Statistical Concurrent Non-malleable Zero Knowledge

Claudio Orlandi[1]*, Rafail Ostrovsky[23], Vanishree Rao[2], Amit Sahai[2], and Ivan Visconti[4]

[1] Department of Computer Science, Aarhus University, Denmark
orlandi@cs.au.dk
[2] Department of Computer Science, UCLA, USA
[3] Department of Mathematics, UCLA, USA
{rafail,vanishri,sahai}@cs.ucla.edu
[4] Dipartimento di Informatica, University of Salerno, Italy
visconti@unisa.it

Abstract. The notion of Zero Knowledge introduced by Goldwasser, Micali and Rackoff in STOC 1985 is fundamental in Cryptography. Motivated by conceptual and practical reasons, this notion has been explored under stronger definitions. We will consider the following two main strengthened notions.

Statistical Zero Knowledge: here the zero-knowledge property will last forever, even in case in future the adversary will have unlimited power.

Concurrent Non-Malleable Zero Knowledge: here the zero-knowledge property is combined with non-transferability and the adversary fails in mounting a concurrent man-in-the-middle attack aiming at transferring zero-knowledge proofs/arguments.

Besides the well-known importance of both notions, it is still unknown whether one can design a zero-knowledge protocol that satisfies both notions simultaneously.

In this work we shed light on this question in a very strong sense. We show a *statistical concurrent non-malleable* zero-knowledge argument system for $\mathcal{NP}$ with a *black-box* simulator-extractor.

1 Introduction

The notion of zero knowledge, first introduced in [10], is one of the most pivotal cryptographic constructs. Depending on both natural and real-world attack scenarios, zero knowledge has been studied considering different conceptual flavors and practical applications.

* Work done while visiting UCLA.

Y. Lindell (Ed.): TCC 2014, LNCS 8349, pp. 167–191, 2014.

Zero Knowledge and Man-in-the-Middle Attacks. In distributed settings such as the Internet, an adversary that controls the network can play concurrently as a verifier in some proofs[1] and as a prover in the other proofs. The goal of the adversary is to exploit the proofs it receives from the provers to then generate new proofs for the verifiers. The original notion of zero knowledge does not prevent such attacks since it assumes the adversarial verifier to only play as a verifier and only in sequential sessions.

The need of providing non-transferable proofs secure against such man-in-the-middle (MiM, for short) attacks was first studied by Dolev, Dwork and Naor in [7]. In [1], Barak, Prabhakaran and Sahai achieved for the first time such a strong form of zero knowledge, referred to as concurrent non-malleable zero knowledge (CNMZK, for short) is possible in the plain model. They provide a poly(λ)-round construction, for λ being the security parameter, based on one-way functions, and a $O(\log(\lambda))$-round construction based on collision-resistant hash functions. More recent results focused on achieving round efficiency with a mild setup [23], computationally efficient constructions [22], security with adaptive inputs [16].

Zero Knowledge and Forward Security. The zero-knowledge property says that the view of the adversarial verifier does not help her in gaining any useful information. This means that it does not include information that can be exploited by a PPT machine. However, even though the execution of a zero-knowledge protocol can be based on the current hardness of some complexity assumptions, it is quite risky to rely on the assumed resilience of such assumptions against more powerful machines of the future. What is zero knowledge in a transcript produced today could not be zero knowledge in the eyes of a distinguisher that will read the transcript in 2040.

It is therefore appealing to provide some forward security flavor so that whatever is zero knowledge today will be zero knowledge forever. Statistical zero knowledge [2,25,21,9,20,12,19] is the notion that satisfies this requirement. It has been achieved in constant rounds using collision-resistant hash functions [14], and even under the sole assumption that one-way functions exist requiring more rounds [13].

Unfortunately, all the known constructions for CNMZK protocols strongly rely on the computational indistinguishability of the output of the simulator. Techniques so far used to design protocols that are then proved to be CNMZK require the protocol to fix a witness in a commitment, that therefore must be statistically binding and thus only computationally hiding. There is therefore no hope to prove those protocol to be statistical zero knowledge. Moreover it does not seem that minor changes can establish the statistical zero knowledge property still allowing to prove CNMZK.

[1] While in our general discussion, we often refer to zero-knowledge proofs, we will finally need to resort to only arguments since our goal is to achieve statistical zero-knowledge property.

The Open Problem. Given the above state-of-the-art a natural question is the following: *is it possible to design an argument system that combines the best of both worlds, namely, a statistical concurrent non-malleable zero-knowledge argument system?*

1.1 Our Contribution

In this work, we provide the first statistical concurrent non-malleable zero-knowledge argument system. Our construction is an argument of knowledge (AoK, for short) and has a black-box simulator-extractor producing a statistically indistinguishable distribution.

As mentioned earlier, Barak et al. [1] presented the first CNMZKAoK protocol; we will refer to their work here as BPS. However, their construction had an inherent limitation that the simulation can only be computational, the reason being the following. In their protocol, the prover needs to commit to a valid witness via a statistically binding non-malleable commitment scheme. The commitment scheme being statistically binding is extremely crucial in their proof of security. This implies that when the simulator cheats and commits to a non-witness, the simulated view can only be computationally indistinguishable and not statistically so.

In this work, we overcome this shortcoming with the following idea. We take the BPS argument as a starting point and modify it. Firstly, we work on the root of the problem – the non-malleable commitment. We replace it with a special kind of a commitment scheme called '*mixed non-malleable commitment*' scheme. The notion of mixed commitment was first introduced by Damgård and Nielsen [6]. Our mixed non-malleable commitment is parameterized by a string that if sampled with uniform distribution makes the scheme statistically hiding and computationally binding. Instead, when it is taken from another (computationally indistinguishable) distribution it is a statistically binding, computationally hiding, and non-malleable. We will construct such a scheme by using as distributions non-DDH and DDH tuples.

The next idea would be to append the (modified) BPS argument to a coin-flipping phase in which the prover and the verifier generate a random string. Thus, in the real-world the above mixed commitment is statistically hiding. This thus enables us to prove statistical simulatability of our protocol. Furthermore, in order to also achieve extractability of witnesses for the arguments given by the adversary, we switch to a hybrid which biases the coin-flipping outcome to a random DDH tuple. Typically, a coin-flipping protocol would involve the verifier committing to its share of randomness, the prover sending its share of randomness in the clear, and finally, the verifier opening the commitment. However, in order to enable the simulator to bias the outcome, instead of the verifier opening the commitment to its share of randomness, it gives only the committed value in the clear and presents an AoK for the randomness used. This argument is again played by using the BPS AoK, since we would need concurrent non-malleability here.

In order to simplify our proofs, we rely on the Robust Extraction Lemma of Goyal et al. [11] that generalizes concurrent extractability of the PRS preamble (or concurrently extractable commitments – CECom, for short) [24] in the following sense. Consider an adversary who sends multiple CECom commitments interleaving them arbitrarily and also interacts with an external party B in an arbitrary protocol. Then, [11] shows how to perform concurrent extraction of the CECom commitments without rewinding the external party B. The extractor designed by them is called the 'robust simulator'.

Technical Challenges. While we will encounter multiple technical challenges, which will be clear as we go ahead, we point out the core technical challenge here and the way we will solve it.

One of the main technical challenges is when we prove witness extractability of our protocol. Namely, in our hybrid argument, we will encounter two consecutive hybrids H_a and H_b, wherein a coin-flipping phase of a particular right hand session is 'intact' in H_a, but is biased in H_b. This results in the mixed commitment changing from statistically hiding to statistically binding. In order to finally be able to argue that the extracted values are indeed valid witnesses, we will need to argue for the hybrid H_b that the value committed in this commitment is a valid witness. Herein, we will need to reduce our claim to computational binding of a CECom commitment in the protocol. Thus, the requirement in this reduction would be that no extraction performed should rewind the external CECom sender. Even the Robust Extraction Lemma will not be helpful here as the Lemma requires that the external protocol have round complexity strictly less than the round complexity of CECom commitments (on which the robust simulator performs extraction) and the external protocol in this case is a CECom commitment itself. The condition for the Lemma thus cannot be met. We get around this difficulty through a carefully designed sequence of hybrid arguments. A similar difficulty arises in the proof of statistical simulatability of our protocol. Here again, we rely on a carefully designed sequence of hybrids.

The second main technical challenge, still of the same flavor as the first one above, is in the proof of witness extractability. Here, we encounter a pair of hybrids: in the former hybrid, we would have a few CECom commitments of the right session being extracted by the robust simulator; in the latter hybrid, the modification introduced would be to change the value committed in a (statistically hiding) CECom commitment of a left session from a valid witness to a zero-string. Here again, we will not be able to argue a reduction to the hiding property of the CECom commitment of the left session in question, just by relying on the Robust Extraction Lemma. Here, we instead present a more detailed hybrid argument. Namely, in the CECom commitment, we change the committed value one sub-commitment at a time [24]. Since every sub-commitment in the standard CECom commitment of [24] ranges over just three rounds, we are now still able to apply the Robust Extraction Lemma.

2 Background

We assume familiarity with interactive Turing machines, denoted ITM. Given a pair of ITMs, A and B, we denote by $\langle A(x), B(y)\rangle(z)$ the random variable representing the (local) output of B, on common input z and private input y, when interacting with A with private input x, when the random tape of each machine is uniformly and independently chosen. In addition, we denote $\mathsf{view}_B^A(x, z)$ to be the random variable representing the content of the random tape of B together with the messages received by B from A during the interaction on common input x and auxiliary input z to B.

If $\mathcal{D}_1$ and $\mathcal{D}_2$ are two distributions, then we denote that they are statistically close by $\mathcal{D}_1 \approx_s \mathcal{D}_2$; we denote that they are computationally indistinguishable by $\mathcal{D}_1 \approx_c \mathcal{D}_2$; and we denote that they are identical by $\mathcal{D}_1 \equiv \mathcal{D}_2$.

Definition 1 (Pseudorandom Language). *An NP-language $L \subseteq \{0,1\}^*$ is said to be a pseudorandom language if the following holds. For $\lambda \in \mathbb{N}$, let $\mathcal{D}_\lambda$ be a uniform distribution over $L \cap \{0,1\}^\lambda$. Then, for every distinguisher $\mathcal{D}$ running in time polynomial in λ, there exists a negligible function* $\mathsf{negl}(\cdot)$ *such that $\mathcal{D}$ can distinguish between $\mathcal{D}_\lambda$ and U_λ with probability at most* $\mathsf{negl}(\lambda)$.

We assume familiarity with notions like witness relation, interactive argument systems, and statistical witness-indistinguishable argument of knowledge (sWIAoK).

The verifier's view of an interaction consists of the common input x, followed by its random tape and the sequence of prover messages the verifier receives during the interaction. We denote by $\mathsf{view}_{\mathcal{V}^*}^{\mathcal{P}}(x, z)$ a random variable describing $\mathcal{V}^*(z)$'s view of the interaction with $\mathcal{P}$ on common input x.

We will use various forms of commitment schemes. We will denote by SB, SH, CB, CH the usual properties that can be enjoyed by classic commitment schemes, namely: statistical binding, statistical hiding, computational binding and computational hiding.

Statistical Concurrent Non-malleable Zero Knowledge. The definition of statistical CNMZK is taken almost verbatim from [1] except for the additional requirement on the simulation being statistical. Let $\langle \mathcal{P}, \mathcal{V}\rangle$ be an interactive proof for an NP-language L with witness relation R_L, and let λ be the security parameter. Consider a man-in-the-middle adversary $\mathcal{M}$ that participates in m_L "left interactions" and m_R "right interactions" described as follows. In the left interactions, the adversary $\mathcal{M}$ interacts with $\mathcal{P}_1, \ldots, \mathcal{P}_{m_L}$, where each $\mathcal{P}_i$ is an honest prover and proves the statement $x_i \in L$. In the right interactions, the adversary proves the validity of statements $\overline{x}_1, \ldots, \overline{x}_{m_R}$. Prior to the interactions, both $\mathcal{P}_1, \ldots, \mathcal{P}_{m_L}$ receive $(x_1, w_1), \ldots, (x_{m_L}, w_{m_L})$, respectively, where for all i, $(x_i, w_i) \in R_L$. The adversary $\mathcal{M}$ receives $x_1, \ldots, x_{m_L}$ and the auxiliary input z, which in particular might contain a-priori information about $(x_1, w_1), \ldots, (x_{m_L}, w_{m_L})$. On the other hand, the statements proved in the right interactions $\overline{x}_1, \ldots, \overline{x}_{m_R}$ are chosen by $\mathcal{M}$. Let $\mathsf{view}_{\mathcal{M}}(x_1, \ldots, x_{m_L}, z)$ denote a random variable that describes the view of $\mathcal{M}$ in the above experiment. Loosely speaking, an interactive argument is statistical concurrent non-malleable zero-knowledge (sCNMZK) if for every man-in-the-middle adversary $\mathcal{M}$, there exists a probabilistic polynomial time machine (called the simulator-extractor) that can *statistically* simulate both the left and the right interactions for $\mathcal{M}$, while outputting a witness for every statement proved by the adversary in the right interactions.

Definition 2 ((Black-Box) Statistical Concurrent Non-Malleable Zero Knowledge Argument of Knowledge). *An interactive protocol $\langle \mathcal{P}, \mathcal{V} \rangle$ is said to be a* (Black-Box) Statistical Concurrent Non-Malleable Zero Knowledge *(sCNMZK) argument of knowledge for membership in an NP language L with witness relation R_L, if the following hold:*

1. *$\langle \mathcal{P}, \mathcal{V} \rangle$ is an interactive argument system;*
2. *For every m_L and m_R that are polynomial in λ, for every PPT adversary $\mathcal{M}$ launching a concurrent non-malleable attack (i.e., $\mathcal{M}$ interacts with honest provers $\mathcal{P}_1, \ldots, \mathcal{P}_{m_L}$ in "left sessions" and honest verifiers $\mathcal{V}_1, \ldots, \mathcal{V}_{m_R}$ in "right sessions"), there exists an expected polynomial time simulator-extractor $\mathcal{SE}$ such that for every set of "left inputs" $x_1, \ldots, x_{m_L}$ we have $\mathcal{SE}(x_1, \ldots, x_{m_L}) = (\mathsf{view}, \overline{w}_1, \ldots, \overline{w}_{m_R})$ such that:*
   - view *is the simulated joint view of $\mathcal{M}$ and $\mathcal{V}_1, \ldots, \mathcal{V}_{m_R}$. Further, for any set of witnesses $(w_1, \ldots, w_{m_L})$ defining the provers $\mathcal{P}_1, \ldots, \mathcal{P}_{m_L}$, the view* view *is distributed statistically indistinguishable from the view of $\mathcal{M}$, denoted* $\mathsf{view}_{\mathcal{M}}(x_1, \ldots, x_{m_L}, z)$*, in a real execution;*
   - *In the view* view*, let trans_ℓ denote the transcript of ℓ-th left execution, and $\overline{\mathsf{trans}}_{\mathrm{t}}$ that of* t*-th right execution, $\ell \in [m_L]$,*$\mathrm{t} \in [m_R]$*. If $\overline{x}_{\mathrm{t}}$ is the common input in $\overline{\mathsf{trans}}_{\mathrm{t}}$, $\overline{\mathsf{trans}}_{\mathrm{t}} \neq \mathsf{trans}_\ell$ (for all ℓ) and $\mathcal{V}_{\mathrm{t}}$ accepts, then $R_L(\overline{x}_{\mathrm{t}}, \overline{w}_{\mathrm{t}}) = 1$ except with probability negligible in λ.*

   The probability is taken over the random coins of $\mathcal{SE}$. Further, the protocol is black-box sCNMZK*, if $\mathcal{SE}$ is a universal simulator that uses $\mathcal{M}$ only as an oracle, i.e., $\mathcal{SE} = \mathcal{SE}^{\mathcal{M}}$.*

We remark here that the statistical indistinguishability is considered only against computationally unbounded distinguishers, and not against unbounded man-in-the-middle adversaries. This restriction is inherent to the definition since we require statistical zero-knowledge and thus cannot simultaneously ask for soundness against unbounded provers.

Extractable Commitment Schemes.

Definition 3 (Extractable Commitment Schemes). *An extractable commitment scheme* $\langle$Sender, Receiver$\rangle$ *is a commitment scheme such that given oracle access to any PPT malicious sender* Sender**, committing to a string, there exists an expected PPT extractor* E *that outputs a pair (τ, σ^*) such that the following properties hold:*

Simulatability. The simulated view τ is identically distributed to the view of Sender* *(when interacting with an honest* Receiver*) in the commitment phase.*

Extractability. the probability that τ is accepting and σ^* correspond to $\perp$ is at most $1/2$. Moreover if $\sigma^* \neq \perp$ then the probability that Sender* *opens τ to a value different than σ^* is negligible.*

Lemma 1. *[15]* $\mathsf{Com}_{\mathsf{nm}}$ *is an extractable commitment scheme.*

As shown in [15], $\mathsf{Com}_{\mathsf{nm}}$ is an extractable commitment scheme. This is in fact the core property of the scheme that is relied upon in proving its non-malleability in [8,15].

Extractable Mixed Robust Non-malleable Commitments w.r.t. 1-Round Protocols. In our protocol we make use of a special kind of commitment scheme, that we call a *extractable mixed robust non-malleable commitment scheme.* These are basically the mixed commitment schemes introduced by Damgård and Nielsen [6] that are also non-malleable (or robust) not only w.r.t. themselves but also w.r.t. 1-round protocols and also extractable.

We shall first discuss how we get mixed non-malleable commitments, and then at the end, we shall discuss how we also get mixed non-malleable commitments that are also robust w.r.t. 1-round protocols.

Intuitively, a mixed non-malleable commitment scheme is a commitment scheme that is parameterized by a string srs in such a way that if srs is from some specific distribution, then commitment scheme is SH, and if srs is from another specific indistinguishable distribution, then the scheme is non-malleable. We require that both the distributions be efficiently samplable. When srs is randomly sampled (from the dominion over which both the distributions are defined), we would require that srs is such that with all but negligible probability the scheme is SH. We denote such a scheme by $\mathsf{NMMXCom}_{\mathsf{srs}}$. More formally:

Definition 4 (Mixed Non-Malleable Commitments). *A commitment scheme is said to be a mixed non-malleable commitment scheme if it is parameterized by a string* srs *and if there exist two efficiently samplable distributions $\mathcal{D}_1$, $\mathcal{D}_2$, such that, $\mathcal{D}_1 \approx_c \mathcal{D}_2$, and if* $\mathsf{srs} \leftarrow \mathcal{D}_1$ *then the commitment scheme is SH and if* $\mathsf{srs} \leftarrow \mathcal{D}_2$ *then the commitment scheme is non-malleable. Furthermore,* $|\mathrm{Supp}(\mathcal{D}_2)|/|\mathrm{Supp}(\mathcal{D}_1)| = \mathsf{negl}(\lambda)$.

Below, we show how to construct such a scheme. At a high level, we achieve this by using a *mixed commitment scheme* which, roughly speaking, is a commitment scheme parameterized by a string srs in such a way that if srs is from some specific efficiently samplable distribution, then commitment scheme is SH, and if srs is from another specific indistinguishable efficiently samplable distribution, then the scheme is SB. We denote such a scheme by $\mathsf{MXCom}_{\mathsf{srs}}$. More formally:

Definition 5 (Mixed Commitments). *A commitment scheme is said to be a mixed commitment scheme if it is parameterized by a string* srs *and if there exist two efficiently samplable distributions $\mathcal{D}_1$, $\mathcal{D}_2$, such that, $\mathcal{D}_1 \approx_c \mathcal{D}_2$, and if* $\mathsf{srs} \leftarrow \mathcal{D}_1$ *then the commitment scheme is SH and if* $\mathsf{srs} \leftarrow \mathcal{D}_2$ *then the commitment scheme is SB. Furthermore,* $|\mathrm{Supp}(\mathcal{D}_2)|/|\mathrm{Supp}(\mathcal{D}_1)| = \mathsf{negl}(\lambda)$.

In [6], Damgård and Nielsen gave two constructions of mixed commitment schemes, one based on one based on the Paillier cryptosystem and the other based on the Okamoto-Uchiyama cryptosystem. For concreteness, we provide a construction below based on Σ-protocols and that builds on previous ideas presented in [5,3,4].

Constructing Mixed Commitments. Let us first describe how to construct a mixed commitment scheme. The idea is to have $\mathcal{D}_1$ be uniform over $\{0,1\}^{\mathrm{poly}(\lambda)}$ and $\mathcal{D}_2$ be uniform over a pseudorandom language L (as per Definition 1) with a Σ-protocol (i.e., public-coin 3-round special-sound special honest-verifier zero-knowledge proof system). Then, to commit to a value β, sender would first run the simulator of the Σ-protocol for the statement that $\mathsf{srs} \in L$ such that the simulated proof has β as the challenge; let (α, β, γ) be the simulated proof. Then the commitment would just be α. The opening would be γ.

Observe that if $\mathsf{srs} \notin L$, then for any β there is only one accepting (α, β, γ), making the scheme parameterized by this srs to be SB. Furthermore, with srs sampled uniformly at random from $\{0,1\}^* \setminus L$, we will also be able to argue that the resulting scheme is CH. On the other hand, if $\mathsf{srs} \in L$, then, for every α (in its valid domain as defined by the Σ-protocol), there exists γ' for every β' such that $(\alpha, \beta', \gamma')$ is an accepting transcript. This implies that there exists an opening of α to any β'. This makes the scheme SH. Furthermore, with srs sampled uniformly at random from L, it shall hold for any PPT machine that it can only run the simulator and it is infeasible for the machine to open α to *also* any $\beta' \neq \beta$ (with some γ' as an opening), assuming special-soundness of the Σ-protocol (Otherwise, one could extract the witness from $(\alpha, \beta, \gamma, \beta', \gamma')$). This makes the system only computationally binding. In detail:

Mixed Commitment from Σ-protocol. Let R_L be a hard relation for a pseudorandom language L i.e., $L = \{\mathsf{srs} \in \{0,1\}^\lambda | \; \exists w : R_L(\mathsf{srs}, w) = 1\}$ and $L \approx_c U_\lambda$. Consider a Σ-protocol for the above language L. The special honest-verifier zero-knowledge property of the Σ-protocol implies existence of a simulator S that on input the instance srs, a string β and a randomness r, outputs a pair (α, γ) such that $(\mathsf{srs}, \alpha, \beta, \gamma)$ is computationally indistinguishable from a transcript $(\mathsf{srs}, \alpha, \beta, \gamma)$ played by the honest prover when receiving β as challenge.

The commitment scheme played by sender C and receiver R that we need goes as follows.

Shared Random String: A random string $\mathsf{srs} \in \{0,1\}^\lambda$ is given as a common input to both the parties;

Commitment Phase: We denote the commitment function by $\mathsf{MXCom}_{\mathsf{srs}}(\cdot;\cdot)$ and to commit to a string $\beta \in \{0,1\}^\lambda$:

1. C runs the Σ-protocol simulator $S(\mathsf{srs}, \beta, r)$ to obtain (α, γ);
2. C sends α to R;

Decommitment Phase: To open α to β:

1. C sends (β, γ) to R;
2. R accepts if $(\mathsf{srs}, \alpha, \beta, \gamma)$ is an accepting transcript for the Σ-protocol.

If $\mathsf{srs} \in L$, then the commitment is computationally binding (since, with two openings one gets two accepting conversations for the same α, and from the special-soundness property of the Σ-protocol one can extract the witness) and statistically hiding (which is directly implied by perfect completeness of the Σ-protocol; i.e., for any α output as the first message by the simulator – for any β as the challenge – for every β', given the witness, one can efficiently compute a final message γ' such that the verifier accepts). If $\mathsf{srs} \notin L$ the commitment is statistically binding (since, for any α, there exists at most one β that makes R accept the decommitment, as there is no witness for $\mathsf{srs} \in L$ and two accepting transcripts (α, β, γ), $(\alpha, \beta', \gamma')$ with $\beta \neq \beta'$ implies a witness owing to the special-soundness property of the Σ-protocol) and computationally hiding (since, if on input α, one can guess β efficiently, then this can be used to decide whether or not $\mathsf{srs} \in L$, a contradiction).

While there are many instantiations for L, we shall work with the following simple one. Define $L = \{(g_1, g_2, g_3, g_4) \in \mathbb{G}^4 | \ \exists a, b : a \neq b \wedge g_1^a = g_2 \wedge g_3^b = g_4\}$ with $\mathbb{G}$ being a prime order group, where DDH is believed to be hard. That is, L is the language of non-DDH triplets. Note that in this case if srs is chosen uniformly at random from $\mathbb{G}^4$ the commitment is statistically hiding with overwhelming probability (most strings are not DDH triplets).

Relaxing the Assumption. Another example for L is the following language: let (G, E, D) be a *dense* cryptosystem (i.e., valid public keys and ciphertexts can be easily extracted from random strings). The language L is:

$$L = \{(pk_0, pk_1, c_0, c_1) | \exists r_0, r_1, m_0, m_1, s_0, s_1 : m_0 \neq m_1, (pk_0, sk_0) \leftarrow G(1^k, r_0),$$

$$c_0 = E_{pk_0}(m_0, s_0), (pk_1, sk_1) \leftarrow G(1^k, r_1), c_1 = E_{pk_1}(m_1, s_1))\}.$$

Also in this case most strings are in the language, while the simulator can choose a string not in the language (i.e., with $m_0 = m_1$).

Moreover, we can plug this mixed commitment MXCom in a zero-knowledge protocol in the SRS model $\mathsf{NMMXCom}$, so that when srs is a random DDH triple, the zero-knowledge protocol is a proof (i.e., statistically sound) and computational zero-knowledge, while when the srs is a random non-DDH triple then the zero-knowledge protocol is statistical zero-knowledge (and computationally sound). For eg., an implementation of Blum's protocol by using MXCom as commitment scheme when the prover commits to the permuted adjacency matrices gives us a computational zero-knowledge proof-of-knowledge (ZKPoK, for short) if srs of the MXCom commitment used is a random DDH tuple and a statistical zero-knowledge argument-of-knowledge (ZKAoK, for short) if the srs is a random non-DDH tuple.

Constructing Mixed Non-malleable Commitments. As mentioned earlier, we show how to construct a mixed non-malleable commitment scheme by using a mixed commitment scheme. For concreteness, we shall work with the mixed commitment scheme MXCom described earlier. To thus recall, by the construction of MXCom, our mixed non-malleable commitment scheme will be non-malleable when srs is a random DDH tuple and, is statistically hiding and computationally binding when srs is a random non-DDH tuple.

Our scheme $\mathsf{NMMXCom}_{\mathsf{srs}}$ is described as follows. At a high level, our approach is to slightly modify the DDN non-malleable commitment scheme in [8]. In fact, we shall describe our modification by considering the concurrent non-malleable commitment scheme that appears in [15] (whose analysis of non-malleability is similar to that of the DDN commitment and is simpler). The protocol in [15] is in fact non-malleable w.r.t. any arbitrary protocols of logarithmic round-complexity, a property that is called $\log(\lambda)$-robust non-malleability. This is one of the properties which will be of a crucial use to us and we shall elaborate on this property shortly. In fact, we only need 1-robust non-malleability. The scheme of [15] is described below.

Common Input : An identifier $\mathrm{ID} \in \{0,1\}^L$, where $L = \mathrm{poly}(\lambda)$. Define $\ell := \log(L) + 1$.

Input for Sender : A string $V \in \{0,1\}^\lambda$.

Sender ← Receiver: Sender chooses $V_1, V_2, \ldots, V_L \leftarrow \{0,1\}^\lambda$ such that $V_1 \oplus V_2 \oplus \ldots \oplus V_L = V$. For each $i \in [L]$, run Stage 1 and Stage 2 in parallel with $v := V_i$ and $\mathrm{id} = (i, \mathrm{ID}_i)$, where ID_i is the i-th bit of ID.

Stage 1 :

Sender ← Receiver: Receiver samples $x \leftarrow \{0,1\}^\lambda$, computes $y = f(x)$, and sends s to Sender. Sender aborts if y is not in the range of f.

Sender → Receiver: Sender chooses $\mathsf{randomness} \leftarrow \{0,1\}^\lambda$ and sends $c = \mathsf{Com}_{sb}(v; \mathsf{randomness})$.

Stage 2 :

Sender → Receiver: 4ℓ special-sound $\mathcal{WI}$ proofs of the statement:
either there exists values v, $\mathsf{randomness}$ such that $c = \mathsf{Com}_{sb}(v; \mathsf{randomness})$
or there exists a value x such that $y = f(x)$
with 4ℓ $\mathcal{WI}$ proofs in the following schedule:
For $j = 1$ to ℓ do: Execute $\mathrm{design}_{\mathrm{id}_j}$ followed by $\mathrm{design}_{1-\mathrm{id}_j}$.

Fig. 1. $O(\log(\lambda))$-round Non-Malleable Commitment of [15]

At a high level, the protocol of the sender who wishes to commit to some value v proceeds as follows. To catch the core of the intuition, we describe here a simplified version of the protocol while ignoring the currently unnecessary details (such as parallel repetitions, etc.); later in the formal description, we shall present the original protocol of [15]. The sender proceeds as follows. In the first stage, upon receiving an output of a one-way function from the receiver, commit to v using a statistically binding commitment scheme Com_{sb}. In the second stage, engage in $\log(\lambda)$ (special-sound) $\mathcal{WI}$ proofs of knowledge of either the value committed to using Com_{sb} or of a pre-image of the one-way function output sent by the receiver. (The number of $\mathcal{WI}$ proofs is logarithmic in the length of the identities of the senders; hence, it is considered to be $\log(\lambda)$ in general). We note here that a special-sound $\mathcal{WI}$ proof can be instantiated by using Blum's Hamiltonicity protocol, wherein the commitment sent by the $\mathcal{WI}$ prover in this protocol is SB.

Now to construct the mixed non-malleable commitment, the idea is to replace the SB commitment Com_{sb} of the first stage and the SB commitment within the Blum's Hamiltonicity protocol (where both the commitments are given by the sender to the receiver) with the mixed commitment $\mathsf{MXCom}_{\mathsf{srs}}$. We shall analyze the properties of the resulting commitment scheme, denoted by $\mathsf{NMMXCom}_{\mathsf{srs}}$, below.

Recall that if srs is a random DDH tuple, then $\mathsf{MXCom}_{\mathsf{srs}}$ is SB and CH. Under this case, the resulting scheme would have the properties identical to the original scheme of [18]; namely it is SB, CH, and non-malleable. On the other hand, if srs is a random non-DDH tuple, then $\mathsf{MXCom}_{\mathsf{srs}}$ is SH and CB. This would render the the resulting scheme to be SH (owing to the SH property of the commitment scheme in the first phase and witness-indistinguishability of the Hamiltonicity protocol that is instantiated with SH commitment) and CB (owing to the computational binding property of the commitment scheme in the first phase; this is due to the fact that decommitment of the scheme in [15] is simply an opening of the commitment of the first phase). In fact, if srs is a random string, then it is a non-DDH tuple with all but negligible probability. Hence, we also have that when srs is a random string, $\mathsf{MXCom}_{\mathsf{srs}}$ is SH and CB with all but negligible probability. For future reference, we shall bookmark this into the following proposition.

Proposition 1. *If* srs *is a uniform DDH tuple, then* $\mathsf{MXCom}_{\mathsf{srs}}$ *is SB, CH, and non-malleable. If* srs *is a uniform random string, then* $\mathsf{MXCom}_{\mathsf{srs}}$ *is SH and CB.*

Robustness w.r.t. 1-Round Protocols of the Mixed Non-Malleable Commitments. Recall that we modified the [15] non-malleable commitment scheme that is robust w.r.t. 1-round protocols to get mixed non-malleable commitment scheme. It turns out that the modified scheme still retains robust w.r.t. 1-round protocols. Here, we only give a high-level description of the reason behind this fact as this can be easily verified. The reason is that robustness of the non-malleable commitment scheme in Figure 1 is proved in [15] by relying only upon the structure (the 'designs', in particular) of the commitment scheme in Figure 1. In particular, this proof does not rely upon the specifics of the underlying commitment scheme. Now recall that the only modification we introduced in the robust non-malleable commitment scheme of [15] to get a mixed non-malleable commitment scheme is the following. Instead of using any underlying commitment scheme, we used a mixed commitment scheme. Thus, the scheme continues to be non-malleable commitment scheme robust w.r.t. 1-round protocols even when the underlying commitment schemes are mixed commitments.

Non-malleability of $\mathsf{NMMXCom}_{\mathsf{srs}}$ *w.r.t.* $\mathsf{Com}_{\mathsf{nm}}$. Another property of $\mathsf{NMMXCom}_{\mathsf{srs}}$ that we need is the following. Let $\mathsf{Com}_{\mathsf{nm}}$ be the NMCom commitment robust w.r.t. 1-round protocol. We shall argue below that $\mathsf{NMMXCom}_{\mathsf{srs}}$ is non-malleable w.r.t. $\mathsf{Com}_{\mathsf{nm}}$.

Proposition 2. *The non-malleable commitment* $\mathsf{NMMXCom}_{\mathsf{srs}}$ *is robust w.r.t. the non-malleable commitment* $\mathsf{Com}_{\mathsf{nm}}$*.*

Proof Sketch. Essentially, the proof is exactly the same as the proof of non-malleability of the non-malleable commitment scheme of [15] presented in Figure 1. We argue this here next. Consider a MiM adversary against non-malleability of $\mathsf{NMMXCom}_{\mathsf{srs}}$ that executes a $\mathsf{Com}_{\mathsf{nm}}$ session on the left by playing the role of the receiver and a $\mathsf{NMMXCom}_{\mathsf{srs}}$ session on the right by playing the role of a sender. The key technique in proving non-malleability in [8,18,15] is to show that, immaterial of the way a MiM adversary interleaves the left and right commitments, there exists at least one $\mathcal{WI}$ proof (within some design) on the right session such that it is 'safe' to rewind the MiM adversary for this proof; by 'safe', we mean that rewinding the MiM adversary at this point can be done without rewinding the external sender on the left. (Recall that to rewind a $\mathcal{WI}$ proof is to rewind to the point between the first and the second message of the proof). To then understand what $\mathcal{WI}$ proof qualifies to be safe to rewind, we begin by giving a high level idea of when a proof *does not* qualify to be safe. Consider any $\mathcal{WI}$ proof $(\alpha_{\mathrm{r}}, \beta_{\mathrm{r}}, \gamma_{\mathrm{r}})$ on the right. If it is trying to use and 'maul' some $\mathcal{WI}$ proof $(\alpha_{\mathrm{l}}, \beta_{\mathrm{l}}, \gamma_{\mathrm{l}})$ on the left, then the right proof is positioned in time with respect to the left one as shown in Figure 2. Observe that rewinding such a proof on the right with a new challenge may make the MiM adversary send a new challenge for the left proof too asking for a new response which tantamounts to rewinding the sender on the left. [8,18,15] provide a characterization for the $\mathcal{WI}$ proofs on the right that qualify as safe for being rewound; however, the details of this characterization itself will not be important to us; the core argument in proving non-malleability in [8,18,15] is an argument that, immaterial of the way a MiM adversary interleaves the left and right commitments, there exists a $\mathcal{WI}$ proof on the right that is safe to rewind. This is so owing to the fact that the adversary can use only one proof on the left for every proof on the right and to the fact that there are exactly the same number of proofs on the left and the right. This would imply that if the left and the right identities are distinct (at least at one bit position), then at proofs corresponding to this bit position, design_0 on the left 'matches up' with design_1 on the right, depicted in Figure 2. With a closer look at this interleaving, it can be easily derived that at least one of the $\mathcal{WI}$ proofs within this design_1 on the right is safe to be rewound.

We first observe that the only way $\mathsf{NMMXCom}_{\mathsf{srs}}$ differs from $\mathsf{Com}_{\mathsf{nm}}$ in Figure 1 is that a specific kind of commitment, namely, a mixed commitment is used to instantiate the underlying commitments used in building $\mathsf{Com}_{\mathsf{nm}}$ in Figure 1. Next, we observe that non-malleability of the commitment scheme $\mathsf{NMMXCom}_{\mathsf{srs}}$ is mainly due to the structure (or designs) of the $\mathcal{WI}$ proofs, and the same arguments on interleaving and safety of rewinding would hold even if the left commitment is under an $\mathsf{Com}_{\mathsf{nm}}$ session. □

We remark that in fact the non-malleable commitments $\mathsf{NMMXCom}_{\mathsf{srs}}$ and $\mathsf{Com}_{\mathsf{nm}}$ are robust w.r.t. each other by the same arguments as above. However, it suffices for us that $\mathsf{NMMXCom}_{\mathsf{srs}}$ is robust w.r.t. $\mathsf{Com}_{\mathsf{nm}}$.

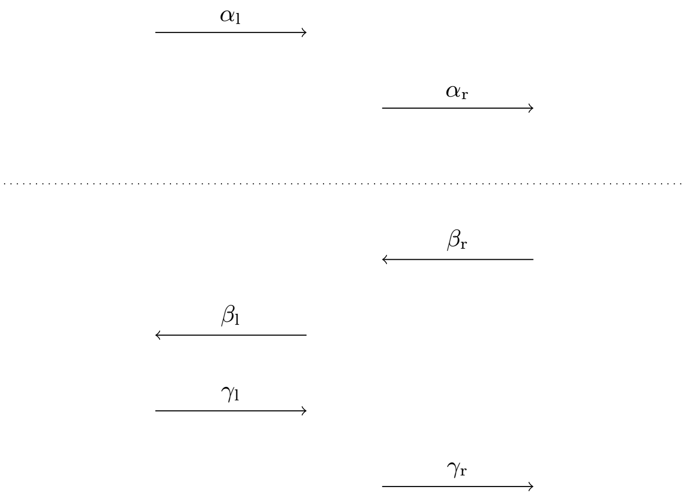

Fig. 2. Prefix (until the dotted line) that is not a safe point

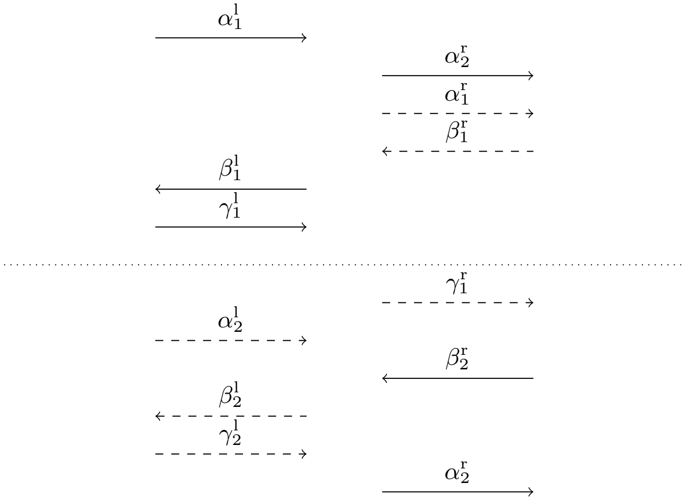

Fig. 3. A design$_0$ matches up with design$_1$

Concurrently Extractable Commitment Schemes. Concurrently extractable commitment (CECom) schemes consist of committing using the PRS preamble, and decommitting by opening all the commitments within the preamble [24]. Roughly speaking, the preamble consists of the sender committing to multiple shares of the value to be committed; then the receiver, in multiple rounds, would challenge the sender to open a subset of them in such a way that the opened shares do not reveal the committed value, but this would somehow facilitate consistency checks as shown in [24,20].

A challenge-response pair in the preamble is called a 'slot'. [20] formalized concurrent extractability and showed that the PRS preamble satisfies it if the number of slots therein is $\omega(\log(\lambda))$. We denote a CECom commitment that is SB by CECom_{sb}, the one that is SH by CECom_{sh}.

Robust Concurrent Extraction. In [24], Prabhakaran et al. demonstrated an extraction procedure by which, for an adversary Sender* that executes multiple concurrent sessions of CECom commitments, commitment information (commitment value and randomness) for each session can be extracted in polynomial time before the corresponding commitment phase is completed.

In [11], Goyal et al. extended the technique of [24] and showed how to perform efficient extractions of CECom commitments when an adversary Sender*, besides concurrently performing CECom commitments, also interacts with an 'external' party B in some arbitrary protocol Π. This setting now additionally requires that the extraction procedure rewinds the adversary Sender* in a way that B does not get rewound in the process. This is achieved in [11] by building a *robust concurrent simulator* (or just 'robust simulator') $\mathsf{RobustSim}$ that interacts with both a *robust concurrent adversary*, which commits to multiple CECom commitments, and an external party B, with which it runs some arbitrary protocol Π. For every CECom commitment that is successfully completed, Goyal et al. show that, the robust concurrent simulator – without rewinding the external party – extracts a commitment information, with all but negligible probability. [11] present this result as the *Robust Extraction Lemma* which informally states that if $\ell_{external} = \ell_{external}(\lambda)$ and $\ell_{cecom} = \ell_{cecom}(\lambda)$ denote the round complexities of Π and the CECom commitment, respectively, the Lemma guarantees the following two properties for $\mathsf{RobustSim}$:

- $\mathsf{RobustSim}$ outputs a view whose statistical distance from the adversary's view is at most $2^{-(\ell_{cecom} - \ell_{external} \cdot \log(T(\lambda)))}$, where, $T(\lambda)$ is the maximum number of total CECom commitments by the adversary.
- $\mathsf{RobustSim}$ outputs commitment information for every CECom commitment sent by the adversary with an assurance that the external party B of protocol Π is not rewound.

3 Statistical Concurrent Non-malleable Zero-Knowledge

We start by giving an intuition behind the design of our protocol. In [1], Barak et al. gave a construction of a computational CNMZK argument of knowledge.

The simulation for this protocol was restricted to be only computational due to the following reason. In their protocol, one of the messages sent by the prover is a non-malleable commitment to a valid witness. Since the non-malleable commitment is SB, and the simulator, unlike an honest prover, does not use a valid witness in this non-malleable commitment, the simulated view was only computationally indistinguishable from the real-world view of a MiM adversary. It will be quite relevant for us to note that the non-malleable commitment being SB was crucially used in the proof of concurrent non-malleability of their protocol, therefore it is not possible to replace the above commitment scheme with a statistically hiding non-malleable commitment. More specifically, the proof would begin with the real-world view and through a series of hybrids would move towards the simulated view. In some certain hybrid along the way there would be introduced PRS rewindings to facilitate simulation. Given such a hybrid that performs PRS rewindings, it would be difficult to establish that one can extract a value out of the non-malleable commitment and that the extracted value is a valid-witness. The difficulty here is in ensuring that the PRS rewindings would not interfere with the non-malleable commitment on which the NMCom extractor is run. The idea in their proof instead was to first prove for the real-world view itself that the value committed in the NMCom commitment is a valid witness, and then make transitions to hybrids by introducing PRS rewindings. The point to be noted here is that it was crucial in their proof that the non-malleable commitment is a *statistically binding* commitment, so that they could put forth arguments on the values committed in it. With this, since introducing PRS rewindings would only bias the distribution of the view output by at most a negligible amount, their proof boiled down to proving that the value committed in the NMCom commitment does not adversely change as we move across various hybrids. Now, since we began with a hybrid where the values committed were valid witnesses, the values committed in the NMCom commitments after the PRS rewindings too are valid witnesses by non-malleability (and in particular statistical binding) of the commitment scheme.

Our idea begins from noticing that statistical binding of the NMCom commitment is crucial in proving extractability of valid witnesses and not important in simulating the view of the adversary. So the core idea is to somehow ensure that when we prove the indistinguishability of the simulation, the commitment scheme is statistically hiding. Instead, when we need to argue that the distribution of the extracted message does not change, then the commitment should be statistically binding. With this being the crux of our idea, the way we shall execute it is via what we call ‘mixed non-malleable commitments’. Intuitively, a mixed non-malleable commitment scheme is associated with two efficiently samplable, computationally indistinguishable distributions, and every commitment is parameterized by some string. Furthermore, one of the distributions is such that if the string is uniformly sampled from this distribution then the commitment is SH and CB; on the other hand, a commitment that is parameterized by a string that is uniformly sampled from the other distribution is SB and CH. Given such a commitment scheme, our protocol basically is an instantiation of the BPS protocol except that the NMCom commitment in the BPS protocol is replaced by a mixed non-malleable commitment. Also, the string that parameterizes this commitment computed jointly by both the prover and the verifier is the outcome of a coin-flipping protocol. Namely, in our mixed non-malleable commitment scheme, the distribution on the parameter that produces a SH, CB commitment is the uniform distribution. Hence, the parameter generated via the coin-flipping protocol is SH and CB, as required. The BPS protocol forms the **Main BPS Phase** and the coin-flipping protocol is run in the **Coin-flipping Phase** of our protocol.

A traditional coin-flipping protocol would involve the verifier committing to a random string in the first round, followed by the prover sending another random string in the clear in the second round, the verifier opening the commitment in the third round, and finally having the prover's and the verifier's strings XOR-ed as the outcome of the coin-flipping protocol. However, now that we would also like to be able to cheat and bias the outcome to another (computationally indistinguishable) distribution (so that the mixed non-malleable commitment would then be SB), we modify the third round. Namely, instead of the third round being the verifier opening the commitment by giving both the committed value and the randomness used, the verifier would only give the committed value and then give an argument that there exists a randomness that would explain the commitment to this value. However, we won't be able to work with just any argument since we are in the concurrent setting. Furthermore, we also would like to ensure that when our simulator cheats in the argument to bias the coin-flipping outcome, the MiM adversary will not get any undue advantage. Thus, the argument that we use here is a CNMZK argument. In particular, we use the BPS argument itself. This argument forms the **BPS$^{\mathsf{CFP}}$ Phase** in our protocol.

Furthermore, towards simplifying our proof, we introduce the following slight modification of the BPS protocol in the 'Main BPS Phase'. In the original BPS protocol, the commitment in which the prover commits the valid witness to is an NMCom commitment; on the other hand, in the 'Main BPS Phase', besides sending the NMCom commitment to the witness, the prover also sends a concurrently extractable (CECom) commitment to the same witness. The simplification we achieve by adding the CECom commitment is that even the extraction of the witnesses (by the simulator-extractor) can be performed just like an extraction on any other CECom commitments in the protocol. Since, for simulation, we anyway need to employ certain techniques for the extraction from the other CECom commitments, we are now able to recycle the same techniques for witness extractions too, thus letting our focus stay on the other crucial subtleties (which we shall see as we get to the proofs of security).

We will now give a formal description of the protocol.

3.1 Our sCNMZKAoK Protocol $\langle \mathcal{P}, \mathcal{V} \rangle$

Ingredients.

1. Let CECom_{sh} and CECom_{sb} be SH and SB concurrently-extractable commitment scheme, respectively. Let each of them be of k_{cecom}-slots, where

$k_{\text{cecom}} \in \omega(\log \lambda)$. Let the sender's randomness space for these commitments be RandSpace_{cecom}.
2. Let Com_{sh} be a SH commitment scheme. Let k_{sh} be its round-complexity, where k_{sh} is a constant.
3. Let sWIAoK be a statistical WIAoK protocol. Let k_{swiaok} be its round-complexity, where k_{swiaok} is a constant.
4. Let $\mathsf{NMMXCom}_{(\cdot)}$ be our mixed non-malleable commitment scheme. Recall that it satisfies extractability and is robust w.r.t. 1-round protocols. Let k_{nmmxcom} be its round-complexity, where k_{nmmxcom} is $O(\log(\lambda))$.
5. Let $\mathsf{Com}_{\mathsf{nm}}$ be the non-malleable commitment scheme (described in Fig. 1). Recall that it satisfies extractability and is robust w.r.t. 1-round protocols. Let k_{nmcom} be its round-complexity.

In summary, the round complexities of the sub-protocols in our protocol are as follows: $k_{\text{cecom}} \in \omega(\log \lambda)$, $k_{\text{swiaok}}, k_{\text{sh}}$ are constants, and $k_{\text{nmcom}}, k_{\text{nmmxcom}} \in O(\log(\lambda))$.

Coin-Flipping Phase (CFP).

cfp_1 $(\mathcal{V} \to \mathcal{P})$: Sample $r_V \leftarrow \{0,1\}^\lambda$, $\mathsf{rand} \leftarrow \text{RandSpace}_{cecom}$ and commit to r_V using CECom_{sh} and randomness rand.

cfp_2 $(\mathcal{P} \to \mathcal{V})$: Sample $r_P \leftarrow \{0,1\}^\lambda$ and send r_P.

cfp_3 $(\mathcal{V} \to \mathcal{P})$: Send r_V.

BPS$^{\mathsf{CFP}}$ Phase.

$\mathsf{bps^{cfp}}_1$ $(\mathcal{P} \to \mathcal{V})$: Sample $\alpha \leftarrow \{0,1\}^\lambda$ and commit to α using CECom_{sb}.

$\mathsf{bps^{cfp}}_2$ $(\mathcal{V} \to \mathcal{P})$: Commit to 0^λ using Com_{sh} and argue knowledge of a commitment information (i.e., a commitment value and randomness) using sWIAoK.

$\mathsf{bps^{cfp}}_3$ $(\mathcal{P} \to \mathcal{V})$: Open the commitment of Step $\mathsf{bps^{cfp}}_1$ to α.

$\mathsf{bps^{cfp}}_4$ $(\mathcal{V} \to \mathcal{P})$: Commit to rand (used as commitment randomness in Step cfp_1) using the NMCom commitment $\mathsf{Com}_{\mathsf{nm}}$. In the rest of the paper, we shall refer to rand as the *sub-witness*.

$\mathsf{bps^{cfp}}_5$ $(\mathcal{V} \to \mathcal{P})$: Send sWIAoK to argue knowledge of either rand or r_{comsh} such that:
1. the value committed to by $\mathcal{V}$ with the NMCom commitment at Step $\mathsf{bps^{cfp}}_4$ is rand and rand explains the CECom commitment at Step cfp_1 to r_V.
2. Randomness r_{comsh} explains Com_{sh} at Step $\mathsf{bps^{cfp}}_2$ being committed to α.

Let $\mathsf{srs} = r_P \oplus r_V$.

Main BPS Phase.

bps_1 $(\mathcal{V} \to \mathcal{P})$: Sample $\sigma \leftarrow \{0,1\}^\lambda$ and commit to it using CECom_{sb}.

bps_2 $(\mathcal{P} \to \mathcal{V})$: Commit to 0^λ using Com_{sh} and argue knowledge of a commitment information (i.e., a commitment value and randomness) using sWIAoK.

bps_3 $(\mathcal{V} \rightarrow \mathcal{P})$: Open the commitment of Step bps_1 to σ.

bps_4 $(\mathcal{P} \rightarrow \mathcal{V})$: Commit to the witness w using mixed commitment $\mathsf{NMMXCom}_{\mathsf{srs}}$.

bps_{4+} $(\mathcal{P} \rightarrow \mathcal{V})$: Commit to the witness w using CECom_{sh}[2].

bps_5 $(\mathcal{P} \rightarrow \mathcal{V})$: Send sWIAoK to argue knowledge of either w, r_{nm}, r_{cecom} or r'_{comsh} such that:

1. r_{nm} and r_{cecom} explain the $\mathsf{NMMXCom}_{\mathsf{srs}}$ commitment of Step bps_4 and the CECom commitment of Step bps_{4+} to w, respectively, and w is such that $R_L(x, w) = 1$,
2. Randomness r'_{comsh} explains Com_{sh} at Step bps_2 being committed to σ.

3.2 Proofs of Security

In this section, we prove that our proposed protocol $\langle \mathcal{P}, \mathcal{V} \rangle$ is a statistical concurrent non-malleable zero-knowledge argument of knowledge. In other words, we show that there exists a simulator-extractor $\mathcal{SE}$ that, for every concurrent MiM adversary $\mathcal{M}$, outputs a view view that is statistically indistinguishable from the view $\mathsf{view}_{\mathcal{M}}(x_1, \ldots, x_{m_L}, z)$ of $\mathcal{M}$ in a real execution, and also outputs valid witnesses $\overline{y}_1, \ldots, \overline{y}_{m_R}$ for all accepting right sessions.

Our Simulator-Extractor. The Simulator-Extractor $\mathcal{SE}$ runs $\mathsf{RobustSim}$ which is the robust concurrent simulator for a robust concurrent attack. The adversary of the robust concurrent attack is a procedure I that we describe below. $\mathcal{SE}$ will then output the output of $\mathsf{RobustSim}^I(z)$. Recall that $\mathsf{RobustSim}$ runs a given adversary that mounts a robust concurrent attack by committing to multiple CECom commitments, where the adversary also interacts with an external party B in an arbitrary external protocol. $\mathsf{RobustSim}$ then is guaranteed to extract commitment information from every CECom commitment sent by the adversary before the completion of its commitment phase, in such a way that the external party B does not get rewound.

Procedure $I(z)$. I incorporates the MiM adversary $\mathcal{M}$, initiates an execution, and simulates its view as follows. Let the m_L left sessions be ordered with some arbitrary ordering. Let the m_R right sessions be ordered as follows: Consider any two right sessions, the i-th and the j-th; $i \leq j$ if and only if the CECom_{sb} commitment at Step bps_1 of the i-th session begins earlier to the CECom_{sb} commitment at Step bps_1 of the j-th session.

For every right session: Run the code of the verifier except isolate CECom_{sh} at Step bps_{4+} and relay it to external receiver. Let value y'_{t} be received from the outside ($\mathsf{RobustSim}$) at the end of the CECom_{sh} commitment.

[2] In order to make the difference from the BPS protocol more easily noticeable, the five steps here that are common to the BPS protocol are numbered in sequence from bps_1 through bps_5, while this ‘extra’ step is given a distinctive notation, bps_{4+}.

For every left session: When $\mathcal{M}$ initiates an ℓ-th new session on the left, I proceeds as follows.

- Run the coin-flipping phase and the $\mathrm{BPS}^{\mathsf{CFP}}$ phase honestly. Let srs be the outcome.
- Isolate CECom_{sb} at Step bps_1 and relay it to an external receiver. Let σ' be the value received from the outside ($\mathsf{RobustSim}$) at the end of the CECom_{sb} commitment.
- Then commit to σ' using Com_{sh} at Step bps_2; also, use the same extracted value as the witness in executing the sWIAoK of Step bps_2.
- In Step bps_3, let $\mathcal{M}$ opens its CECom_{sb} (of Step bps_1) to σ. Abort if $\sigma \neq \sigma'$.
- Commit to 0^λ using the mixed non-malleable commitment $\mathsf{NMMXCom}_{\mathsf{srs}}$ in Step bps_4.
- Commit to 0^λ using the CECom_{sh} commitment in Step bps_{4+}.
- Use σ' committed to in Step bps_2 as the witness in executing sWIAoK of Step bps_5.

When $\mathcal{M}$ halts, I outputs the view of $\mathcal{M}$ together with $y'_1, \ldots, y'_{m_R}$, and halts.

Statistical simulation. We shall prove that the view output by $\mathcal{SE}$ is distributed statistically close to the real-world view of the MiM adversary $\mathcal{M}$.

Theorem 1. *For every PPT adversary $\mathcal{M}$, $\{\mathsf{view}_{\mathcal{M}}(x_1, \ldots, x_{m_L})\}_{x_1,\ldots,x_{m_L} \in L} \approx_s \{\mathsf{view}\}_{x_1,\ldots,x_{m_L} \in L}$.*

We only provide an intuition to the proof here below. Full proof appears in the full version of the paper.

Proof Sketch. To prove the indistinguishability, we first take note of the ways in which the view generated by the simulator differs from the real-world view of the MiM adversary. Basically, the differences are that: for left sessions, the simulator does not use valid witnesses but tries to get 'fake' witnesses via the robust simulator; and for the right sessions, the simulator tries to extract witnesses via the robust simulator. While we know that using the robust simulator can incur at most negligible distance, what still remains to be shown is that the simulator using fake-witnesses for the left sessions also creates at most negligible distance from the real-view. For this, we simply rely on the statistical properties of the sub-protocols in which the simulator uses different values; namely, we rely upon SH of Com_{sh} of Step bps_2, sWI property of sWIAoK of Step bps_2, SH of the mixed non-malleable commitment of Step bps_4, and sWI of sWIAoK of Step bps_5– the steps at which the simulator uses different values in left sessions. Except for SH of the mixed non-malleable commitment of Step bps_4, all the above properties are already guaranteed by the corresponding primitives themselves; however, on the other hand, to ensure that the mixed non-malleable commitment – parameterized by srs which is the outcome of the coin-flipping protocol – is SH, we need to ensure that srs is uniformly random with all but negligible probability. Before we proceed, we thus prove that in the real-world view srs is uniform in every left session with all but negligible probability.

Claim. In the real-world view $\mathsf{view}_{\mathcal{M}}(x_1, \ldots, x_{m_L})$, for every left session, srs is uniformly random with all but negligible probability.

Proof Sketch. We begin by outlining the structure of the proof.

1. First, we show that, there exists a PPT algorithm that can extract a value r'_V from CECom_{sh} of Step cfp_1 of every left session *before* Step cfp_2 of that session is reached. Thus, since r_P is sent to the adversary after r'_V is extracted, r'_V is independent of r_P, and since r_P is uniformly random, $r_P \oplus r'_V$ is also uniformly random with all but negligible probability.
2. Then, we show that, in every left session, with all but negligible probability, $r'_V = r_V$, where, r_V is the value sent by $\mathcal{M}$ in Step cfp_3.

The above items together imply that $\mathsf{srs} = r_P \oplus r_V$ is uniformly random, with all but negligible probability.

We prove the first step above by relying upon the Robust Extraction Lemma. Basically, the PPT algorithm (mentioned in the first step above) just emulates honest provers and honest verifiers to $\mathcal{M}$ except that it relays the CECom_{sh} of Step cfp_1 of every left session to $\mathsf{RobustSim}$ for extraction. We establish the second step as follows. Recall that a commitment information for r'_V of CECom_{sh} of Step cfp_1 in question is extractable as shown for the first step. Furthermore, from the witness-extractability of the BPS protocol in $\text{BPS}^{\mathsf{CFP}}$ phase, we can extract a witness – that we call sub-witness – for r_V being committed in the same CECom_{sh} commitment. Thus, if $r_V \neq r'_V$, we break CB of CECom_{sh}.

However, the proof is still not complete. The reason is for an implicit assumption in proving the second step above that the BPS argument given by the adversary in $\text{BPS}^{\mathsf{CFP}}$ phase of the left session is sound. To prove this, we establish the following claim.

Sub-Claim 1. *In the real world view, if* $\mathsf{BPS}^{\mathsf{CFP}}$ *phase of the ℓ-th left session is accepted by the prover $\mathcal{P}_\ell$, then the value committed to by $\mathcal{M}$ in* $\mathsf{Com}_{\mathsf{nm}}$ *at Step* ${\mathsf{bps}^{\mathsf{cfp}}}_4$ *of the ℓ-th left session is a valid sub-witness.*

Proof Sketch. Intuitively, $\mathsf{Com}_{\mathsf{nm}}$ at Step ${\mathsf{bps}^{\mathsf{cfp}}}_4$ of the ℓ-th left session contains a valid sub-witness owing to

computational hiding of CECom_{sb} – to argue that $\mathcal{M}$ does not learn α, committed to by the prover in CECom_{sb}, and use it in its commitment Com_{sh} and sWIAoK at Step ${\mathsf{bps}^{\mathsf{cfp}}}_2$,

knowledge-soundness of sWIAoK in Step ${\mathsf{bps}^{\mathsf{cfp}}}_2$– to extract knowledge of commitment information (i.e., commitment value and randomness) for Com_{sh} in Step ${\mathsf{bps}^{\mathsf{cfp}}}_2$ and to verify that the extracted value will not be α,

knowledge-soundness of sWIAoK in Step ${\mathsf{bps}^{\mathsf{cfp}}}_5$– to argue that either the value committed to in $\mathsf{Com}_{\mathsf{nm}}$ at Step ${\mathsf{bps}^{\mathsf{cfp}}}_4$ is a valid sub-witness or to argue knowledge of a commitment information for Com_{sh} in Step ${\mathsf{bps}^{\mathsf{cfp}}}_2$ with commitment value as α,

and finally, computational binding of Com_{sh} at Step ${\mathsf{bps}^{\mathsf{cfp}}}_2$ to show the knowledge extracted is not α as a commitment value.

We prove each of the above steps by carefully designing interfaces that launch robust concurrent attacks and by crucially relying upon the Robust Extraction Lemma for extraction of commitment information out of these interfaces. □

With this, we continue with a hybrid argument by moving from the real-world view to the simulated view. This is facilitated by the already established facts that the messages where the simulator deviates in its behavior from the real-world are statistically hiding (in some sense). □

Witness Extractability. We shall prove that the values $y'_1, \ldots, y'_{m_R}$ extracted by the simulator-extractor $\mathcal{SE}$ are valid witnesses for the statements of the corresponding right sessions.

Theorem 2. *For every PPT adversary $\mathcal{M}$, the output of the simulator $\mathcal{SE}(x_1, \ldots, x_{m_L}, z) = (\mathsf{view}, \overline{y}_1, \ldots, \overline{y}_{m_R})$ is such that, $\forall i \in [m_R]$, $(\overline{x}_i, \overline{y}_i) \in R_L$.*

We discuss some of the core technical difficulties of the proof together with a high-level proof structure. Full proof appears in the full version of the paper

Proof Sketch. Recall that in our protocol, the prover commits to a valid witness in $\mathsf{NMMXCom}_{\mathsf{srs}}$ at Step bps_4 and also commits to the same valid witness in CECom_{sh} at Step bps_{4+} (accompanied by a sWIAoK later in Step bps_5 for correctness of behavior). Note that both of these commitments are extractable. However, we cannot in a straight-forward manner employ the proof techniques of [1] or [17] to prove that the values extracted from these commitments by the simulator are indeed valid witnesses.

We begin by pointing out the reason why we are not able to simply make use of the proofs of [1] or [17]. In both [1] and [17], the prover commits to the witness with a non-malleable commitment. Thus, the commitment is *statistically binding*. Their proofs essentially proceed in the following manner: First, prove that the values committed to in the non-malleable commitments are valid witnesses. Secondly, move to a hybrid where extractions are performed to extract 'trapdoors' for cheating in the left sessions and to extract witnesses of the right sessions. Although cheating by the simulator on the left sessions may adversely change the values committed by $\mathcal{M}$ in the commitments of the right sessions, one can argue that the values committed to in the commitments of the right sessions are still valid witnesses owing to non-malleability of the commitment schemes.

Indeed, the statistically binding NMCom commitments are the reason why the protocols of [1] and [17] are not statistical CNMZK, but only computationally so. Our approach, to recall, is to use a mixed NMCom commitment which is parameterized by a string that is output of the coin-flipping phase that precedes the main argument phase. Thus, in the real-world, as proven earlier for Theorem 1, the parameter is a uniform random string rendering the mixed NMCom commitment to be SH. (Recall that the commitment being SH was crucial in proving statistical simulation in Theorem 1). Thus, it is not clear how to solely rely on the proof techniques of [1,17] for our proof.

Our proof technique instead is as follows. We begin with the real-world experiment where the outcome of the coin-flipping protocol is a uniform random string and thus the commitment scheme at Step bps_4 is a SH commitment. Then we start moving towards the hybrid which cheats in right sessions by biasing the outcome of the coin-flipping protocol to a uniform DDH tuple. The technical challenge will be the following. Fix any right session. Let H_a and H_b be the two hybrids in our hybrid sequence such that, the commitment at Step bps_4 in H_a is SH while the same commitment is SB in H_b (due to cheating in the coin-flipping protocol). Here, we need to establish that in H_b, the committed value in the commitment at Step bps_4 is a valid witness. We establish this through a careful design of hybrids and their sequence. We expand on our techniques and the whole high-level structure of the proof here below. We shall discuss the further multiple technical difficulties in the full proof in the full version of the paper.

We begin with a hybrid that is identical to the real-world view. Then we gradually modify the behavior of the hybrid for the right sessions towards biasing the coin-flipping protocol outcome to a random DDH tuple (from a uniform random string). Here, we will also prove that the values committed to by the MiM adversary in the mixed commitment at Step bps_4 is a valid witness (note that, with the outcome of coin-flipping being a random DDH tuple, this commitment scheme is now SB, thus allowing us to put forth arguments on the values committed in it). Next, we further move to hybrids which also behave differently in the left sessions by using 'trapdoors' (or fake-witnesses) extracted from the adversary itself (instead of valid witnesses). Here, we argue that such deviation in the hybrids' behavior for the left sessions does not adversely change the values committed to in the mixed NMCom commitments of the right sessions. Finally, we thereby reach a hybrid that behaves the same as our simulator-extractor, thus proving that the values extracted by $\mathcal{SE}$ are indeed valid witnesses.

Observe that it is easy to prove indistinguishability of hybrids as we change hybrids' behavior for the left sessions. The reason is that the left sessions will still have the outcome of coin-flipping to be uniformly random and thus the corresponding mixed commitment is SH. Thus, hybrids using fake-witnesses instead of the real ones will only introduce negligible statistical distance. However, the challenging part would be to argue indistinguishability of hybrids as they deviate in their behavior on the right sessions. We expand on the difficulty and our techniques briefly here below.

In order for hybrids to start cheating in coin-flipping phases of the right sessions, it is crucial that the hybrids are ordered carefully. Note that, we cannot at once move to a hybrid which changes the outcome of the coin-flipping phase due to soundness of the BPS protocol in $\mathrm{BPS}^{\mathsf{CFP}}$ phase. Thus, we first *simulate* this BPS protocol. We do so by extracting a trapdoor from the adversary in a way similar to [1]. Then, the next hybrid would be 'free' to bias the coin-flipping outcome to a random DDH tuple. However, note that this change is not statistically indistinguishable but only computationally so. Hence, this may adversely change the values committed to in the NMCom commitments in the protocol. However, with a careful sequence of arguments, we will be able to obtain a reduction to robustness w.r.t. 1-round protocols. Here it will be crucial to ensure that the other rewindings performed by the hybrids would not rewind the external NMCom receiver of the reduction.

Let us now consider the first hybrid that biases the coin-flipping outcome of the i-th right session. By this hybrid, we will already have biased coin-flipping outcomes of the first $i-1$ sessions. We thus need to make sure that this biasing will also not adversely change the values committed to in the mixed NMCom commitments at Step bps_4 of the first $i-1$ right sessions. Here again we rely on w.r.t. 1-round protocols for these NMCom commitments too.

A major technical difficulty would be the following. Fix any right session. Consider the first hybrid that biases the coin-flipping outcome of this session. Note that the previous hybrid had coin-flipping outcome to be a random string and thus the mixed commitment at Step bps_4 of the right session here to be SH. But in the current hybrid, due to the bias, the commitment scheme is SB. Here we need to argue that the committed value is a valid witness. As shown in the full proof, this would entail proving computational binding of a CECom_{sh} commitment. Here, we are no longer able to rely only upon the Robust Extraction Lemma to ensure us of successful extractions for the following reason. In Robust Extraction Lemma, it is essential that the external protocol whose party is not supposed to be rewound is such that its round complexity is strictly less than the number of slots of the CECom commitments extracted from. However, in the current case, the external protocol itself is a CECom commitment and hence this condition can not be met. We get around this difficulty again with a careful sequencing of hybrid arguments.

Furthermore, the above technical difficulty arises at another juncture in the proof of witness extractability. Namely, we encounter a hybrid where coin-flippings of all right sessions are biased, and in the subsequent hybrid we start changing the values committed in CECom_{sh} commitments of the left sessions. Here, we are still able to rely on the robustness of the concurrent extraction as follows. Although one cannot use the Robust Extraction Lemma for a reduction to statistical hiding of the entire left CECom_{sh} commitment, we can consider intermediate hybrids where, at a time, only one sub-commitment of the CECom_{sh} commitment is changed. Thus, we are still able to use robustness of the concurrent extraction since the sub-protocol in question is only of three rounds (as per the standard CECom commitment of [24]).

Then, once we ensure that the commitments at Step bps_4 of right sessions contain valid witnesses, we proceed to argue that the values extracted from the CECom_{sh} commitments are are valid witnesses with the following argument. We, along the way, show that the adversary cannot have a trapdoor, namely, r'_{comsh} that explains Com_{sh} at Step bps_2 being committed to σ. This implies that, for every right session, the witness that is extractable from the sWIAoK argument at Step bps_5 of is an opening of the CECom_{sh} commitment (together with the opening of the $\mathsf{NMMXCom}_{\mathsf{srs}}$ commitment of Step bps_4) to a valid witness.

With this, we finally are at a hybrid that extracts valid witnesses from the right sessions. Furthermore, this hybrid is identical to our simulator-extractor, thus proving witness extractability of our protocol $\langle \mathcal{P}, \mathcal{V} \rangle$. □

Acknowledgments. C. Orlandi is in part supported by the Danish Council for Independent Research (DFF).

R. Ostrovsky and V. Rao are in part supported by NSF grants 09165174, 1065276, 1118126 and 1136174, OKAWA Foundation Research Award, IBM Faculty Research Award, Xerox Faculty Research Award, B. John Garrick Foundation Award, Teradata Research Award, and Lockheed-Martin Corporation Research Award. This material is based upon work supported by the Defense Advanced Research Projects Agency through the U.S. Office of Naval Research under Contract N00014 – 11 – 1 – 0392. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

V. Rao and A. Sahai are in part supported by a DARPA/ONR PROCEED award, NSFgrants 1228984, 1136174, 1118096, and 1065276, a Xerox Faculty Research Award, a Google Faculty Research Award, an equipment grant from Intel, and an Okawa Foundation Research Grant. This material is based upon work supported by the Defense Advanced Research Projects Agency through the U.S. Office of Naval Research under Contract N00014-11- 1-0389. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense, the National Science Foundation, or the U.S. Government.

I. Visconti is in part supported by MIUR Project PRIN “GenData 2020”.

References

1. Barak, B., Prabhakaran, M., Sahai, A.: Concurrent non-malleable zero knowledge. In: FOCS, p. 345 (2006); full version available on eprint arhive
2. Bellare, M., Micali, S., Ostrovsky, R.: The (true) complexity of statistical zero knowledge. In: STOC, pp. 494–502 (1990)
3. Catalano, D., Visconti, I.: Hybrid trapdoor commitments and their applications. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 298–310. Springer, Heidelberg (2005)
4. Catalano, D., Visconti, I.: Hybrid commitments and their applications to zero-knowledge proof systems. Theor. Comput. Sci. 374(1-3), 229–260 (2007)
5. Damgård, I., Groth, J.: Non-interactive and reusable non-malleable commitment schemes. In: Proceedings of the 35th Annual ACM Symposium on Theory of Computing, San Diego, CA, USA, June 9-11, pp. 426–437. ACM (2003)
6. Damgård, I., Nielsen, J.B.: Perfect hiding and perfect binding universally composable commitment schemes with constant expansion factor. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 581–596. Springer, Heidelberg (2002)
7. Dolev, D., Dwork, C., Naor, M.: Non-malleable cryptography (extended abstract). In: STOC, pp. 542–552 (1991)
8. Dolev, D., Dwork, C., Naor, M.: Nonmalleable cryptography. SIAM J. Comput. 30(2), 391–437 (2000); preliminary version in STOC 1991

9. Goldreich, O., Sahai, A., Vadhan, S.P.: Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In: STOC, pp. 399–408 (1998)
10. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems. In: Proc. 17th STOC, pp. 291–304 (1985)
11. Goyal, V., Lin, H., Pandey, O., Pass, R., Sahai, A.: Round-efficient concurrently composable secure computation via a robust extraction lemma. IACR Cryptology ePrint Archive 2012, 652 (2012)
12. Goyal, V., Moriarty, R., Ostrovsky, R., Sahai, A.: Concurrent statistical zero-knowledge arguments for np from one way functions. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 444–459. Springer, Heidelberg (2007)
13. Haitner, I., Nguyen, M.H., Ong, S.J., Reingold, O., Vadhan, S.P.: Statistically hiding commitments and statistical zero-knowledge arguments from any one-way function. SIAM J. Comput. 39(3), 1153–1218 (2009)
14. Halevi, S., Micali, S.: Practical and provably-secure commitment schemes from collision-free hashing. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 201–215. Springer, Heidelberg (1996)
15. Lin, H., Pass, R.: Non-malleability amplification. In: STOC, pp. 189–198 (2009)
16. Lin, H., Pass, R.: Concurrent non-malleable zero knowledge with adaptive inputs. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 274–292. Springer, Heidelberg (2011)
17. Lin, H., Pass, R., Tseng, W.-L.D., Venkitasubramaniam, M.: Concurrent non-malleable zero knowledge proofs. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 429–446. Springer, Heidelberg (2010)
18. Lin, H., Pass, R., Venkitasubramaniam, M.: Concurrent non-malleable commitments from any one-way function. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 571–588. Springer, Heidelberg (2008)
19. Mahmoody, M., Xiao, D.: Languages with efficient zero-knowledge pcps are in szk. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 297–314. Springer, Heidelberg (2013)
20. Micciancio, D., Ong, S.J., Sahai, A., Vadhan, S.: Concurrent zero knowledge without complexity assumptions. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 1–20. Springer, Heidelberg (2006)
21. Okamoto, T.: On relationships between statistical zero-knowledge proofs. J. Comput. Syst. Sci. 60(1), 47–108 (2000)
22. Ostrovsky, R., Pandey, O., Visconti, I.: Efficiency preserving transformations for concurrent non-malleable zero knowledge. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 535–552. Springer, Heidelberg (2010)
23. Ostrovsky, R., Persiano, G., Visconti, I.: Constant-round concurrent non-malleable zero knowledge in the bare public-key model. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part II. LNCS, vol. 5126, pp. 548–559. Springer, Heidelberg (2008)
24. Prabhakaran, M., Rosen, A., Sahai, A.: Concurrent zero knowledge with logarithmic round-complexity. In: Proc. 43rd FOCS (2002)
25. Sahai, A., Vadhan, S.P.: A complete problem for statistical zero knowledge. J. ACM 50(2), 196–249 (2003)