangle$.

commitment schemes so attractive is that they facilitate the unconditional study of zero-knowledge proofs. Furthermore, since their hiding and the binding properties are not required to hold simultaneously, they can achieve perfect hiding and binding against a computationally unbounded sender. Hence, such schemes may be useful to collapse the number of rounds in **PZK** proofs.

As a warm up, suppose that public-coin **PZK** problems admit instance-dependent commitment-schemes, and that the schemes are perfectly hiding and constant-round. Since any **PZK** problem has an **AM** proof [20, 2, 47], we could plug the scheme into the zero-knowledge protocol of [8] for **AM** and get a constant-round **PZK** proof. That is, we would collapse the rounds in public-coin **PZK** proofs to a constant.

However, there are a few difficulties. Firstly, current constructions of instance-dependent commitment schemes are only statistically or computationally hiding [36, 52, 42, 45, 11].[2] Thus, it is not clear at all whether perfectly hiding schemes can be constructed from public-coin **PZK** proofs. Secondly, even if such schemes exist, we need to construct ones that are also constant-round. Finally, obtaining constant-round perfectly hiding instant-dependent commitment schemes from public-coin **PZK** problems might be too strong of a requirement. That is, we do not know if a weaker condition may suffice to collapse the rounds in public-coin **PZK** proofs to a constant.

1.2 Our results

We give the first *perfectly* hiding instance-dependent commitment scheme, and our definition uses the same requirements as the scheme of Vadhan [52]: we require hiding on YES instances and binding on NO instances, we consider the honest-verifier case (as was done also in [42, 45]), and since the sender is inefficient, we require that the scheme be simulatable.

Theorem 1.1 *If a problem admits an honest-verifier perfect zero-knowledge (**HVPZK**) proof, then it has perfectly hiding instance-dependent commitment scheme. If the proof is constant-round (or public-coin), then so is the scheme. The scheme is secure against honest receivers, and the sender is inefficient. If the proof is **HVSZK** or **HVCZK**, then the scheme is statistically (respectively, computationally) hiding.*

We remark that there are various definitions for commitments and instance-dependent commitments in the literature, each tailored to a specific application. In our case the difficulty is in achieving a perfectly hiding scheme. Thus, in our definition we observe that unlike in commitments schemes, where the sender always succeeds in producing a commitment, in *instance-dependent* commitment schemes this is only necessary on YES instances, but not on NO instances. This allows us to achieve perfect hiding.

Since our scheme inherits its round complexity from the protocol, it does not collapse the round complexity of public-coin **PZK** proofs to a constant. Hence, the difficulty in collapsing the rounds of public-coin **PZK** proofs is *not* in obtaining a perfectly hiding instance-dependent commitment scheme, but rather in obtaining such a scheme that is also constant-round. But do we have to obtain such a scheme in order to achieve this goal? Using the idea behind the zero-knowledge proof for GRAPH-ISOMORPHISM [22], we show that indeed, the two questions are equivalent. That is, *constant-round*, perfectly hiding instance-dependent commitment schemes are not only sufficient, but also necessary to collapse the round complexity of public-coin **PZK** proofs to a constant.

Corollary 1.2 *A problem admits a* constant-round, *perfectly (respectively, statistically) hiding instance-dependent commitment scheme if and only if it admits a* constant-round ***HVPZK** (respectively, **HVSZK**) proof. The same applies to **HVCZK** if the problem admits a constant-round interactive proof.*

[2] a perfectly hiding scheme was given in [29] using the technique of [14], but it only applies to V-bit zero-knowledge protocols.

This corollary yields a simple and elegant equivalence between zero-knowledge and instance-dependent commitments. Such an equivalence was first given by Vadhan [52], and it was recently improved by Ong and Vadhan [45] to constant rounds schemes with an efficient sender. The difference between our equivalence to that of [45] is that the later only applies to the statistical and the computational setting, but it yields a scheme with additional properties. In contrast, our equivalence applies in all settings (including the perfect), but since our scheme inherits its properties from the protocol, it does not have any additional properties.

In the second part of this paper we construct a non-interactive, perfectly hiding instance-dependent scheme (with an efficient sender) for the class of problems admitting public-coin **PZK** proofs. Unfortunately, the scheme is not binding, but for any polynomial p, the fraction of random inputs to the scheme that violates the binding property can be made as small as $1/2^{p(n)}$, where n is the input length.

Lemma 1.3 *If a problem admits a public-coin **HVPZK** proof, then it has a* non-interactive*, perfectly hiding instance-dependent commitment scheme with an efficient sender. Given common input x of length n, the scheme is binding on all but $1/2^n$ fraction of its random inputs.*

Informally, this lemma shows that we can collapse the rounds of public-coin **PZK** proofs if we can make sure that the prover does not choose its randomness from a small set. This relationship might be implicit in other works [33, 45], but here we show that it holds for the case of public-coin **PZK** proofs. Also, there is a variety of techniques that allow two parties to choose random strings (e.g., hashing [26, 13], interactive hashing [46, 12, 40], and random selection [9, 48]), and our lemma provides an avenue where such techniques might be used.

To address the binding property of our scheme, we defined a *preamble*. Informally, the preamble provides a framework for choosing randomness for the sender, while at the same time making sure that perfect hiding is maintained. It can be thought of as an *instance-dependent randomness selection protocol* (because the randomness is chosen jointly, and depends on instance)

We tested the preamble idea on the simple cases of 3-round public-coin **PZK** proofs and non-interactive perfect zero-knowledge (**NIPZK**) proofs. Although we could only construct the preamble under assumptions on the soundness of the underlying problem, our investigation yielded some consequences of independent interest. For example, we used the circuits of UNIFORM (the **NIPZK**-complete problem of [34]) in the commitment scheme of Naor [39], and obtained a new (essentially, non-interactive) instant-dependent commitment scheme with an efficient sender for **NIPZK** problems admitting a small soundness error.

1.3 Related Work

For a long time, the only general result about **PZK** proofs was the transformation of Damgård, Goldreich, and Wigderson [13] from *constant-round*, *public-coin* honest-verifier perfect zero-knowledge (**HVPZK**) proofs to ones that are **PZK**. Recently, an *error shifting technique* was discovered [34], leading to new complete and hard problems for the perfect setting, and we use these results here.

The round complexity of interactive protocols has been extensively studied. In the context of **IP**, Goldwasser and Sipser [26] showed that any interactive proof can be transformed into a *public-coin* proof with essentially the same number of rounds. The famous collapse theorem of Babai and Moran [4, 33] showed that for any **AM** problem (i.e., any problem with a *constant-round* public-coin proof) the number of rounds can be collapsed to two. Informally, the idea is to let the verifier send its randomness in advance, but at the same time play many copies of the protocol with the prover (in parallel). To prevent an exponential growth in the size of the game tree, after each four rounds the verifier chooses one branch on which the game will continue. We cannot use this idea to collapse the rounds in interactive zero-knowledge proofs because it is not known how to simulate different branches of the interaction. For the relationship between **AM** and **NP** we refer the reader to [10, 3, 32, 38], and we mention that the *unbounded* levels of the public-coin proof hierarchy are not believed to be contained in the *bounded* levels of the polynomial-time hierarchy [1].

The round complexity of zero-knowledge protocols has always been of great interest (c.f., [31, 5], and the recent works of [30, 37]). Fortnow [20], and Aiello and Håstad [2] showed that **SZK** $\subseteq$ **AM**, and an alternative proof was given in [47]. Since **PZK** $\subseteq$ **SZK**, this means that **PZK** $\subseteq$ **AM**. As for **CZK**, if one way functions exist, then any **IP** proof can be turned into a **CZK** proof with essentially the same number of rounds [8]. Thus, collapsing the rounds of **CZK** proofs essentially boils down to collapsing the rounds of **IP** (equivalently, **PSPACE** [49]). We note that Goldreich and Krawczyk [21] showed that public-coin **CZK** proofs *with a negligible soundness error* exist only for trivial problems (i.e., problems in **BPP**). This result extends to **PZK** proofs (because **PZK** $\subseteq$ **CZK**), but it does not apply here because we consider general public-coin proofs. A review on constant-round zero-knowledge *arguments* for **NP** can be found in [7].

2 Definitions

This paper uses standard definitions. We start with the notion of an interactive protocol, originally due to Goldwasser, Micali, and Rackoff [27]. Informally, an interactive protocol is simply a pair of functions sending messages to each other until one of the functions terminate. Formally,

Definition 2.1 (Interactive Protocols) *An interactive protocol is a pair $\langle P, V\rangle$ of functions. The* interaction *between P and V on common input x is the following random process.*

1. *Let r_P and r_V be random inputs to P and V, respectively.*
2. *repeat the following for $i = 1, 2, \ldots$*
   - *(a) If i is odd, let $m_i = P(x, m_1, \ldots, m_{i-1}; r_P)$.*
   - *(b) If i is even, let $m_i = V(x, m_1, \ldots, m_{i-1}; r_V)$.*
   - *(c) If $m_i \in \{\texttt{accept, reject, fail}\}$, then exit loop.*

We say that V accepts *x if $m_i = \texttt{accept}$ for an even i. Interactions yield* transcripts *$\langle x, m_1, \ldots, m_p; r_V\rangle$, and we call the strings m_i* messages. *The probability space containing all the transcripts is called* the view of *V* on *x, and is denoted $\langle P, V\rangle(x)$. The* round complexity *of $\langle P, V\rangle$ is a function p such that for any x, and any interaction on input x, the number of messages exchanged is at most $p(|x|)$. We say that $\langle P, V\rangle$ is* constant round *if p is a constant.*

We say that $\langle P, V\rangle$ is public coin *if V always sends independent portions of r_V, and its last message is a deterministic function of the messages exchanged.*

Now we can define interactive proofs [27]. Informally, a problem has an interactive proof if it has an interactive protocol in which a *common input* x is given to the prover and the verifier, the verifier runs in time polynomial in $|x|$, and it accepts if x is a YES instance, and rejects if x is a NO instance. The distance between the probabilities to accept and reject x is $1/p(|x|)$, where p is a polynomial (e.g., n^a). Formally,

Definition 2.2 (Interactive proofs and arguments) *Let $\Pi = \langle \Pi_Y, \Pi_N\rangle$ be a problem, and let $\langle P, V\rangle$ be an interactive protocol. We say that $\langle P, V\rangle$ is an* interactive proof *for Π if there is a, and $c(n), s(n) : \mathbb{N} \to [0, 1]$ such that $1 - c(n) > s(n) + 1/n^a$ for any n, and the following conditions hold.*

- *Efficiency: V is a probabilistic Turing machine whose running time over the entire interaction is polynomial in $|x|$ (this implies that the number of messages exchanged is polynomial in $|x|$).*
- *Completeness: if $x \in \Pi_Y$, then V accepts in $\langle P, V\rangle(x)$ with probability at least $1 - c(|x|)$. The probability is over r_P and r_V (the randomness for P and V, respectively).*
- *Soundness: if $x \in \Pi_N$, then for any function P^* it holds that V accepts in $\langle P^*, V\rangle(x)$ with probability at most $s(|x|)$. The probability is over the randomness r_V for V.*

If the soundness condition holds with respect to non-uniform polynomial-size circuits, then we say that $\langle P, V\rangle$ is an interactive argument *for* Π.

The function c is the completeness error*, and the function s is the* soundness error*. We say that $\langle P, V\rangle$ has* perfect completeness *(respectively,* perfect soundness*) if $c \equiv 0$ (respectively, $s \equiv 0$).*

We denote by **IP** the class of problems admitting interactive-proofs [27], and by **AM** the class of problems admitting *public-coin, constant-round* interactive-proofs [4, 33].

Definition 2.3 (Efficient prover) *Let $\langle P, V\rangle$ be an interactive proof or argument for an* ***NP*** *problem $\Pi = \langle \Pi_Y, \Pi_N\rangle$. We say that P is an* efficient prover *if given an arbitrary* ***NP*** *witness w for $x \in \Pi_Y$ the prover runs in time polynomial in $|x|$.*

2.1 Indistinguishability

The notion of zero-knowledge is based on indistinguishability between two ensembles: the output of the simulator, and interactions between the prover and the verifier.

A *probability ensemble* is a sequence $\{Y_x\}_{x\in I}$ of random variables, where I is countable set of strings. Indistinguishability is defined in terms of distance between ensembles. A function $f(n)$ is *negligible* if all of its outputs are small when the inputs are large enough. Formally, f is negligible on I if for any polynomial p there is N such that for all $x \in I$ of length at least N it holds that $f(|x|) < 1/p(|x|)$. When I is clear from the context we simply say that $f(n)$ is *negligible*.

We define three notions of indistinguishability: perfect (which will be our main focus), statistical, and computational. Computational indistinguishability is defined in terms of advantage of a *distinguisher* D. Given two distributions Y_x and Z_x, and a circuit D whose output is 0 or 1, the *advantage* of D to distinguish Y_x from Z_x is defined as

$$\mathsf{adv}(D, Y_x, Z_x) \overset{\text{def}}{=} |\Pr[D(Y_x) = 1] - \Pr[D(Z_x) = 1]|,$$

where $\Pr[D(X) = 1]$ is the probability that D outputs 1 given an element chosen according to the distribution X. Notice that if D is probabilistic, then according to our convention this probability is also over the uniform distribution on the randomness of D.

Statistical indistinguishability makes no reference to circuits. Given two discrete distributions X and Y, the *statistical distance* between them is

$$\Delta(\mathrm{X}, \mathrm{Y}) \overset{\text{def}}{=} 1/2 \cdot \sum_{\alpha} |\Pr[X = \alpha] - \Pr[Y = \alpha]| = \max_{S}(|\Pr[X \in S] - \Pr[Y \in S]|).$$

The formal definition of the three notions of indistinguishability follows.

Definition 2.4 (Indistinguishability) *Two probability ensembles $\{Y_x\}_{x\in I}$ and $\{Z_x\}_{x\in I}$ are* computationally indistinguishable *if $\mathsf{adv}(D, Y_x, Z_x)$ is negligible on I for all non-uniform polynomial-size circuits D. They are* statistically identical *(respectively,* statistically indistinguishable*) if $\Delta(\mathrm{Y_x}, \mathrm{Z_x})$ is identically 0 (respectively, negligible) on I.*

Variants of the problem STATISTICAL-DISTANCE (SD) will play a central role in this paper. This problem originated from the study of **SZK** [47], and its instances are pairs of circuits that can be seen as distributions (under the convention that the inputs are uniformly chosen). Instances of SD are statistically close as YES instances, and statistically far as NO instances. Formally,

Definition 2.5 *The problem* $\mathrm{SD}^{\alpha,\beta}$ [47] is the pair $\langle \mathrm{SD}^{\alpha}_{\mathrm{Y}}, \mathrm{SD}^{\beta}_{\mathrm{N}}\rangle$, where

$$\mathrm{SD}^{\alpha}_{\mathrm{Y}} = \{\langle X_0, X_1\rangle |\ \Delta(\mathrm{X}_0, \mathrm{X}_1) \leq \alpha\}\ , \ \text{and}$$
$$\mathrm{SD}^{\beta}_{\mathrm{N}} = \{\langle X_0, X_1\rangle |\ \Delta(\mathrm{X}_0, \mathrm{X}_1) \geq \beta\}\ , \ \text{and}$$

X_0 and X_1 are circuits (treated as distributions).

Notice that $\mathrm{SD} \overset{\text{def}}{=} \mathrm{SD}^{1/3,2/3}$ is **SZK**-complete, and since **SZK** is closed under complement [43, 47], $\overline{\mathrm{SD}}$ is also **SZK**-complete. Since we are dealing with the perfect setting, we will be working only with $\mathrm{SD}^{0,1/2}$.

2.2 Zero-Knowledge

Informally, an interactive proof (or an interactive argument) is *zero-knowledge* if there is a simulator such that the view of the verifier and the output of the simulator are indistinguishable.

Recall that [13] introduced a relaxed definition of perfect zero-knowledge where the simulator is allowed to fail with probability at most $1/2$, and conditioned on not failing the output of the simulator is required to be indistinguishable from the view of the verifier. We remark that all of our results hold regardless of whether the simulator is allowed to fail or not (indeed, we consider the honest-verifier case, where the two notions are equivalent [34]). Thus, to simplify the presentation we use a definition where the simulator is not allowed to fail. We use S^V to denote a Turing machine S with oracle access to Turing machine V.

Definition 2.6 (Zero-knowledge protocols) *A protocol $\langle P, V\rangle$ for a problem $\Pi = \langle \Pi_Y, \Pi_N\rangle$ is* perfect *(respectively,* statistical, computational*)* zero-knowledge *if there is a probabilistic, polynomial-time Turing machine S, called* the simulator*, such that for any probabilistic, polynomial-time Turing machine V^*,*

$$\{\langle P, V^*\rangle(x)\}_{x\in\Pi_Y} \quad \text{and} \quad \left\{S^{V^*}(x)\right\}_{x\in\Pi_Y}$$

are statistically-identical (respectively, statistically indistinguishable, computationally indistinguishable.) The class of problems admitting perfect (respectively, statistical, computational) zero-knowledge protocols is denoted ***PZK*** *(respectively,* ***SZK***, ***CZK****.) When the above ensembles are indistinguishable for $V^* = V$ we say that $\langle P, V\rangle$ is* honest-verifier, perfect *(respectively,* statistical, computational*)* zero-knowledge*, and we denote the respective classes by* ***HVPZK***,***HVSZK****, and* ***HVCZK****.*

3 Trivial Instance-Dependent Commitment Schemes

In this section we prove Theorem 1.1 by showing how to obtain a *perfectly hiding* instance-dependent commitment scheme from any **HVPZK** proof. Our idea also applies to **HVSZK** and **HVCZK** proofs. As a consequence, we get an equivalence between zero-knowledge and instance dependent commitments, thus proving Corollary 1.2.

We start with our definition of instance-dependent commitment schemes. This definition has the same requirements as the scheme of Vadhan [52]: we require hiding on YES instances and binding on NO instances, we consider the honest-verifier case (as was done also in [42, 45]), and since the sender is inefficient, we require that the scheme be simulatable. Notice that we do not care if the prover fails to produce commitments on NO instances (because when x is a NO instance we only care about soundness). Thus, we allow the sender to fail in the commit phase, and require that the failure probability be negligible on YES instances.

Definition 3.1 *An* instance-dependent commitment scheme *for a problem* $\Pi = \langle \Pi_Y, \Pi_N \rangle$ *is a protocol* $\langle S, R \rangle$ *between a* sender S *(with input a bit* b*) and a* receiver R. *The randomness of* R *and* S *is denoted* r_S *and* r_R*, respectively. The running time of* R *is polynomial in* $|x|$*, where* x *is the* common input. *The protocol has two parts:*

- **The commit phase.** *This is the first part of the protocol. If both* S *and* R *follow their instructions, then with probability at least* $1-2^{-|x|}$ *over their randomness this stage ends successfully. The* commitment *of* S *to* b *is denoted by* $\langle S_b, R \rangle(x)$*. It contains* x*, the messages exchanged in this phase, and* r_R*.*

- **The reveal phase.** *This is the second part of the protocol. In this part* S opens the commitment to b *by sending* b *and* r_S *to* R*. The receiver either accepts or rejects* b*. In this stage the view of the receiver is simply denoted* $\langle r_S, b \rangle$*.*

The protocol satisfies three properties:

Hiding. $\langle S, R \rangle$ *is* perfectly (respectively, statistically, computationally) hiding *on* Π_Y *if* $\{\langle S_0, R\rangle(x)\}_{x\in\Pi_Y}$ *and* $\{\langle S_1, R\rangle(x)\}_{x\in\Pi_Y}$ *are identical (respectively, statistically indistinguishable, computationally indistinguishable).*

Biding. $\langle S, R \rangle$ *is* statistically binding *on* Π_N *if for any function* S^* *and common input* $x \in \Pi_N$*, the probability over* r_R *that* R *accepts both* 0 *and* 1 *in the reveal phase is at most* $1/2^{|x|}$.

Simulation *(against an honest receiver).* $\langle S, R \rangle$ *is* perfectly (respectively, statistically, computationally) simulatable against the honest receiver *if there is a probabilistic Turing machine* M *that runs in time polynomial in* x *such that for any* b *it holds that* $\{M(x, b)\}_{x\in\Pi_Y}$ *and* $\{\langle\langle S_b, R\rangle(x), \langle r_S, b\rangle\rangle\}_{x\in\Pi_Y}$ *are identical (respectively, statistically indistinguishable, computationally indistinguishable).*

We start with the forward direction of Corollary 1.2.

Lemma 3.2 *If a problem admits a perfectly hiding instance-dependent commitment scheme, then it has a* ***HVPZK*** *proof. If the scheme is constant-round (or public-coin), then so is the* ***HVPZK*** *proof.*

Proof: We use the idea behind the proof systems for GRAPH-ISOMORPHISM [22]. That is, the prover commits to the bit 0, and the verifier replies with a random bit b. The verifier accepts only if the prover opens the commitment to the bit b. Soundness follows from the fact that the commitment is binding on NO instances. The hiding property of the scheme guarantees that the same commitment can be opened to both 0 and 1, and thus the protocol is complete. The protocol is **HVPZK** because the simulator can guess b, and then simulate a commitment to b with the honest receiver by executing $M(x, b)$, where M is guaranteed by the simulation requirement from Definition 3.1. Notice that the fact that the scheme may fail does not affect the perfect simulation because, just like the prover, the simulator will fail in the commit phase. □

The above proof also applies to **HVSZK** problems, but it may not apply **HVCZK** because the prover may not be able to open commitments to both 0 and 1. Instead, we can plug the instance-dependent commitment scheme in the protocol of [8] for **AM**, and if the underlying problem has a constant round interactive proof, then we get a constant-round public-coin **HVCZK** proof. Notice that in all cases we can apply the transformation of [13] to the constant-round honest-verifier zero-knowledge proof, and obtain a constant-round zero-knowledge proof (against malicious verifiers).

We proceed to prove Theorem 1.1 by showing how to construct instance-dependent commitment schemes from zero-knowledge protocols. Combining this with the above lemma, we obtain Corollary 1.2. Again, we deal with **HVPZK**, but the proof easily extends to **HVSZK** and **HVCZK**.

Proof of Theorem 1.1: Let Π be a problem admitting a constant-round **HVPZK** proof $\langle P, V\rangle$. Since we deal with the honest verifier, the completeness and soundness error can be reduced to $1/2^n$. We use $\langle P, V\rangle$ to construct an instance-dependent commitment scheme for Π. The idea is to use the soundness property of $\langle P, V\rangle$ to obtain binding, the completeness and zero-knowledge properties to obtain hiding, and the zero-knowledge property to obtain simulation.

Formally, let S_b denote the sender with a bit b, let R to denote the receiver, and let x denote the input. In the commit phase S and R execute P and V on input x, respectively. There are two cases.

- If V accepts, then the sender does not send b, and the commit phase terminates successfully. Notice that the bit b takes no part in the execution of the commit phase. In the reveal phase the sender simply reveals b (without sending its randomness), and the receiver accepts.

- If V rejects, then both the commit and the reveal phases terminate. That is, in the commit phase the sender sends `fail`, and in the reveal phase the sender does not send anything and the receiver rejects.

We verify the properties of the scheme. Let $n \overset{\text{def}}{=} |x|$. If x is a NO instance, then the scheme is binding because R rejects with probability at least $1 - 2^{-n}$ over r_R. If x is a YES instance and both S and R follow their instructions, then the commit phase terminates successfully because V accepts x with probability at least $1 - 2^n$ over the randomness of S and R. Since S does not send b in the commit phase, the scheme is perfectly hiding. Notice that with probability at most $1/2^n$ the sender fails in the commit phase, but the bit b is still hidden. The simulator M for $\langle S, R\rangle$ simply mimics the sender, and it can be easily constructed from the **HVPZK** simulator S of $\langle P, V\rangle$. Formally, $M(x, b)$ obtains a transcript $\langle x, m_1, m_2, \ldots, m_v; r_V\rangle$ of $S(x)$, and if V accepts in this transcript, then M outputs $\langle\langle x, m_1, m_2, \ldots, m_v; r_V\rangle, \langle \epsilon, b\rangle\rangle$, where ϵ is the empty string, and b is the bit of the sender. Otherwise, just like the prover, it adds the `fail` message to the transcript, and outputs $\langle\langle x, m_1, m_2, \ldots, m_v, \texttt{fail}; r_V\rangle, \epsilon\rangle$. □

Notice that in the above proof S is a **HVPZK** simulator, and thus M perfectly simulates the commitment. However, although b is not involved in the commit phase, if S is a **HVSZK** or a **HVCZK** simulator, then M will only statistically or computationally simulate the commit phase, and thus the hiding property will be statistical or computational, respectively.

4 Instance-Dependent Commitments from Hard Problems

In this section we prove Lemma 1.3 by constructing a perfectly hiding instance-dependent commitment scheme. Although our scheme is not binding, the binding property holds on almost all the inputs, and this shows that we can collapse the rounds of public-coin **PZK** proofs if we can make sure that the prover does not choose its randomness from a small set.

We build on IDENTICAL DISTRIBUTIONS (ID), the hard problem of [34] for the class of problems admitting public-coin **HVPZK** proofs. This problem originated from the reduction of Sahai and Vadhan [47]. Instances of ID are triplets $\langle X_0, X_1, Z \rangle$ of circuits, where the circuit Z can be ignored (because in any zero-knowledge proof or an instance-dependent commitment scheme for ID, the verifier can sample Z and reject immediately if $\Pr[Z = 1] \leq 1/3$). Thus, throughout this paper, when we refer to ID, we actually refer to instances $\langle X_0, X_1 \rangle$ of $\mathrm{SD}^{0,1/2}$. That is, as YES instances X_0 and X_1 represent the same distribution, and as NO instances they represent statistically far distributions.

Definition 4.1 *The problem* IDENTICAL DISTRIBUTIONS *is* $\mathrm{ID} \overset{\mathit{def}}{=} \langle \mathrm{ID_Y}, \mathrm{ID_N} \rangle$, *where*

$$\mathrm{ID_Y} = \{\langle X_0, X_1, Z\rangle | \ \Delta(\mathrm{X_0}, \mathrm{X_1}) = 0 \quad \textit{and} \quad \Pr[Z = 1] \geq 2/3\}\ , \ \textit{and}$$
$$\mathrm{ID_N} = \{\langle X_0, X_1, Z\rangle | \ \Delta(\mathrm{X_0}, \mathrm{X_1}) \geq 1/2 \ \ \textit{or} \ \ \Pr[Z = 1] \leq 1/3\}\,.$$

4.1 A Perfectly Hiding Scheme That is Almost Binding

Our goal is to construct a constant-round, perfectly hiding instance-dependent commitment scheme for ID. Micciancio and Vadhan [36] showed that $\mathrm{SD}^{0,1}$ has such a scheme: a commitment to the bit b is a random sample of X_b. With respect to ID, this idea guarantees perfect hiding on YES instance because X_0 and X_1 represent the same distribution, and thus it is impossible to determine b from y. However, this idea does not guarantee binding when $\langle X_0, X_1 \rangle$ is a NO instance of ID because there could be r and r' such that $X_0(r) = X_1(r')$, which may allow the sender to open y as a commitment to both 0 or 1 .

Our idea is to use *multiple intertwined samples*. That is, we use $n = |\langle X_0, X_1 \rangle|$ additional samples, and the string r appears in all of them. Formally, to commit to a bit b the prover chooses $n+1$ random strings $r, r_1, \ldots, r_n$, and it sends to the verifier the commitment $\vec{y} = \langle X_b(r), X_b(r \oplus r_1), \ldots, X_b(r \oplus r_n) \rangle$. As before, in the reveal phase the prover sends b and $r, r_1, \ldots, r_n$, and the verifier checks that $\vec{y}$ was computed correctly. This scheme is described in Figure 1.

The first observation about the modified scheme is that if $r, r_1, \ldots, r_n$ are uniformly chosen, then the strings $r, r \oplus r_1, \ldots, r \oplus r_n$ are also uniformly chosen and *independent*. Thus, the modified scheme retains the perfect hiding property. The second observation is that the modified scheme is not binding. However, notice that in the previous scheme the sender could cheat using any pair $\langle r, r' \rangle$ for which $X_0(r) = X_1(r')$, and many such pairs may exist. In contrast, in the modified scheme the sender can cheat using only a small fraction of the strings $r_1, \ldots, r_n$, regardless of the number of pairs $\langle r, r' \rangle$ for which $X_0(r) = X_1(r')$ (intuitively, replacing $X_0(r)$ with $X_1(r')$ affects the rest of the samples, which requires a cheating sender to adjust the strings $r_1, \ldots, r_n$). Hence, $\vec{y}$ cannot be opened as a commitment to both 0 and 1, except for a small fraction of the strings $r_1, \ldots, r_n$. To formalize this, we start with one sample.

Lemma 4.2 *Let X_0 and X_1 be circuits. Let r and r' be stings such that $X_0(r) = X_0(r')$, and let $\alpha \overset{\mathit{def}}{=} \Delta(\mathrm{X_0}, \mathrm{X_1})$. If r_1 is uniformly chosen, then the probability that $X_0(r \oplus r_1) = X_1(r' \oplus r_1)$ is at most $1 - \alpha$.*

An instance-dependent scheme $\langle S, R\rangle$

Common input: a pair of circuits$\langle X_0, X_1\rangle$. Let $n = |\langle X_0, X_1\rangle|$.
Private input for S**:** a bit b.

The *sender* S commits to a bit b as follows:

1. S uniformly chooses a string r, and computes $y \overset{\text{def}}{=} X_b(r)$.
2. S uniformly chooses strings $r_1, \ldots, r_n$, and computes $y_i \overset{\text{def}}{=} X_b(r \oplus r_i)$.
3. S sends $\vec{y} \overset{\text{def}}{=} \langle y, y_1, \ldots, y_n\rangle$ to the *receiver* R.

In the reveal phase S sends $r, r_1, \ldots, r_n$ to R.

Figure 1: A perfectly hiding scheme whose binding property holds on almost all the random inputs.

Proof: We use two sets in our analysis. The first set contains strings y that are more likely to be outputted by X_0 than by X_1, and the second set is defined analogously. Formally,

$$X_0^+ \overset{\text{def}}{=} \{y|\Pr[X_0 = y] \geq \Pr[X_1 = y]\}\,, \text{ and } X_1^+ \overset{\text{def}}{=} \{y|\Pr[X_1 = y] > \Pr[X_0 = y]\}\,.$$

Using these sets we upper bound the probability that $X_0(r \oplus r_1) = X_1(r' \oplus r_1)$.

$$\begin{aligned}
& \Pr_{r_1}[X_0(r \oplus r_1) = X_1(r' \oplus r_1)] \\
= \;& \Pr_{r_1}[(X_0(r \oplus r_1) = X_1(r' \oplus r_1)) \wedge X_1(r' \oplus r_1) \notin X_0^+] + \\
& \Pr_{r_1}[(X_0(r \oplus r_1) = X_1(r' \oplus r_1)) \wedge X_1(r' \oplus r_1) \in X_0^+].
\end{aligned}$$

Clearly, the first expression in the above sum is upper bounded by $\Pr_{r_1}[X_1(r' \oplus r_1) \notin X_0^+]$. The same applies to the second expression, but we use the equality in this expression to replace $X_1(r' \oplus r_1) \in X_0^+$ with $X_0(r \oplus r_1) \in X_0^+$. Hence, we get that

$$\begin{aligned}
& \Pr_{r_1}[(X_0(r \oplus r_1) = X_1(r' \oplus r_1)) \wedge X_1(r' \oplus r_1) \notin X_0^+] + \\
& \Pr_{r_1}[(X_0(r \oplus r_1) = X_1(r' \oplus r_1)) \wedge X_1(r' \oplus r_1) \in X_0^+] \\
\leq \;& \Pr_{r_1}[X_1(r' \oplus r_1) \notin X_0^+] + \Pr_{r_1}[X_0(r \oplus r_1) \in X_0^+] \\
= \;& 1 - \Pr_{r_1}[X_1(r' \oplus r_1) \in X_0^+] + \Pr_{r_1}[X_0(r \oplus r_1) \in X_0^+].
\end{aligned}$$

Now we use a fact that follows from the definition of statistical distance (see Fact 3.1.9 in [51]). According to this fact, $\Delta(\mathrm{X}_0, \mathrm{X}_1) = \Pr[X_0 \in X_0^+] - \Pr[X_1 \in \; X_0^+]$. Thus, since r and r' are fixed, we get that

$$\Delta(\mathrm{X}_0, \mathrm{X}_1) = \Pr_{r_1}[X_0(r \oplus r_1) \in X_0^+] - \Pr_{r_1}[X_1(r' \oplus r_1) \in X_0^+].$$

Since $\Delta(\mathrm{X}_0, \mathrm{X}_1) = \alpha$, we get that $\Pr_{r_1}[X_0(r \oplus r_1) = X_1(r' \oplus r_1)] \leq 1 - \alpha$. □

It follows that by taking more samples, we can reduce the number of strings that allow a cheating sender to open commitments to both 0 and 1. Formally, let X_0 and X_1 be circuits on inputs of length m, and let

$n = |\langle X_0, X_1 \rangle|$. We claim that for any r and r', if $r_1, \ldots, r_{2n}$ are uniformly chosen, then with probability at most 2^{-n} it holds that

$$\langle X_0(r), X_0(r \oplus r_1), \ldots, X_0(r \oplus r_{2n}) \rangle = \langle X_1(r), X_1(r \oplus r_1), \ldots, X_1(r \oplus r_{2n}) \rangle.$$

This is so because by Lemma 4.2, the probability over $r_1, \ldots, r_{2n}$ that the above equality holds is at most $(1 - \alpha)^{2n} \leq 2^{-2n}$ (recall that $\alpha = \Delta(\mathrm{X}_0, \mathrm{X}_1) \geq 1/2$ when $\langle X_0, X_1 \rangle$ is a NO instance of ID). Since there are at most $2^{2m} \leq 2^n$ pairs $\langle r, r' \rangle$ for which $X_0(r) = X_1(r')$, our scheme is not binding with probability at most $2^n \cdot 2^{-2n} \leq 2^{-n}$. Lemma 1.3 follows.

5 A Preamble for Jointly Choosing Randomness

In the previous section we constructed a scheme that is not binding if the sender chooses its randomness from a small set. In this section we define a *preamble* that provides a framework for choosing randomness for the sender, while at the same time making sure that perfect hiding is maintained. Such a preamble would fix the binding property of our scheme, thus collapsing the round complexity of public-coin **PZK** proofs to a constant. We then test the preamble on the simple cases of 3-round public-coin **PZK** proofs (Section 5.1) and **NIPZK** proofs (Section 5.2), and obtain interesting consequences.

Motivation. Since the randomness of our scheme is chosen by the sender, a cheating sender may be able to open the commitment to both 0 and 1. Hence, it makes sense to restrict the randomness used by the sender. In the statistical setting Goldreich and Vadhan [25] used the hashing technique of Goldwasser and Sipser [26], whereby one party chooses a hash function h, and the other party is restricted to strings r such that $h(r) = 0$. Indeed, forcing the sender to use randomness from the small set $h^{-1}(0)$ will make our scheme binding, but in the perfect setting it compromises the hiding property (a similar issue occurs in [42], where the interactive hashing technique due to Ding, Harnik, Rosen, and Shaltiel [16] is used; interactive hashing was introduced by Naor, Ostrovsky, Venkatesan, and Yung [40]).

Thus, we need an *instance-dependent randomness selection protocol*. That is, a protocol that restricts the randomness of the sender in a way that depends on the common input. We formalize this using a *preamble*. The first part of the preamble defines a set A which is big on YES instances and small on NO instances. Intuitively, A represents all the choices of randomness for the sender, and it contains a small subset B of strings that violate the binding property. The second part uses the set A to define a string r such that on YES instances r can be any string in A, and on NO instances r is unlikely to be a string in B. More formally,

1. **Defining a set.** Let x be an instance of ID, and let $p(n)$ denote the length of the random input to our scheme. The sender and the receiver execute a protocol that defines a set $A \subseteq \{0,1\}^{p(n)}$, where $n \stackrel{\text{def}}{=} |x|$. If x is a YES instance, then $|A| = 2^{p(n)}$, and if x is a NO instance, then $A \ll 2^{p(n)}$.

2. **Randomizing the set.** Let $B \subsetneq \{0,1\}^{p(n)}$ be the set of "bad" strings (those that violate the binding property of our scheme). Using A the parties define a string r. If x is a YES instance, then r can equally be any string in $A = \{0,1\}^{p(n)}$, and if x is a NO instance, then r is unlikely to be in B.

Suppose that we could construct such a preamble for ID. We could then execute S and R from our scheme in Figure 1, and have S commit to its bit b using r as randomness. If x is a YES instance, then r can be any string in A, and thus $\vec{y}$ perfectly hides b. If x is a NO instance, then $r \notin B$ with high probability over the randomness of the receiver, and thus $\vec{y}$ binds the sender to b.

5.1 The Case of 3-round Public-Coin PZK Proofs

Our goal is to construct a preamble for any problem that admits a public-coin **PZK** proof. Since we do not know how to do it, we deal with the simple case of 3-round public-coin **PZK** proofs. Notice that the preamble must have an *efficient sender*, or else we could directly apply Theorem 1.1 to the 3-round public-coin **PZK** proof, and obtain a constant-round, perfectly hiding instance-dependent commitment scheme.

Consider a 3-round, public-coin **PZK** proof $\langle P, V\rangle$ with a simulator M, and let $\langle x, m_1, r_1, m_2\rangle$ denote the output of $M(x)$. That is, on input x the prover sends m_1, the verifier sends r_1, the prover replies with m_2, and based on these messages the verifier accepts of rejects. To simplify the presentation, we let $|r_1| = n \stackrel{\text{def}}{=} |x|$, and denote by n^c the length of the random input to our commitment scheme $\langle S, R\rangle(x)$.

Preamble - Step 1**.** The first step of our preamble is to define a set A. This can be done by having the sender execute $M(x)$, obtain a transcript $\langle x, m_1, r_1, m_2\rangle$, and send m_1 to the receiver. We define A to be the set of all r_1 such that $M(x) = \langle x, m_1, r_1, m_2\rangle$ and $V(x, m_1, r_1, m_2) = \texttt{accept}$. Actually, we want A to contain strings of length n^c. Thus, we let the sender sample M for n^{c-1} times, obtain a vector $\vec{r}$ of n^{c-1} messages r_1, and send a vector of n^{c-1} messages m_2 to the receiver. Suppose that $\langle P, V\rangle$ has soundness error $1/2$ and perfect completeness. Thus, if x is a YES instance, then $A = \{0,1\}^{n^c}$, and if x is a NO instance, then A contains at most a $1/2^{n^c-1}$ fraction of the strings in $\{0,1\}^{n^c}$.

Preamble - Step 2**.** The second step of our preamble is to define a string r that would later be used by the sender in our commitment scheme. Let B be the set of all strings that violate the binding property of our scheme. By Lemma 1.3, B contains at most a $1/2^n$ fraction of the strings in $\{0,1\}^{n^c}$. We remark that if $A \cap B = \emptyset$, then we could simply define $r = \vec{r}$ (i.e., the randomness for S is the concatenation of the n^c messages r_1), but of course, this may not be the case. Suppose that $\langle P, V\rangle$ has a very small soundness error of $1/2^{n-(n^{-c+2})/2}$. In such a case we can let the receiver send a random string r' to the sender, and define $r \stackrel{\text{def}}{=} \vec{r} \oplus r'$. When x is a NO instance the probability that $r \in B$ is at most $|A| \cdot 1/2^n = (2^n/2^{n-(n^{-c+2})/2})^{n^{c-1}}/2^n = 2^{-n/2}$.

Thus, if the sender in our scheme uses r as its randomness, then the scheme is binding on NO instances. If x is a YES instance, then r is hidden from the receiver, and thus our scheme is perfectly hiding. We can remove the assumption on perfect completeness by allowing the sender to fail (this happens with small probability because, after executing M many times, the sender is likely to obtain n^c accepting transcripts). Unfortunately, we do not know how to remove the restriction on the soundness.

5.2 The Case of Non-Interactive Perfect Zero-Knowledge (NIPZK) Proofs

Our goal was to collapse the number of rounds in public-coin **PZK** proofs to a constant. We could achieve this goal if our scheme was binding. We tried to construct a preamble that would fix the binding property, but we were unsuccessful even for the simple case of 3-round public-coin **PZK** proofs.

In this section we want to provide a better understanding into the difficulties involved. Thus, we try to construct the preamble for the other simple case of **NIPZK** proofs. Although we could not construct the preamble, our investigation yields two interesting consequences. Firstly, we show how to use the circuits from the study of **NIPZK** in the commitment scheme of Naor [39]. This leads to a new perfectly-hiding instance-dependent commitment for **NIPZK** problems with a small soundness error. Secondly, we show how to use hash functions without damaging the hiding property. This is useful because, as we mentioned earlier, most hashing techniques (e.g., [26, 16]) do not apply in the perfect setting.

Since we are dealing with the non-interactive setting, our underlying problem will be the **NIPZK**-complete problem of [34]. This problem, called UNIFORM (UN), was obtained by modifying the reductions of Goldreich, Sahai, and Vadhan [24], originally due to De Santis, Di Crescenzo, Persiano, and Yung. Intuitively, YES instances of UN are circuits that represent the uniform distribution, and NO instances are circuits that have a small range. Actually, the circuits have an additional output bit, but it can be ignored (in the same way that we ignored the circuit Z of the problem IDENTICAL DISTRIBUTIONS). Thus, throughout this section we will be working with a variant of STATISTICAL DISTANCE FROM UNIFORM (SDU), the **NISZK**-complete problem of [24]. The definition of UNIFORM can be found in Appendix A.

Definition 5.1 *Define* $\mathrm{SDU}' \stackrel{def}{=} \langle \mathrm{SDU}'_Y, \mathrm{SDU}'_N \rangle$ *as*

$$\mathrm{SDU}'_Y = \{X|\ \Delta(\mathrm{X}, \mathrm{U_n}) = 0\}\ , \text{ and}$$
$$\mathrm{SDU}'_N = \{X|\ \mathrm{Rng(X)} < 2^{\mathrm{n}}/3\}\ ,$$

where X is a circuit with n output bits, and U_n is the uniform distribution on $\{0,1\}^n$.

Motivation. Our goal is to construct a constant-round, perfectly hiding instance-dependent commitment scheme, this time for SDU'. Again, the scheme must have an *efficient sender*, or else it trivially exists by Theorem 1.1 (because **NIPZK** proofs are constant-round **HVPZK** proofs in particular).

As a warm up, consider the commitment scheme of Naor [39], which uses a pseudo-random generator $G : \{0,1\}^n \to \{0,1\}^{3n}$. In this scheme the receiver sends a random string $r \in \{0,1\}^{3n}$ to the sender. The sender chooses a random string $r' \in \{0,1\}^n$, and commits to 0 by sending $G(r') \oplus r$, and to 1 by sending $G(r')$. To see why this scheme is binding against computationally unbounded senders, consider a commitment $G(r')$. Since the range of G contains at most a $1/2^{2n}$ fraction of the strings $\{0,1\}^{3n}$, the probability that $G(r') \oplus r$ falls back into $\mathrm{Rng(G)}$ (that is, $G(r') \oplus r = G(r'')$ for some r'') is at most 2^{-2n}. Thus, the scheme is binding with probability at least $|\mathrm{Rng(G)}| \cdot 2^{-2\mathrm{n}} = 2^{-\mathrm{n}}$.

We apply this idea to instances of SDU'. That is, on circuit X with n output bits the receiver sends a uniformly chosen $r \in \{0,1\}^n$, and the sender commits to 0 by sending $X(r') \oplus r$, and to 1 by sending $X(r')$. The resulting instance-dependent scheme is perfectly hiding on YES instances. If SDU' has a very small soundness error of $2^{-n/4}$, then its NO instances satisfy $|\mathrm{Rng(X)}| \leq 2^{\mathrm{n}/4}$, and by the same argument as above, the probability over r that there are r' and r'' such that $X(r') \oplus r = X(r'')$ is at most $|\mathrm{Rng(X)}| \cdot 2^{-3\mathrm{n}/4} \leq 2^{-\mathrm{n}/2}$. Of course, the range of X may be bigger, and thus we cannot use this idea.

Constructing a Preamble. We modify the scheme of Naor [39] using hash functions. Intuitively, the sender will commit to 0 by sending $h'(r)$, and to 1 by sending $X(r_0)$. The string r_0 is chosen by the sender, the string r is chosen by the receiver, and h' is a hash function chosen jointly. The idea is that, if X is a NO instance, then it has a small range, and thus it is unlikely that $h'(r) \in \mathrm{Rng(X)}$. The scheme guarantees perfect hiding on YES instances, but as we shall see it does not guarantee binding. We formally describe our scheme using the preamble idea.

Let X be a circuit with n output bits, let c be some constant, and define $X' \stackrel{\mathrm{def}}{=} \oplus^{n^c} X$ as the circuit that takes n^{c-1} strings r_i, and outputs $X(r_1), \ldots, X(r_{n^{c-1}})$. The circuit X' has n^c output gates. If X is a YES instance, then X' represents the uniform distribution, and if X is a NO instance, then X' has a small range.

Preamble - Step 1. In the first step the sender picks two samples of X', and sends the XOR to the receiver. That is, the sender picks r_0, r_1, computes $h_0 = X'(r_0), h_1 = X'(r_1)$, and sends $y = h_0 \oplus h_1$ to the receiver.

Using the notation of our preamble, A is a set of hash functions. If X is a YES instance, then $A = \{0,1\}^{n^c}$, and if X is a NO instance, then A contains a small fraction of $\{0,1\}^{n^c}$.
Preamble - Step 2. In the second step the receiver replies with a uniformly chosen hash function h and an input r for h. The sender uses h to define $h' = h_0 \oplus h$. Informally, this ensures that h' does not belong to the set B of hash functions that do not evenly spread their domain over their range.

Now we execute the scheme. The sender commits to $b = 0$ by sending $h'(r)$, and to $b = 1$ by sending $X(r_0)$. If X is a YES instance, then both h' and r_0 are hidden from the receiver, and the scheme is hiding. If X is a NO instance, then both X and X' have small ranges. Since h' is a good hash function, it is unlikely to map r to $\mathrm{Rng(X)}$, and thus the scheme should be binding. Unfortunately, this is not the case because, although h' is likely to be a good hash function, there are $|\mathrm{Rng(X')}|$ possibilities for h_0. In other words, although h' is good, $h'(r)$ may fall into $\mathrm{Rng(X)}$.

6 Conclusion

We initiated a preliminary investigation into the question whether the round complexity of public-coin **PZK** proofs can be collapsed to a constant. We gave the first perfectly hiding instance-dependent commitment scheme, and showed that obtaining such a scheme that is also constant round is equivalent to achieving this collapse. We then tried to construct a constant-round, perfectly hiding scheme using the circuits from the hard problem for public-coin **PZK** proofs [34]. Although we could not fix the binding property of the scheme, our attempts had some interesting consequences, including a connection between choosing the randomness of the sender and collapsing the rounds, the definition of the preamble, the difficulty in constructing the preamble, and the use of the circuits of the **NIPZK**-complete problem in the scheme of Naor [39].

Acknowledgements. We would like to thank Omkant Pandey and Oded Goldreich for useful discussions on an earlier version of this paper, and to Salil Vadhan for his advice on the preamble from Section 5. Ivan Visconti greatly contributed to this paper, especially to the hashing idea from Section 5.2.

References

[1] William Aiello, Shafi Goldwasser, and Johan Håstad. On the power of interaction. *Combinatorica*, 10(1):3–25, 1990.

[2] William Aiello and Johan Håstad. Statistical zero-knowledge languages can be recognized in two rounds. *J. of Computer and System Sciences*, 42(3):327–345, June 1991.

[3] Vikraman Arvind and Johannes Köbler. On pseudorandomness and resource-bounded measure. *Theor. Comput. Sci.*, 255(1-2):205–221, 2001.

[4] László Babai. Trading group theory for randomness. In *STOC*, pages 421–429, 1985.

[5] Boaz Barak, Yehuda Lindell, and Salil Vadhan. Lower bounds for non-black-box zero knowledge. *J. Comput. Syst. Sci.*, 72(2):321–391, 2006.

[6] M. Bellare, S. Micali, and R. Ostrovsky. The (true) complexity of statistical zero knowledge. In *STOC '90: Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 494–502, New York, NY, USA, 1990. ACM.

[7] Mihir Bellare, Markus Jakobsson, and Moti Yung. Round-optimal zero-knowledge arguments based on any one-way function. In *EUROCRYPT*, pages 280–305, 1997.

[8] Michael Ben-Or, Oded Goldreich, Shafi Goldwasser, Johan Håstad, Joe Kilian, Silvio Micali, and Phillip Rogaway. Everything provable is provable in zero-knowledge. In *CRYPTO*, pages 37–56, 1988.

[9] Manuel Blum. Coin flipping by telephone - a protocol for solving impossible problems. In *COMPCON*, pages 133–137, 1982.

[10] Ravi B. Boppana, Johan Håstad, and Stathis Zachos. Does co-NP have short interactive proofs? *Inf. Process. Lett.*, 25(2):127–132, 1987.

[11] André Chailloux, Dragos Florin Ciocan, Iordanis Kerenidis, and Salil P. Vadhan. Interactive and noninteractive zero knowledge are equivalent in the help model. In *TCC*, pages 501–534, 2008.

[12] Ivan Damgård. Interactive hashing can simplify zero-knowledge protocol design without computational assumptions (extended abstract). In *CRYPTO*, pages 100–109, 1993.

[13] Ivan Damgård and Oded Goldreich Avi Wigderson. Hashing functions can simplify zero-knowledge protocol design (too). Technical Report RS-94-39, BRICS, November 1994.

[14] Ivan B. Damgård. On the existence of bit commitment schemes and zero-knowledge proofs. In *CRYPTO '89: Proceedings on Advances in cryptology*, pages 17–27, New York, NY, USA, 1989. Springer-Verlag New York, Inc.

[15] Whitfield Diffie and Martin E. Hellman. Multiuser cryptographic techniques. In *AFIPS National Computer Conference*, pages 109–112, 1976.

[16] Yan Zong Ding, Danny Harnik, Alon Rosen, and Ronen Shaltiel. Constant-round oblivious transfer in the bounded storage model. *J. Cryptology*, 20(2):165–202, 2007.

[17] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithm. *IEEE Transactions on Information Theory*, IT-31(4):469–472, 1984.

[18] Shimon Even, Alan L. Selman, and Yacov Yacobi. The complexity of promise problems with applications to public-key cryptography. *Information and Control*, 61(2):159–173, May 1984.

[19] Uriel Feige, Amos Fiat, and Adi Shamir. Zero-knowledge proofs of identity. *J. Cryptology*, 1(2):77–94, 1988.

[20] 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.

[21] Oded Goldreich and Hugo Krawczyk. On the composition of zero-knowledge proof systems. *SIAM J. Comput.*, 25(1):169–192, 1996.

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

[23] Oded Goldreich, Amit Sahai, and Salil P. Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *STOC*, pages 399–408, 1998.

[24] Oded Goldreich, Amit Sahai, and Salil P. Vadhan. Can statistical zero knowledge be made non-interactive? or on the relationship of SZK and NISZK. In *CRYPTO*, pages 467–484, 1999.

[25] Oded Goldreich and Salil P. Vadhan. Comparing entropies in statistical zero-knowledge with applications to the structure of SZK. In *IEEE Conference on Computational Complexity*, pages 54–73, 1999.

[26] S. Goldwasser and M. Sipser. Private-coins versus public-coins in interactive proof systems. In Silvio Micali, editor, *Advances in Computing Research*, volume 5, pages 73–90. JAC Press, Inc.,1989.

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

[28] Toshiya Itoh, Yuji Ohta, and Hiroki Shizuya. A language-dependent cryptographic primitive. *J. Cryptology*, 10(1):37–50, 1997.

[29] Bruce Kapron, Lior Malka, and Venkatesh Srinivasan. Characterizing non-interactive instance-dependent commitment-schemes (NIC). In *34th International Colloquium on Automata, Languages and Programming (ICALP 2007)*, volume 4596 of *LNCS*, pages 328–339, 2007.

[30] Jonathan Katz. Which languages have 4-round zero-knowledge proofs? In *TCC*, pages 73–88, 2008.

[31] Joe Kilian, Erez Petrank, and Charles Rackoff. Lower bounds for zero knowledge on the internet. In *FOCS*, pages 484–492, 1998.

[32] Adam Klivans and Dieter van Melkebeek. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM J. Comput.*, 31(5):1501–1526, 2002.

[33] Babai László and Shlomo Moran. Arthur-merlin games: A randomized proof system and a hierarchy of complexity classes. *J. of Computer and System Sciences*, 36:254–276, 1988.

[34] Lior Malka. How to achieve perfect simulation and a complete problem for non-interactive perfect zero-knowledge. In *TCC*, pages 89–106, 2008.

[35] Daniele Micciancio, Shien Jin Ong, Amit Sahai, and Salil P. Vadhan. Concurrent zero knowledge without complexity assumptions. In *TCC*, pages 1–20, 2006.

[36] Daniele Micciancio and Salil P. Vadhan. Statistical zero-knowledge proofs with efficient provers: Lattice problems and more. In *CRYPTO*, pages 282–298, 2003.

[37] Daniele Micciancio and Scott Yilek. The round-complexity of black-box zero-knowledge: A combinatorial characterization. In *TCC*, pages 535–552, 2008.

[38] Peter Bro Miltersen and N. V. Vinodchandran. Derandomizing arthur-merlin games using hitting sets. *Computational Complexity*, 14(3):256–279, 2005.

[39] Moni Naor. Bit commitment using pseudorandomness. *J. Cryptology*, 4(2):151–158, 1991.

[40] Moni Naor, Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Perfect zero-knowledge arguments for *p* using any one-way permutation. *J. Cryptology*, 11(2):87–108, 1998.

[41] Minh-Huyen Nguyen, Shien Jin Ong, and Salil Vadhan. Statistical zero-knowledge arguments for NP from any one-way function. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006)*, pages 3–14, October 2006. Berkeley, CA.

[42] Minh-Huyen Nguyen and Salil Vadhan. Zero knowledge with efficient provers. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 287–295, New York, NY, USA, 2006. ACM Press.

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

[44] Shien Jin Ong and Salil P. Vadhan. Zero knowledge and soundness are symmetric. In *EUROCRYPT*, pages 187–209, 2007.

[45] Shien Jin Ong and Salil P. Vadhan. An equivalence between zero knowledge and commitments. In *TCC*, pages 482–500, 2008.

[46] Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Interactive hashing simplifies zero-knowledge protocol design. In *EUROCRYPT*, pages 267–273, 1993.

[47] Amit Sahai and Salil P. Vadhan. A complete problem for statistical zero-knowledge. *J. ACM*, 50(2):196–249, 2003.

[48] Saurabh Sanghvi and Salil P. Vadhan. The round complexity of two-party random selection. In *STOC*, pages 338–347, 2005.

[49] Adi Shamir. IP = PSPACE. *J. ACM*, 39(4):869–877, 1992.

[50] Martin Tompa and Heather Woll. Random self-reducibility and zero-knowledge interactive proofs of possession of information. In *28th FOCS*, pages 472–482, 1987.

[51] Salil P. Vadhan. *A study of statistical zero-knowledge proofs*. PhD thesis, MIT, 1999.

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

A The problem UNIFORM

We give the definition of the **NIPZK**-complete problem of [34], called UNIFORM (UN). Given a circuit X, we use the convention that $n+1$ denotes the number of output bits of X. We need the following notation.

- T_X is the set of outputs of X that end with a 1. Formally, $T_X \overset{\text{def}}{=} \{x | \exists r\ X(r) = x,$ and the suffix of x is $1\}$. Informally, when a problem is reduced to UN, its soundness and completeness properties imply that the size of T_X is large for YES instances of UN, and small for NO instances of UN.
- X' is the distribution on the first n bits that X outputs. That is, X' is obtained from X by taking a random sample of X, and then outputting the first n bits. Again, when a YES instance of a **NIPZK** problem is reduced to the circuit X, the distribution X' is uniform on $\{0,1\}^n$.

Now, letting X be a circuit with $n+1$ output bits, we say that X is β-*negative* if $|T_X| \leq \beta \cdot 2^n$. That is, T_X is small, and contains at most $\beta \cdot 2^n$ strings. We say that X is α-*positive* if X' is the uniform distribution on $\{0,1\}^n$ and $\Pr_{x \leftarrow X}[x \in T_X] \geq \alpha$. This implies that T_X is large, and contains at least $\alpha \cdot 2^n$ strings.

Definition A.1 *The problem* UNIFORM *is defined as* $\mathrm{UN} \overset{def}{=} \langle \mathrm{UN_Y}, \mathrm{UN_N} \rangle$*, where*

$$\begin{aligned} \mathrm{UN_Y} &= \{X | X \textit{ is } 2/3 - \textit{positive}\}\,, \textit{ and} \\ \mathrm{UN_N} &= \{X | X \textit{ is } 1/3 - \textit{negative}\}\,. \end{aligned}$$

ECCC ISSN 1433-8092

http://eccc.hpi-web.de/