Single-to-Multi-Theorem Transformations for Non-Interactive Statistical Zero-Knowledge

Marc Fischlin Felix Rohrbach

Cryptoplexity, Technische Universität Darmstadt, Germany
www.cryptoplexity.de
marc.fischlin@cryptoplexity.de felix.rohrbach@cryptoplexity.de

Abstract. Non-interactive zero-knowledge proofs or arguments allow a prover to show validity of a statement without further interaction. For non-trivial statements such protocols require a setup assumption in form of a common random or reference string (CRS). Generally, the CRS can only be used for one statement (single-theorem zero-knowledge) such that a fresh CRS would need to be generated for each proof. Fortunately, Feige, Lapidot and Shamir (FOCS 1990) presented a transformation for any non-interactive zero-knowledge proof system that allows the CRS to be reused any polynomial number of times (multi-theorem zero-knowledge). This FLS transformation, however, is only known to work for either computational zero-knowledge or requires a structured, non-uniform common reference string.

In this paper we present FLS-like transformations that work for non-interactive statistical zero-knowledge arguments in the common *random* string model. They allow to go from single-theorem to multi-theorem zero-knowledge and also preserve soundness, for both properties in the adaptive and non-adaptive case. Our first transformation is based on the general assumption that one-way permutations exist, while our second transformation uses lattice-based assumptions. Additionally, we define different possible soundness notions for non-interactive arguments and discuss their relationships.

Keywords. Non-interactive arguments, statistical zero-knowledge, soundness, transformation, one-way permutation, lattices, dual-mode commitments

1 Introduction

In a non-interactive proof for a language $\mathcal{L}$ the prover P shows validity of some theorem $x \in \mathcal{L}$ via a proof π based on a common string crs chosen by some external setup procedure. The common requirements are completeness —that the honest prover is able to convince the verifier V for true statements x— and soundness —that the verifier will not accept false statements $x \notin \mathcal{L}$ from malicious provers. Blum et al. [BFM88] showed that such non-interactive proofs can also be zero-knowledge [GMR89], saying that a simulator can create a proof π on behalf of P if it has the ability to place some trapdoor information in crs.

1.1 Flavors of Non-Interactive Zero-Knowledge

Non-interactive zero-knowledge protocols come in many variations:

- If the prover is computationally unbounded then one speaks of a NIZK *proof system* whereas in *arguments* or *argument systems* the prover runs in polynomial time [BCC88].

- Zero-knowledge may be *computational* (NICZK) or *statistical* (NISZK) or even *perfect* (NIPZK). Note that non-interactive statistical (or perfect) zero-knowledge for $\mathcal{NP}$ requires that the prover is computationally bounded, unless the polynomial hierarchy collapses [Ps05].

- The common string crs may be uniformly distributed over all bit strings of a certain length, in which case one speaks of the *common random string* or, less frequently, of the *uniform reference string* model. In any other case the string may have more structure and one calls it a *common reference string* or, sometimes, also *public parameter* model. In this work, we will focus on the case where the crs is uniformly distributed.

Another important aspect is the question of when malicious parties choose their challenge statement x. Both zero-knowledge and soundness come in an adaptive and in a non-adaptive version. The adaptive versions say that the adversary may choose the statement x after having seen the common reference string. For zero-knowledge this means that the simulator must prepare crs independently of x and then find a valid proof π after learning a maliciously chosen $x \in \mathcal{L}$. Adaptive soundness says that the malicious prover P^* first receives crs and then tries to find a false statement $x \notin \mathcal{L}$ with a convincing proof π.

Remarkably, for soundness one usually merely distinguishes between non-adaptive and adaptive notions. But there are also different ways how to capture the fact that a malicious prover P^* needs to succeed for an invalid statement $x \notin \mathcal{L}$. Either one assumes that the prover only outputs invalid statements, thus excluding some adversaries, or one penalizes the prover and declares it to lose if it chooses some $x \in \mathcal{L}$.[1] The penalizing definition implies the exclusive one. We note that Arte and Bellare [AB20], in a concurrent work, have proposed a similar distinction between exclusive and penalizing soundness.

Both notions, exclusive and penalizing soundness, already appeared implicitly in the literature, e.g., the work by Blum et al. [BDMP91] gives both an adaptive and a non-adaptive soundness definition in the exclusive setting. Indeed, non-adaptive soundness in the literature is often cast in this style. In contrast, for adaptive soundness nowadays one often encounters the penalizing variant. It seems, however, that the adaptive/exclusive version is already sufficient for many applications, e.g., to build universally composable NIZK protocols [GOS12]. We discuss this in more detail in Section 3 when defining the different versions.

1.2 From Single-Theorem to Multi-Theorem Proofs

In this work we focus on another important property of NIZK, namely, if the crs can be used only once (*bounded* or *single-theorem*) or is applicable for many proofs (*unbounded* or *multi-theorem*). The latter is of course preferable, and indeed Feige et al. [FLS90, FLS99] show how to generally turn single-theorem NICZK proofs and arguments into multi-theorem zero-knowledge protocols. We call this the FLS-transformation.

The idea of the FLS-transformation is to augment the common random string by an extra uniformly distributed portion $\mathrm{crs}^{\mathrm{aux}}$ and let the prover for this $\mathcal{NP}$-language show that "$x \in \mathcal{L}$ or $\mathrm{crs}^{\mathrm{aux}}$ is the output of a pseudorandom generator". This allows the simulator to create this part $\mathrm{crs}^{\mathrm{aux}}$ pseudorandomly and use the generator's seed as a witness for simulating the or-proof. If the original proof is zero-knowledge, then it is also witness indistinguishable [FS90], and then one cannot distinguish or-proofs generated by the genuine prover with the witness for x from proofs created by the simulator with the witness for $\mathrm{crs}^{\mathrm{aux}}$.

Soundness, on the other hand, is not affected because a random string $\mathrm{crs}^{\mathrm{aux}}$ is not pseudorandom, except with exponentially small probability. Hence, for invalid x the "or" of the statements $x \notin \mathcal{L}$ or "$\mathrm{crs}^{\mathrm{aux}}$ is pseudorandom" would not be satisfied either with overwhelming probability. This implies that a prover would still need to break soundness of the or-protocol.

[1] We use here the terminology from [BHK15] for the comparable scenario of admissible decryption queries in chosen-ciphertext security.

The FLS-transformation, per se, is only known to work for non-interactive *computational* zero-knowledge. The reason is that the pseudorandom string $\mathsf{crs}^{\mathsf{aux}}$ of the zero-knowledge simulator is only computationally indistinguishable from a truly random string. There exists a folklore "dual version" of the FLS-transformation for non-interactive perfect (and therefore also statistical) zero-knowledge, where the crs contains a pseudorandom value by construction. But this transformation requires a structured, non-uniformly chosen crs, whereas we are interested in the setting of common *random* strings. For completeness, we provide a formal description of that folklore result along our terminology in Appendix A.

It is thus unclear if it can be used equally smoothly for statistical zero-knowledge in the common random string model. For example, Peikert and Shiehian [PS19] recently presented a statistical zero-knowledge argument for $\mathcal{NP}$ based on LWE in the common random string model, which is only zero-knowledge for a single theorem. They therefore asked whether there is an FLS-like transformation to achieve multi-theorem zero-knowledge in the statistical case.

1.3 Known NISZK Constructions

There are only a few known constructions of NISZK and NIPZK protocols for the general class $\mathcal{NP}$. Groth et al. [GOS06, GOS12] were the first to give a NIPZK argument for $\mathcal{NP}$ based on specific number-theoretic constructions over bilinear groups. Their protocol achieves multi-theorem adaptive zero-knowledge, but only non-adaptive/exclusive soundness (although this can be extended to some limited form of adaptive soundness, called adaptive culpable soundness). It is cast in the common reference string model.

Abe and Fehr [AF07] later showed how to achieve NIPZK arguments for $\mathcal{NP}$ under some form of the knowledge-of-exponent assumption. Their protocol achieves adaptive multi-theorem zero-knowledge and is adaptively sound (in the penalizing setting). This protocol is again in the common reference string model.

Sahai and Waters [SW14] show how to build NIPZK arguments for $\mathcal{NP}$ based on indistinguishability obfuscation and one-way functions. Their solution is adaptive multi-theorem zero-knowledge and non-adaptively/exclusively sound. It is designed in the common reference string model.

Peikert and Shiehian [PS19] constructed NISZK arguments for $\mathcal{NP}$ based on the LWE assumption. Their construction is based on the NIZK framework of Canetti et al. [CCRR18, CCH$^+$19] as well as Holmgren and Lombardi [HL18] which, among others, constructs a non-adaptively/exclusively sound NISZK argument for $\mathcal{NP}$ in the common random string model. Their protocol is adaptively zero-knowledge for single theorems. The instantiation of Peikert and Shiehian [PS19] uses the LWE assumption to implement the primitives and inherits the characteristics of the solutions in [CCRR18, HL18, CCH$^+$19].

Libert et al. [LPWW20] recently showed how to build *designated-verifier* statistical zero-knowledge arguments based on the (kernel) k-linear assumption, and how this construction can also be turned into a public verifiable NISZK argument. Their public verifiable construction achieves multi-theorem zero-knowledge and non-adaptive/exclusive soundness in the common reference string model.

In another construction, Libert et al. [LNPT19] achieve multi-theorem zero-knowledge in the common *random* string model. Their protocol provides non-adaptive/non-uniform soundness, i.e., where one quantifies over all inputs $x \notin \mathcal{L}$ and the crs is chosen as part of the experiment. We will later argue that in the non-adaptive case this notion is equivalent to non-adaptive/exclusive and to non-adaptive/penalizing soundness for non-uniform provers.

1.4 Our Results

In this work we show multiple FLS-SZK-transformations which preserve statistical zero-knowledge. Moreover, they allow to preserve non-adaptive or adaptive zero-knowledge and also inherit the adaptive security of soundness (in the exclusive variant). In detail, we show:

Work	Soundness	CRS uniform?	ZK	Required
FLS* [FLS90, FLS99]	adaptive/ penalizing	✓	computational	PRGs
folklore* (see Appendix A)	adaptive/ exclusive	✗	perfect	PRGs
Groth et al. [GOS06, GOS12]	non-adaptive/ exclusive	✗	perfect	bilinear groups
Abe and Fehr [AF07]	adaptive/ penalizing	✗	perfect	knowledge-of-exponent
Sahai and Waters [SW14]	non-adaptive/ exclusive	✗	perfect	iO
Libert et al. [LPWW20]	non-adaptive/ exclusive	✗	statistical	k-linear
Libert et al. [LNPT19]	non-adaptive/ non-uniform	✓	statistical	LWE
this work*	adaptive/ exclusive	✓	statistical	OWP or LWE+SIS
	adaptive/ exclusive	✓	perfect	+expected simulation

Figure 1: Comparison of different multi-theorem NIZK schemes. The entries marked with * are actually transformations for the single-to-multi-theorem cases.

- For statistical zero-knowledge we show how to transform any single-theorem non-adaptive zero-knowledge NISZK argument for $\mathcal{NP}$-languages into one which is a multi-theorem adaptively zero-knowledge NISZK argument in the common random string model. This requires only the existence of one-way permutations[2].
- For perfect zero-knowledge we show that our transformation can be augmented to preserve perfect zero-knowledge. This, however, comes at the cost of having a zero-knowledge simulator which runs in expected polynomial-time.
- Finally, we show that we can build a transformation for statistical zero-knowledge from the Learning with Errors (LWE) and Shortest Integer Solution (SIS) assumptions in the common random string model. This transformation, in contrast to the construction by Libert et al. [LNPT19], even works for *adaptively* sound NISZK arguments. This fits in nicely with the recent construction of statistical zero-knowledge arguments based on LWE [PS19].
- Additionally, we define and discuss the different soundness properties for non-interactive arguments and analyze their relationship. In particular, we show that in the non-adaptive case, the notions of exclusive, penalizing, and non-uniform soundness are all equivalent when considering non-uniform provers.

Our techniques for the constructions based on general assumptions uses a "dual" version of the original FLS-transformation. That is, instead of building the or-language for $\mathrm{crs}^{\mathrm{aux}}$ being pseudorandom, we use that $\mathrm{crs}^{\mathrm{aux}}$ is *not* pseudorandom. Since this is in general a $\mathsf{co}\mathcal{NP}$-language we need to make sure that

[2]Note that we define one-way permutations as one-way functions that are 1-1 and length-preserving, not as a family of such functions.

it is also in $\mathcal{NP}$. We achieve this by using the Blum-Micali-Yao pseudorandom generator [Yao82, BM84] based on one-way permutations and hardcore bits, which lies in $\mathcal{NP} \cap \mathsf{co}\mathcal{NP}$. Soundness for our dual FLS-transformation then follows since we can let the malicious prover run on a pseudorandom string $\mathrm{crs}^{\mathrm{aux}}$ instead, since this is indistinguishable for the efficient prover in an argument. Then the or of the two statements, $x \in \mathcal{L}$ or $\mathrm{crs}^{\mathrm{aux}}$ is not pseudorandom, is again not satisfied.

The construction based on LWE and SIS uses a primitive called dual-mode commitment scheme, i.e., a commitment which can be either perfectly-binding or statistically-hiding, based on the choice of how to generate the public key. The public keys for both modes are computationally indistinguishable. We note that the usefulness of such dual-mode commitments for non-interactive zero-knowledge is well known, starting with the work by Groth et al. [GOS06] where this technique was called parameter switching, to recent efforts like the construction of Libert et al. [LPWW20]. Most times, however, the solutions work over certain structures and yield arguments in the common reference string model.

Here, we use a construction of Gorbunov et al. [GVW15] to build these dual-mode commitments where the (statistically-hiding) public key and a commitment can be chosen as uniform bit strings. As in the FLS transformation we extend the CRS by a public key string pk and a random commitment string c and extend the language to "$x \in \mathcal{L}$ or c is a commitment to 1". For the simulator, we choose our public key to be statistically-hiding. In our construction, a statistically-hiding public key will be statistically close to a uniformly random string and indeed generate a commitment to the value 1. However, for the soundness game we exchange the public key by a perfectly-binding one and change the commitment to 0, thereby forcing the malicious prover to prove x to be in $\mathcal{L}$.

1.5 Squeezing in into Possibility and Impossibility Results

There are some known impossibility results for statistical and perfect zero-knowledge arguments. Strictly speaking, these results do not infringe with our results here, since we show how to *transform* statistical zero-knowledge arguments (from single to multiple theorems) but do not give constructions. Still, one may wonder if the combination of our transformations with the impossibility results have any implications on potential constructions.

Abe and Fehr [AF07] were the first to show that NISZK arguments cannot be proven to be adaptively sound via so-called direct black-box reductions, unless the language is in $\mathcal{P}/\mathrm{poly}$. One property which such direct reductions has is that one can use an efficient alternative to the crs generator which in addition outputs the simulator's trapdoor information (property II.(b) in [AF07]). Our construction, however, bypasses this property because for the soundness proof it generates a bad crs which does not have a trapdoor. In this sense, our technique indicates that the notion of direct black-box reductions may be too restrictive.

Pass [Pas16], using similar ideas and techniques as [AF07], shows that *adaptive* statistical and perfect zero-knowledge arguments with *adaptive* soundness cannot be based on hard primitives via black-box reductions. How does the result of Pass [Pas16] match our results? First we remark that our NIPZK is indeed *adaptively* sound and *adaptively* zero-knowledge. But the simulator only runs in polynomial time averaged over its internal randomness. Such simulators escape the results in [Pas16].

Yet, the most striking difference between the results in [AF07, Pas16] and our transformations lies in the distinct notions of adaptive soundness. We show that our transformations preserve adaptive/exclusive soundness. Opposite to that, the impossibility results of [AF07, Pas16] rely on the ability of the malicious prover to occasionally output theorems $x \in \mathcal{L}$. Put differently, they rule out the stronger form of adaptive/penalizing sound arguments, whereas we argue that adaptive/exclusive soundness is preserved. As remarked above, however, adaptive/exclusive sound arguments may still be sufficient for applications.

1.6 Concurrent Work

As mentioned earlier, Arte and Bellare [AB20] have touched upon the issue of different soundness notions in non-interactive proofs as well. Their starting point are dual-mode systems in which the common reference string can be generated in two modes, and in how far such systems allow for transference of security properties in the different modes. Our work instead focuses on the transformations for multi-theorem statistical zero-knowledge arguments.

Arte and Bellare define notions of penalizing and exclusive soundness, called SND-P and SND-E, with which our adaptive notions for soundness coincide (for efficient provers).[3] Remarkably, they show a separating example of their exclusive and penalizing soundness notion in the adaptive case, under the decisional Diffie-Hellman assumption. This example applies to our notions in the adaptive setting as well. We complement this result by showing that the notions are equivalent in the non-adaptive case, assuming non-uniform provers.

Another notably difference between the two works lies in the applications of the different soundness notions. Arte and Bellare discuss the example of the Bellare-Goldwasser signature scheme where penalizing soundness is required and exclusive soundness is insufficient. We argue along the implication of culpability that exclusive soundness may suffice in many settings.

2 Preliminaries

An $\mathcal{NP}$-relation $\mathcal{R}$ consists of pairs (x, ω) of theorems and witnesses where the length of witness is polynomially bounded in the length of the theorem, and where one can efficiently decide membership. More formally, there exists a polynomial-time Turing machine $M_{\mathcal{R}}$ and a polynomial $p_{\mathcal{R}}$ such that

$$\mathcal{R} = \{(x, \omega) \mid |\omega| \leq p_{\mathcal{R}}(|x|) \wedge M_{\mathcal{R}}(x, \omega) = 1\}.$$

The induced language $\mathcal{L}_{\mathcal{R}}$ is given by

$$\mathcal{L}_{\mathcal{R}} = \{x \in \{0,1\}^* \mid \exists \omega : (x, \omega) \in \mathcal{R}\}.$$

2.1 Non-Interactive Arguments

A non-interactive argument or proof system for an $\mathcal{NP}$-relation is now a protocol in which the setup algorithm Setup generates a common string crs which the prover P then uses to generate a proof π for the input (x, ω). The verifier V then checks this proof against crs and x only. There are some length restrictions, of course, namely that the length of the theorem x determines the length of the common string. In particular, we assume that there is a polynomial p_{Setup} such that $\mathrm{crs} \in \{0,1\}^{p_{\mathsf{Setup}}(n)}$ for any $\mathrm{crs} \xleftarrow{\$} \mathsf{Setup}(1^n)$. Let $\mathcal{R}(1^n) = \{(x, \omega) \in \mathcal{R} \mid |x| = n\}$ and $\mathcal{L}_{\mathcal{R}}(1^n) = \{x \in \mathcal{L}_{\mathcal{R}} \mid |x| = n\}$ denote the restriction of inputs of the relation and language with length $|x| = n$ such that the length of the common string for such inputs is given by $p_{\mathsf{Setup}}(n)$. Note that the verifier can easily check that $|x|$ matches the security parameter n such that we can assume that this is always the case.

We note that the string crs generated by Setup may be uniformly distributed, in which case we speak of a common random string. It may have a different distribution, in which case we call it a common reference string. In particular, we see a common random string as a special case of a common reference string.

The usual completeness notion of non-interactive arguments and proofs asks that the verifier V accepts genuine proofs π generated by the prover P for input $x \in \mathcal{L}_{\mathcal{R}}$. Soundness, on the hand, demands that the

[3] Strictly speaking, their notion of exclusiveness allows for a negligible error which could be integrated in our notion as well.

verifier does not accept false proofs generated by a malicious prover P^* for inputs $x \notin \mathcal{L}_{\mathcal{R}}$. As explained in the introduction there are various possibilities to define soundness, which we will discuss in Section 3, and just use one example of the possible definitions here.

Definition 2.1 (Non-interactive Argument) *A non-interactive argument for an $\mathcal{NP}$-relation $\mathcal{R}$ (in the common reference string model) is a triple of probabilistic polynomial-time algorithms $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ satisfying the completeness and soundness condition:*

(Perfect) Completeness: *For every $n \in \mathbb{N}$, every $(x, \omega) \in \mathcal{R}(1^n)$, every $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$, every $\pi \xleftarrow{\$} \mathsf{P}(1^n, x, \omega, crs)$ we have that $\mathsf{V}(1^n, x, \pi, crs) = 1$ with probability 1.*

(Non-Adaptive/Exclusive) Soundness: *For every (possibly malicious) probabilistic polynomial-time prover P^* outputting only $x \notin \mathcal{L}_{\mathcal{R}}$ there exists a negligible function $\epsilon(n)$ such that for every $n \in \mathbb{N}$ we have*

$$\text{Prob}\,[\,\mathsf{V}(1^n, x, \pi, crs) = 1] \leq \epsilon(|x|),$$

where the probability is over $(x, st) \xleftarrow{\$} \mathsf{P}^*(1^n)$, $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$, as well as $\pi \xleftarrow{\$} \mathsf{P}^*(1^n, st, crs)$, and V's randomness.

We say that the argument is in the common random *string model if $\mathsf{Setup}(n)$ outputs uniformly distributed strings over $\{0,1\}^{p_{\mathsf{Setup}}(n)}$ for every $n \in \mathbb{N}$.*

2.2 Zero-Knowledge

We next define zero-knowledge with the usual notion of a simulator ZKSim. In the non-interactive setting this algorithm has the advantage to choose the common string crs to simulate proofs. In the bounded case the distinguisher only gets to see a single proof for a chosen theorem, where the proof is either genuine or fabricated by the simulator. We simultaneously define the single-theorem and multi-theorem case where the distinguisher learns one or many (genuine or simulated) proofs. We first define both cases in the adaptive setting where the distinguisher selects the theorems in dependence of the common string and of previous proofs and in the non-adaptive case where the distinguisher chooses the statement(s) in advance. We stress that we are interested in statistical zero-knowledge here such that the distinguisher is unbounded, except that it can only ask for polynomially many proofs. We also allow the simulator to run in *expected* polynomial time in specially marked cases.

Definition 2.2 (Statistical and Perfect Zero Knowledge) *Let $\mathcal{R}$ be an $\mathcal{NP}$-relation and let $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ be a non-interactive argument for $\mathcal{R}$. The argument is zero-knowledge if it satisfies one of the following properties:*

Non-adaptive multi-theorem zero-knowledge: *For any unbounded algorithm D there exists a probabilistic algorithm ZKSim, the simulator, running in (expected) polynomial time, such that the advantage*

$$\mathsf{Adv}^{naSZK}_{\Pi,\mathsf{ZKSim},\mathsf{D}}(1^n) := \Pr\left[\mathsf{Expt}^{naSZK}_{\Pi,\mathsf{ZKSim},\mathsf{D}}(1^n) = 1\right] - \frac{1}{2}$$

is negligible for polynomially bounded q, where experiment $\mathsf{Expt}^{naSZK}_{\Pi,\mathsf{ZKSim},\mathsf{D}}(1^n)$ is defined in Figure 2. If the advantage of any such D is always 0 then the argument is called perfect zero-knowledge.

Adaptive multi-theorem zero knowledge: *For any unbounded algorithm D there exists a probabilistic algorithm ZKSim, the simulator, running in (expected) polynomial time, such that the advantage*

$$\mathsf{Adv}^{aSZK}_{\Pi,\mathsf{ZKSim},\mathsf{D}}(1^n) := \Pr\left[\mathsf{Expt}^{aSZK}_{\Pi,\mathsf{ZKSim},\mathsf{D}}(1^n) = 1\right] - \frac{1}{2}$$

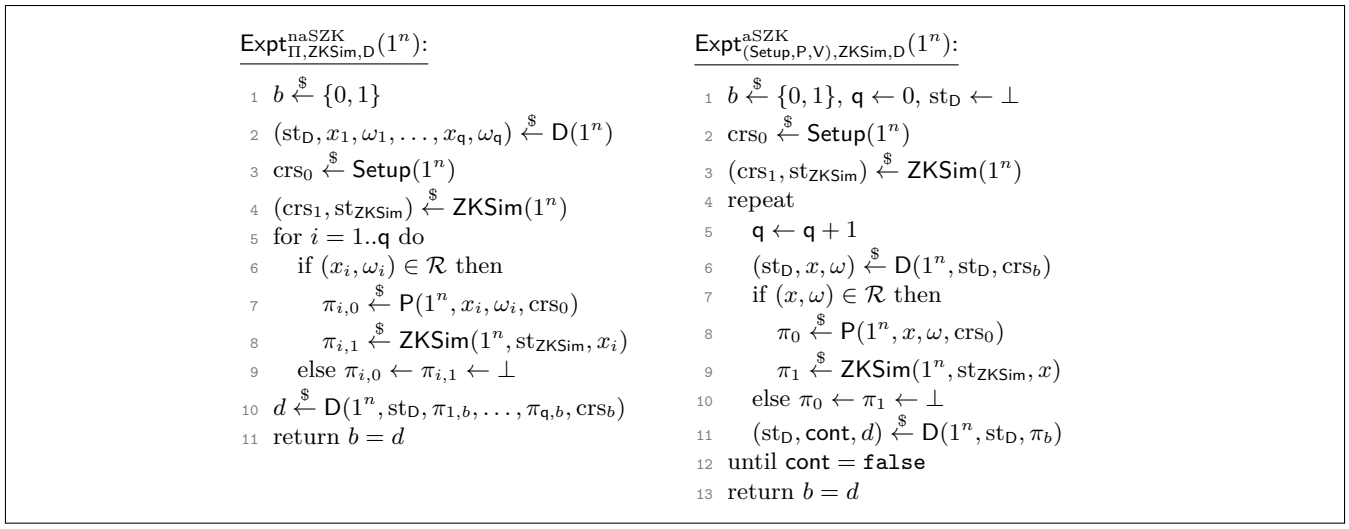

Figure 2: Non-adaptive and adaptive statistical zero-knowledge experiments.

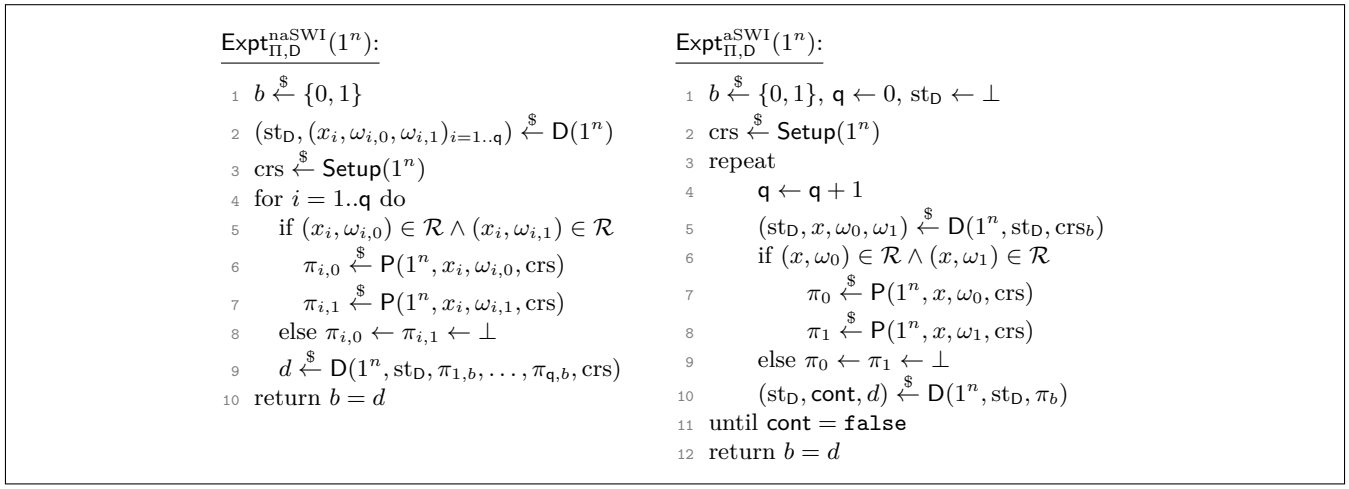

Figure 3: Non-adaptive and adaptive statistical witness indistinguishability experiments.

is negligible for polynomially bounded q*, where experiment* $\mathsf{Expt}^{aSZK}_{\Pi,\mathsf{ZKSim},\mathsf{D}}(1^n)$ *is defined in Figure 2. If the advantage of any such* D *is always* 0 *then the argument is called perfect zero-knowledge.*

The argument is single-theorem zero-knowledge of the corresponding type if the property holds for $\mathsf{q} = 1$*.*

Definition 2.3 (Statistical Witness Indistinguishability) *Let* $\mathcal{R}$ *be an* $\mathcal{NP}$*-relation. A non-interactive argument* $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ *for* $\mathcal{R}$ *is called* statistical witness indistinguishable *(NISWI) if it satisfies one of the following properties:*

Non-Adaptive multi-theorem witness indistinguishability: *For any unbounded algorithm* D *the advantage*

$$\mathsf{Adv}^{naSWI}_{\Pi,\mathsf{D}}(1^n) := \Pr\left[\mathsf{Expt}^{naSWI}_{\Pi,\mathsf{D}}(1^n) = 1\right] - \frac{1}{2}$$

is negligible for polynomially bounded q*, where the experiment* $\mathsf{Expt}^{naSWI}_{\Pi,\mathsf{D}}(1^n)$ *is defined in Figure 3. If the advantage of any such* D *is always* 0 *then the argument is called perfect witness indistinguishable.*

Adaptive multi-theorem witness indistinguishability: *For any unbounded algorithm D the advantage*

$$\mathsf{Adv}^{aSWI}_{\Pi,D}(1^n) := \Pr\left[\mathsf{Expt}^{aSWI}_{\Pi,D}(1^n) = 1\right] - \frac{1}{2}$$

is negligible for polynomially bounded q, where the experiment $\mathsf{Expt}^{aSWI}_{\Pi,D}(1^n)$ is defined in Figure 3. If the advantage of any such D is always 0 then the argument is called perfect witness indistinguishable.

The argument is single-theorem witness indistinguishable of the corresponding type if the property holds for $\mathsf{q} = 1$.

2.3 From Single-Theorem Zero-Knowledge to Multi-Theorem Witness Indistinguishability

We repeat here the well known fact that zero-knowledge implies witness indistinguishability, and that witness indistinguishability is closed under repetitions [FS90]. We state the results here for sake of completeness and according to our terminology in the statistical setting.

Lemma 2.4 *Any adaptive resp. non-adaptive single-theorem NISZK argument is also an adaptive resp. non-adaptive single-theorem NISWI argument.*

Proof (Sketch). We only argue the adaptive case; the non-adaptive case follows analogously. We can perform a game hop starting with the witness-indistinguishability experiment $\mathsf{Expt}^{\mathrm{aSWI}}_{\Pi,\mathsf{D}}(1^n)$. In this hop we replace the CRS and both proofs π_0 and π_1 in each iteration by simulated ones, all created by the simulator ZKSim without knowledge of the witnesses ω_0 and ω_1 but using the same trapdoor. Note that we can view the proofs in the WI experiment as two sequentially requested proofs in the ZK experiment, such that the SZK property ensures that this hop is statistically indistinguishable. (In the non-adaptive case we would split each entry $(x_i, \omega_{i,0}, \omega_{i,1})$ in D's initial choice into two entries $(x_i, \omega_{i,0})$ and $(x, \omega_{i,1})$.)

But now both proofs π_0 and π_1 are created without the specific witness, and since the simulator does not update its state for giving proofs, the order in which the proofs are computed is irrelevant. In this case the bit b is perfectly hidden from the distinguisher such that the advantage in predicting b is 0. □

Lemma 2.5 *Any adaptive resp. non-adaptive single-theorem NISWI argument is also an adaptive resp. non-adaptive multi-theorem NISWI argument.*

Proof (Sketch). We again only discuss the adaptive case since the non-adaptive case follows analogously. The proof follows by a hybrid argument. For this we reduce the multi-theorem distinguisher D to a bounded one D_1 which only makes one query. Let $\mathsf{Q}(n)$ be a polynomial upper bound on the number of queries q which D makes. The bounded distinguisher D_1 initially picks an index $i \xleftarrow{\$} \{1, 2, \ldots, \mathsf{Q}(n)\}$ and then internally runs in the first stage (Line 5) the distinguisher D up to the i-th query $(\mathrm{st}_{\mathsf{D}}, x, \omega_0, \omega_1)$. All requested proofs up to this step are computed internally by D_1 via P and the left witness, and returned to D. The i-th query is then computed externally, and D_1 then hands the proof back to D. In the final steps till halting, D_1 computes the remaining proofs for ω_1, and eventually returns D's decision bit d unchanged.

It can be shown that the advantage of the bounded distinguisher D_1 is at most a factor $\mathsf{Q}(n)$ larger than the one of D. Since $\mathsf{Q}(n)$ is polynomial, the difference is negligible. □

3 Soundness of Non-Interactive Arguments

Soundness of a non-interactive argument assures that a (computationally-bound) malicious prover is unable to convince the verifier of a false statement. Commonly, soundness is defined in two variants: Adaptive soundness, with allows the (possibly malicious) prover P^* to chose the statement to prove x before seeing the common random string crs, and non-adaptive soundness, in which the prover P^* has to decide on the statement x before the common random string crs is generated.

Remarkably, there is another dimension of definitional choice for soundness which often goes unnoticed in the literature. This dimension refers to the question how we measure success of the malicious prover. Clearly, the malicious prover should not make the verifier accept for a statement x not in the language. But there are two possibilities to capture the non-membership requirement. One is to disallow P^* to output $x \in \mathcal{L}$ at all. The other one is to declare P^* to lose if it picks $x \in \mathcal{L}$. Following the work of Bellare et al. [BHK15] about the question how to deal with inadmissible decryption queries in CCA-secure encryption schemes, we call the former stipulation of P^* outputting only $x \notin \mathcal{L}$ *exclusive*, because it excludes certain adversaries. The latter is called *penalizing* as it punishes P^* if it chooses $x \in \mathcal{L}$.

3.1 Soundness Definitions

In total, we define five soundness notions: adaptive vs. non-adaptive, and exclusive vs. penalizing, as well as a non-uniform variant that only exists for the non-adaptive case. We typically speak of non-adaptive/exclusive and adaptive/penalizing soundness etc. to distinguish the different types. Figure 4 provides an overview. It is also easy to see that adaptive soundness implies non-adaptive soundness in both settings, and penalizing soundness implies exclusive soundness in any of the other dimensions. The latter is easy to see because any malicious prover P^* breaking exclusive soundness must output $x \notin \mathcal{L}$ such that this prover also satisfies the winning condition in the penalizing setting. In this chapter, we highlight the further connections between these definitions and their implications.

The difference between exclusive and penalizing soundness may appear to be insignificant. Indeed, for non-interactive *proofs* it is folklore to show that the weakest one of the five notions, non-adaptive/exclusive soundness, implies the strongest one, adaptive/penalizing soundness. See for instance [Gol06]. This may explain why today's literature mostly distinguishes between the (exclusive) non-adaptive notion and the (penalizing) adaptive notion. An exception is the seminal paper by Blum et al. [BDMP91] which defines the adaptive version according to the exclusive dimension (without using our terminology here, of course). We emphasize, however, that the equivalence of all notions is not known to hold for non-interactive *arguments*.

Is a more fine-grained distinction between exclusive and penalizing soundness in arguments necessary? We argue that it is. Roughly, the difference is that in the exclusive case the malicious prover (and any other party) knows that its output is not in the language, in the penalizing case even the prover may itself be oblivious about this. This is an important ingredient in Pass' impossibility result to build adaptive sound and adaptive statistical zero-knowledge arguments based on black-box reductions [Pas16]. The result crucially relies on the malicious prover choosing a (random or pseudorandom) statement for which it does not know the status. In other words, this impossibility results rules out the strongest form of adaptive/penalizing soundness.

We next argue that the weaker form of adaptive/exclusive soundness is very relevant. It is easy to see that this notion implies a slightly weaker notion of adaptive/*culpable* soundness [GOS12]. This notion is similar to our definition of adaptive/exclusive soundness, but also requires the malicious prover to output an efficiently verifiable witness (denoted ω_{guilt} in [GOS12]) that the statement x is *not* in the language $\mathcal{L}$. Our exclusive notion asks P^* to output $x \notin \mathcal{L}$. We prove the implication that adaptive/exclusive yields adaptive/culpable soundness formally in Section 3.3.

The noteworthy fact is that adaptive/culpable soundness suffices for many applications. One of the

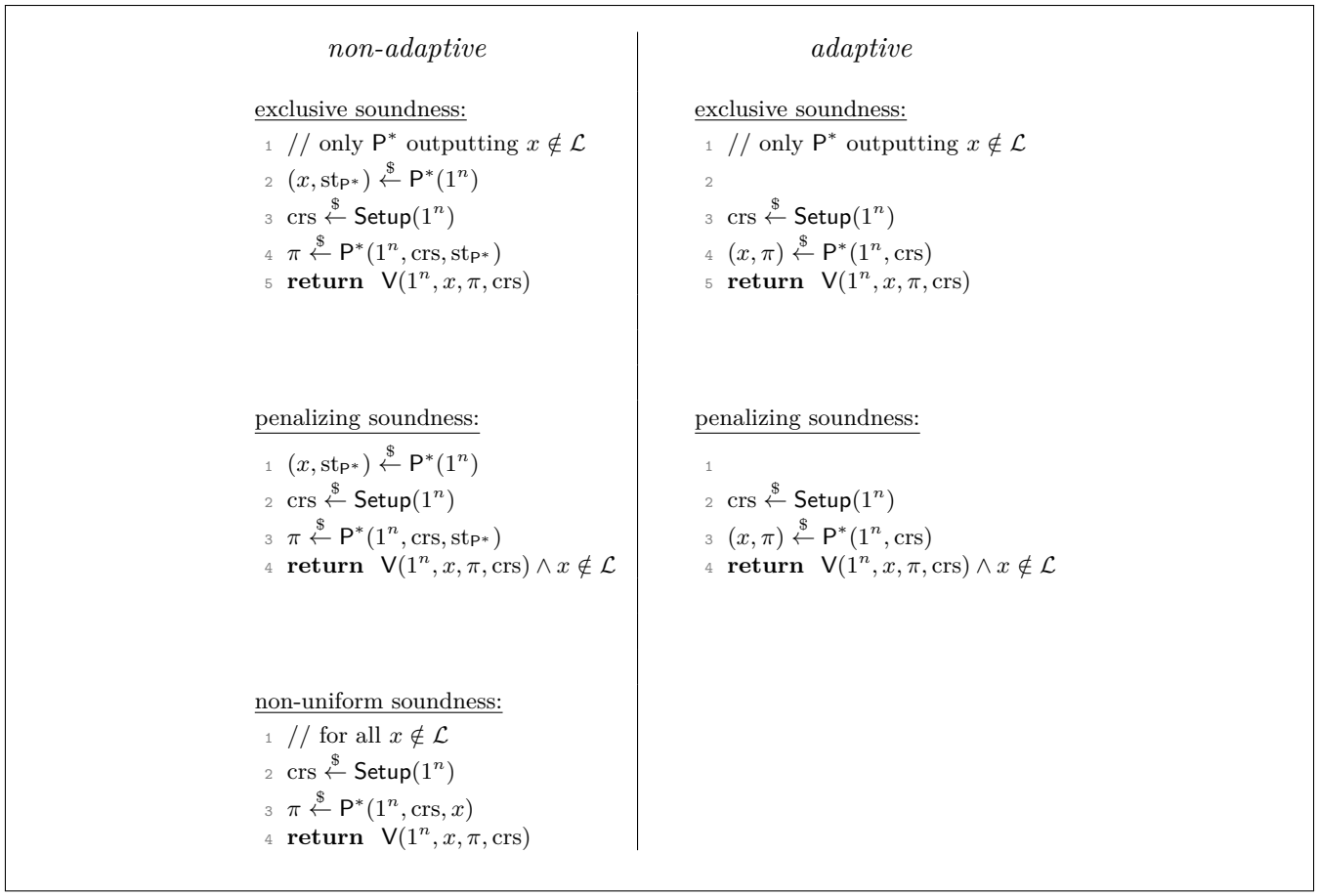

Figure 4: Different notions of soundness.

most important ones is the possibility to derive universally composable NIZK argument [GOS12]. Other applications include correctness proofs for shuffles [GL07, FL16, FLSZ17] or for e-voting [CG15]. Since adaptive/exclusive soundness implies adaptive/culpable soundness, any protocol satisfying the exclusive notion is also applicable in such settings.

We can now define non-interactive arguments with the different soundness properties:

Definition 3.1 (Soundness of non-interactive Arguments) *A non-interactive argument for an $\mathcal{NP}$-relation $\mathcal{R}$ (in the common reference string model) is a triple of probabilistic polynomial-time algorithms $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ satisfying the completeness as well as at least one of the soundness conditions:*

Non-Adaptive/Exclusive Soundness: *For every (possibly malicious) probabilistic polynomial-time prover P^* outputting only $x \notin \mathcal{L}_\mathcal{R}$ there exists a negligible function $\epsilon(n)$ such that for every $n \in \mathbb{N}$ we have*

$$\mathrm{Prob}\,[\,\mathsf{V}(1^n, x, \pi, crs) = 1] \leq \epsilon(|x|),$$

where the probability is over $(x, st) \xleftarrow{\$} \mathsf{P}^*(1^n)$, $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$, as well as $\pi \xleftarrow{\$} \mathsf{P}^*(1^n, st, crs)$, and V's randomness.

Non-Adaptive/Penalizing Soundness: *For every (possibly malicious) probabilistic polynomial-time prover P^* there exists a negligible function $\epsilon(n)$ such that for every $n \in \mathbb{N}$ we have*

$$\mathrm{Prob}\,[\,\mathsf{V}(1^n, x, \pi, crs) = 1 \wedge x \notin \mathcal{L}_\mathcal{R}] \leq \epsilon(|x|),$$

where the probability is over $(x, st) \xleftarrow{\$} \mathsf{P}^*(1^n)$*,* $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$*, as well as* $\pi \xleftarrow{\$} \mathsf{P}^*(1^n, st, crs)$*, and* V*'s randomness.*

Adaptive/Exclusive Soundness: *For every (possibly malicious) probabilistic polynomial-time prover* P^* *outputting only* $x \notin \mathcal{L}_{\mathcal{R}}$ *there exists a negligible function* $\epsilon(n)$ *such that for every* $n \in \mathbb{N}$ *we have*

$$\text{Prob}\,[\,\mathsf{V}(1^n, x, \pi, crs) = 1] \leq \epsilon(|x|),$$

where the probability is over $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$*,* $(x, \pi) \xleftarrow{\$} \mathsf{P}^*(1^n, crs)$*, and* V*'s randomness.*

Adaptive/Penalizing Soundness: *For every (possibly malicious) probabilistic polynomial-time prover* P^* *there exists a negligible function* $\epsilon(n)$ *such that for every* $n \in \mathbb{N}$ *we have*

$$\text{Prob}\,[\,\mathsf{V}(1^n, x, \pi, crs) = 1 \wedge x \notin \mathcal{L}_{\mathcal{R}}] \leq \epsilon(|x|),$$

where the probability is over $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$*,* $(x, \pi) \xleftarrow{\$} \mathsf{P}^*(1^n, crs)$*, and* V*'s randomness.*

Non-Adaptive/Non-Uniform Soundness: *For every (possibly malicious) probabilistic polynomial-time prover* P^* *there exists a negligible function* $\epsilon(n)$ *such that for every* $n \in \mathbb{N}$ *and every* $x \notin \mathcal{L}_{\mathcal{R}}$ *with* $|x| = n$*, we have*

$$\text{Prob}\,[\,\mathsf{V}(1^n, x, \pi, crs) = 1 \wedge x \notin \mathcal{L}_{\mathcal{R}}] \leq \epsilon(|x|),$$

where the probability is over $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$*, and* $\pi \xleftarrow{\$} \mathsf{P}^*(1^n, x, crs)$*, and* V*'s randomness.*

3.2 Equivalence of the Non-Adaptive Soundness Notions

We now show that the non-adaptive soundness definitions are all equivalent if we allow the malicious provers to be non-uniform:

Theorem 3.2 *For non-uniform (malicious) provers, a non-interactive argument* $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ *has non-adaptive/exclusive soundness iff it has non-adaptive/non-uniform soundness, and has non-adaptive/non-uniform soundness iff it has non-adaptive/penalizing soundness.*

Proof. Non-adaptive/exclusive soundness follows directly from non-adaptive/penalizing soundness, therefore we only need to show that non-adaptive/non-uniform soundness follows from non-adaptive/exclusive soundness and that non-adaptive/penalizing soundness follows from non-adaptive/non-uniform soundness.

We start by by showing non-adaptive/non-uniform soundness follows from non-adaptive/exclusive soundness. Let $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ be the non-interactive argument in question. Assume that there exists a successful malicious prover $\mathsf{P}^*_{na/nu}$ against the non-adaptive/non-uniform soundness, i.e., for any negligible function $\epsilon(n)$ there exists an $x \notin \mathcal{L}$ such that

$$\text{Prob}\Big[\,V(\text{crs}, x, \mathsf{P}^*_{na/nu}(\text{crs}, x))\Big] > \epsilon(|x|),$$

where the probability is over $\text{crs} \xleftarrow{\$} \mathsf{Setup}(1^n)$, as well as $P^*_{na/nu}$'s and V's randomness. We can now construct a malicious prover $\mathsf{P}^*_{na/ex}$ against non-adaptive/exclusive soundness as follows: We define the first-stage algorithm $\mathsf{P}^*_{na/ex,1}(1^n)$ to choose $x \notin \mathcal{L}$ of length n non-uniformly, such that $P^*_{na/nu}$'s success probability is maximized. The state st is left empty. Further, the second-stage algorithm $\mathsf{P}^*_{na/ex,2}$ merely calls $\mathsf{P}^*_{na/nu}$ internally, ignoring the state st. Then, the success probability of $\mathsf{P}^*_{na/ex}$ is at least as large as the one of $\mathsf{P}^*_{na/nu}$ and thus non-negligible.

Next, we show that non-adaptive/penalizing soundness follows from non-adaptive/non-uniform soundness. Assume that there exists a successful malicious prover $\mathsf{P}^*_{na/pn}$ against the non-adaptive/penalizing soundness, i.e., for any negligible function ϵ there exists an $n \in \mathbb{N}$ such that

$$\text{Prob}[\mathsf{V}(\text{crs}, x, \pi) = 1 \wedge x \notin \mathcal{L})] > \epsilon(n),$$

where the probability is over $(x, \text{st}) \xleftarrow{\$} \mathsf{P}^*_{na/pn,1}(1^n)$, $\text{crs} \xleftarrow{\$} \mathsf{Setup}(1^n)$, $\pi \xleftarrow{\$} \mathsf{P}^*_{na/pn,2}$ as well as V's internal randomness.

We can now construct a malicious prover $\mathsf{P}^*_{na/nu}$ against non-adaptive/non-uniform soundness as follows: For each input length n, we fix the pair $(\bar{x}, \bar{\text{st}})$, $\bar{x} \in \{0,1\}^n, \bar{x} \notin \mathcal{L}$, on which $\mathsf{P}^*_{na/pn,2}$'s success probability is maximized (we bound the length of $\bar{\text{st}}$ by $\mathsf{P}^*_{na/pn,1}$'s running time). Next we define $\mathsf{P}^*_{na/nu}$ as follows: On input x, $\mathsf{P}^*_{na/nu}$ checks whether x equals $\bar{x}$, and if that is the case, it internally calls $\mathsf{P}^*_{na/pn,2}(crs, \bar{x}, \bar{\text{st}})$ to generate a proof. Otherwise, $\mathsf{P}^*_{na/nu}$ returns an empty proof. Note that we use the non-uniformity to save the sequence of $(\bar{x}, \bar{\text{st}})$ for each input length. It is again easy to see that this prover is indeed a successful malicious prover against non-adaptive/non-uniform soundness. □

For adaptive soundness, Arte and Bellare [AB20] showed that there exists a protocol that provides *adaptive/exclusive* soundness but not *adaptive/penalizing* soundness. This indicates that a NISZK protocol with *adaptive/exclusive* soundness might indeed be achievable, compared to one with *adaptive/penalizing* soundness, for which Pass [Pas16] showed a black-box impossibility result.

3.3 Exclusive Soundness Implies Culpable Soundness

In this section we show that adaptive/exclusive soundness implies the notion of adaptive/culpable soundness of [GOS12]. We first recall the definition of culpable soundness (according to our terminology). For an $\mathcal{NP}$-relation $\mathcal{R}$ let $\mathcal{R}_{\text{guilt}}$ be an $\mathcal{NP}$-relation for the complement of $\mathcal{L}_{\mathcal{R}}$, i.e., $x \notin \mathcal{L}_{\mathcal{R}}$ means that there is a polynomial size ω_{guilt} such that $(x, \omega_{\text{guilt}}) \in \mathcal{R}_{\text{guilt}}$. Note that the relation $\mathcal{R}_{\text{guilt}}$ is efficiently verifiable as an $\mathcal{NP}$-relation (and $\mathcal{L}_{\mathcal{R}}$ is therefore in co-$\mathcal{NP}$).

Definition 3.3 (Adaptive/Culpable Soundness) *A non-interactive argument $(\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ for an $\mathcal{NP}$-relation $\mathcal{R}$ (in the common reference string model) has adaptive culpable soundness if for any PPT algorithm P^*_{culp} there exists a negligible function ϵ such that*

$$\text{Prob}[\,\mathsf{V}(1^n, x, \pi, crs) = 1 \wedge (x, \omega_{guilt}) \in \mathcal{R}_{guilt}] \leq \epsilon(n),$$

where the probability is over $crs \xleftarrow{\$} \mathsf{Setup}(1^n)$, $(x, \pi, \omega_{guilt}) \xleftarrow{\$} \mathsf{P}^*_{culp}(1^n, crs)$, and V's internal randomness.

Proposition 3.4 *A non-interactive argument $(\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ for an $\mathcal{NP}$-relation $\mathcal{R}$ (in the common reference string model) which has a corresponding relation $\mathcal{R}_{guilt}$ and is adaptive/exclusive sound is also adaptive/culpable sound.*

Proof. Assume that we have a successful prover $\mathsf{P}^*_{\text{culp}}$ against culpable soundness. We construct a malicious prover P^*_{ex} against exclusive soundness as follows. P^*_{ex} receives as input crs and forwards this to $\mathsf{P}^*_{\text{culp}}$ which, then, outputs $(x, \pi, \omega_{\text{guilt}})$. Our prover P^*_{ex} checks in polynomial time if $(x, \omega_{\text{guilt}}) \in \mathcal{R}_{\text{guilt}}$. If not it immediately outputs $\perp$, else it returns (x, π).

Note that since we interpret outputs $\perp$ as $\perp \notin \mathcal{L}_{\mathcal{R}}$ our prover P^*_{ex} only outputs values not in the language. It is thus an admissible attacker against exclusive soundness. Furthermore, $\mathsf{P}^*_{\text{culp}}$ can only win for $x \notin \mathcal{L}_{\mathcal{R}}$ such that only outputting (x, π) for those x cannot decrease the success probability. This yields that P^*_{ex} has the same success probability as $\mathsf{P}^*_{\text{culp}}$. □

4 Constructions based on General Assumptions

In this section we discuss our constructions based on one-wayness.

4.1 Multi-theorem NISZK based on One-way Permutations

Our approach uses the same idea as in [FLS90] of having crs^{aux}, but we apply it in a dual way. That is, we use a language saying that crs^{aux} is *not* pseudorandom. Since this is in general a $\mathsf{co}\mathcal{NP}$-relation we use the Blum-Micali-Yao [Yao82, BM84] generator for one-way permutations,

$$G(s) = f^{|s|}(s) \| \operatorname{hb}(s) \| \operatorname{hb}(f(s)) \| \ldots \| \operatorname{hb}(f^{|s|-1}(x)),$$

where s is the seed of length $|s| = n$, f is a one-way permutation, $f^i(s)$ the i-fold iteration of f for input s, and hb is a hardcore bit for f. Proving that a string crs^{aux} is *not* in the range of G is easy if one presents the unique seed s such that the first bits are equal to $f^{|s|}(s)$ and that the remaining bits are not the hardcore bits.

For our simulator we can thus generate a perfectly distributed common random string by picking s randomly, computing $G(s)$, and randomly flipping the hardcore bits:

$$\text{crs}^{\text{aux}} \leftarrow G(s) \oplus 0^{|s|} \| t$$

where each bit $t_i \xleftarrow{\$} \{0,1\}$ in $t = t_1 \| \ldots \| t_{|s|}$ is chosen uniformly and independently. Unless all t_i's are 0 — which happens with probability $2^{-|s|}$— this gives the simulator a witness for crs^{aux} not being pseudorandom in form of s, t. If $t = 0^{|s|}$ the we let the simulator abort. This unlikely event of all t_i's being 0 causes our simulator to be statistical zero-knowledge instead of being perfect zero-knowledge.

For the malicious prover in the soundness game we will hand over a pseudorandom string $G(s)$ instead of a truly random one. For the bounded prover this is computationally indistinguishable. But then the prover does not have a witness for the or-part and would thus need to break soundness of the other protocol part for $x \notin \mathcal{L}_{\mathcal{R}}$. This step preserves any exclusive soundness notion but not penalizing soundness, because we need to be able to detect diverging success behavior of the prover in the two cases (which we may not necessarily be able to in the penalizing setting since we cannot check if x is in the language or not).

Below we formally define the augmented language $\mathcal{L}^{\text{or}}_{\mathcal{R}}$ as

$$\mathcal{L}^{\text{or}}_{\mathcal{R}} = \left\{ (x, y) \;\middle|\; \exists \omega : (x, \omega) \in \mathcal{R} \vee \exists s, t \in \{0,1\}^{\lfloor |y|/2 \rfloor} : y = G(s) \oplus 0^{|s|} \| t, t \neq 0^{|s|} \right\}$$

and the corresponding relation $\mathcal{R}^{\text{or}}$ accordingly. Note that this is an $\mathcal{NP}$-relation such that, if we have any single-theorem statistical NIZK for general $\mathcal{NP}$-relations, then we also have an multi-theorem statistical witness-indistinguishable argument for this relation $\mathcal{R}^{\text{or}}$.

For pseudorandomness of G we consider for any probabilistic polynomial-time algorithm $\mathcal{D}$ the probability that $\mathcal{D}(1^n, y_{b'}) = b'$ where the probability is taken over $b' \xleftarrow{\$} \{0,1\}$, $y_0 \leftarrow G(s)$ for $s \xleftarrow{\$} \{0,1\}^n$, $y_1 \xleftarrow{\$} \{0,1\}^{2n}$. Let $\mathsf{Adv}^{\text{PRG}}_{G,\mathsf{D}}(1^n) := \Pr\left[\mathcal{D}(1^n, y_{b'}) = b'\right] - \frac{1}{2}$ be $\mathcal{D}$'s advantage. We say that G is a pseudorandom generator if for any probabilistic polynomial-time algorithm $\mathcal{D}$ this advantage is negligible. Note that the Blum-Micali-Yao generator based on a one-way permutation f achieves this property.

Construction 4.1 (SZK-FLS-Transformation) *Let $\mathcal{R}$ be an $\mathcal{NP}$-relation. Let f be a one-way permutation and $\Pi^{or} = (\mathsf{Setup}^{or}, \mathsf{P}^{or}, \mathsf{V}^{or})$ be a multi-theorem non-interactive statistical witness-indistinguishable argument for the $\mathcal{NP}$-relation $\mathcal{R}^{or}$. We construct a multi-theorem non-interactive statistical zero knowledge argument $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ for $\mathcal{R}$ as follows (see also Figure 5):*

$\mathsf{Setup}(1^n)$	$\mathsf{P}(1^n,x,\omega,\mathrm{crs})$	$\mathsf{V}(1^n,x,\pi,\mathrm{crs})$
$\mathrm{crs}^{\mathrm{or}} \xleftarrow{\$} \mathsf{Setup}^{\mathrm{or}}(1^n)$	// $\mathrm{crs} = \mathrm{crs}^{\mathrm{or}} \Vert \mathrm{crs}^{\mathrm{aux}}$	// $\mathrm{crs} = \mathrm{crs}^{\mathrm{or}} \Vert \mathrm{crs}^{\mathrm{aux}}$
$\mathrm{crs}^{\mathrm{aux}} \xleftarrow{\$} \{0,1\}^{2n}$	$\pi^{\mathrm{or}} \xleftarrow{\$} \mathsf{P}^{\mathrm{or}}((x,\mathrm{crs}^{\mathrm{aux}}),\omega,\mathrm{crs}^{\mathrm{or}})$	$d \xleftarrow{\$} \mathsf{V}^{\mathrm{or}}((x,\mathrm{crs}^{\mathrm{aux}}),\pi,\mathrm{crs}^{\mathrm{or}})$
$\mathrm{crs} \leftarrow \mathrm{crs}^{\mathrm{or}} \Vert \mathrm{crs}^{\mathrm{aux}}$	$\pi \leftarrow \pi^{\mathrm{or}}$	**return** d
return crs	**return** π	

Figure 5: SZK-FLS-Transformation for multi-theorem NISZK argument (additional input 1^n omitted for P^{or} and V^{or} for space reasons).

CRS: *We define the sampling algorithm* $\mathsf{Setup}(1^n)$ *for the common random string crs for our construction as*

$$\mathsf{Setup}(1^n) = \mathsf{Setup}^{or}(1^n) \Vert U_{2n},$$

where U_{2n} *is the uniform distribution on all* $2n$*-bit strings.*

Prover: *The prover* P*, receiving* 1^n*,* $crs = crs^{or}||crs^{aux}$*,* x *and* ω *(for* $\mathcal{R}$*) as input, uses* (x, crs^{aux}) *and* ω *for the augmented relation* $\mathcal{R}^{or}$ *and computes a witness-indistinguishable proof* π^{or} *for this* $\mathcal{NP}$*-relation using the string* crs^{or}*.*

Verifier: *The verifier* V *receives* 1^n*,* $crs = crs^{or} \Vert crs^{aux}$*,* x*, and a proof* π^{or} *for* $\mathcal{R}^{or}$*. The verifier accepts iff* $\mathsf{V}^{or}(1^n, (x, crs^{aux}), \pi^{or}, crs^{or})$ *accepts.*

Theorem 4.2 *Let* $\mathcal{R}$ *be an* $\mathcal{NP}$*-relation. Assuming that* $\Pi^{or} = (\mathsf{Setup}^{or}, \mathsf{P}^{or}, \mathsf{V}^{or})$ *is a non-interactive statistical single-theorem zero-knowledge argument for* $\mathcal{R}^{or}$ *and that* f *is a one-way permutation, the non-interactive argument system* $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ *in Construction 4.1 is a multi-theorem statistical zero-knowledge argument. Furthermore, if the underlying protocol* Π^{or} *is (non-adaptively resp. adaptively) exclusively sound, then so is the derived protocol* Π*; if* Π^{or} *is adaptive resp. non-adaptive zero-knowledge, then so is* Π*.*

Proof. (Perfect) Completeness: Note that the verifier V accepts a genuine proof $\pi^{\mathrm{or}} \xleftarrow{\$} \mathsf{P}(1^n, x, \omega, \mathrm{crs})$ for original data $\mathrm{crs} = \mathrm{crs}^{\mathrm{or}} \Vert \mathrm{crs}^{\mathrm{aux}} \xleftarrow{\$} \mathsf{Setup}(1^n)$ and $x \in \mathcal{L}_{\mathcal{R}}$ if and only if V^{or} accepts π^{or} for $(x, \mathrm{crs}^{\mathrm{aux}})$ under $\mathrm{crs}^{\mathrm{or}}$. The latter is always true since $x \in \mathcal{L}_{\mathcal{R}}$ such that the pair $(x, \mathrm{crs}^{\mathrm{aux}})$ of the or-relation is also in $\mathcal{L}^{\mathrm{or}}_{\mathcal{R}}$, the output of P is given by the output of P^{or} for valid input, and the verifier V^{or} accepts genuine proofs of P^{or}.

Non-adaptive/Exclusive Soundness: Assume that Π^{or} is non-adaptively/exclusively sound. Our argument to show that Π, too, has this property is as follows. We will first substitute the "real" common random string by one in which the augmented component $\mathrm{crs}^{\mathrm{aux}}$ is always in the range of the pseudorandom generator G. This will be indistinguishable for the bounded prover P^* such that P^* outputs a valid proof with roughly equal probability for pseudorandom G. In this step we exploit the property of non-adaptive/exclusive soundness that $x \notin \mathcal{L}_{\mathcal{R}}$ is chosen before crs. But then the or-language does not have a witness for either part, such that the malicious prover would have to break (non-adaptive) exclusive soundness of the protocol for $\mathcal{R}^{\mathrm{or}}$.

More formally, let crs be a CRS generated as described above and crs_G an artificial CRS generated as

$$\mathrm{crs}_G \leftarrow \mathsf{Setup}^{\mathrm{or}}(1^n)||G(s),$$

where s is chosen uniformly from $\{0,1\}^n$. In a first game hop we argue that a successful malicious prover P^* for such a CRS is almost as successful as for a genuine one, that is,

$$\text{Prob}\,[\,\mathsf{V}(1^n, x, \pi, \text{crs}) = 1] \approx \text{Prob}\,[\,\mathsf{V}(1^n, x, \pi, \text{crs}_G) = 1]$$

are negligibly close, where the probability is over $(x, \text{st}) \xleftarrow{\$} \mathsf{P}^*(1^n)$, $\text{crs} \xleftarrow{\$} \mathsf{Setup}(1^n)$ and $\pi \xleftarrow{\$} \mathsf{P}^*(1^n, \text{st}, \text{crs})$ and V's randomness in the first case, and accordingly over $(x, \text{st}) \xleftarrow{\$} \mathsf{P}^*(1^n)$, $\text{crs}_G \xleftarrow{\$} \mathsf{Setup}^{\text{or}}(1^n)||G(s)$, $\pi \xleftarrow{\$} \mathsf{P}^*(1^n, \text{st}, \text{crs}_G)$ and V's randomness in the second case.

We show the indistinguishability by defining a distinguisher $\mathcal{D}$ against the pseudorandom generator G. For security parameter n the distinguisher receives a string $y \in \{0,1\}^{2n}$ as input, either picked uniformly at random, or being the output of the pseudorandom generator. The distinguisher then invokes the prover and verifier to decide:

$\underline{\mathcal{D}(1^n, y)}$
$(x, \text{st}) \xleftarrow{\$} \mathsf{P}^*(1^n)$
$\text{crs}^{\text{or}} \xleftarrow{\$} \mathsf{Setup}^{\text{or}}(1^n)$
$\text{crs} \leftarrow \text{crs}^{\text{or}} \| y$
$\pi \xleftarrow{\$} \mathsf{P}^*(1^n, \text{st}, crs)$
return $\mathsf{V}(1^n, x, \pi, \text{crs})$

We claim that the distinguishing advantage bounds the difference between the two games, where G_0 is the original soundness game (with output 1 indicating that P^* has won) and G_1 describes the game where we use the artificial string crs_G instead. Since the two games correspond syntactically to the cases that the distinguisher receives a random y resp. a pseudorandom y we get:

$$\text{Prob}\,[\,\mathsf{G}_0(1^n) = 1] - \text{Prob}\,[\,\mathsf{G}_1(1^n)] \leq 2 \cdot \mathsf{Adv}^{\text{PRG}}_{G,\mathsf{D}}(1^n).$$

Next we turn the malicious prover P^* in G_1 against non-adaptive/exclusive soundness against the unbounded scheme Π into one of the same type for the augmented scheme Π^{or}. Note that we are guaranteed that P^* always outputs $x \notin \mathcal{L}_{\mathcal{R}}$ by assumption. Our prover P^*_{or} against Π^{or} works as follows:

$\underline{\mathsf{P}^*_{\text{or}}(1^n)}$
$(x, \text{st}) \xleftarrow{\$} \mathsf{P}^*(1^n)$
$s \xleftarrow{\$} \{0,1\}^n$
$\text{crs}^{\text{aux}} \leftarrow G(s)$
$\text{st}_{\text{or}} \leftarrow (\text{st}, \text{crs}^{\text{aux}})$
return $((x, \text{crs}^{\text{aux}}), \text{st}_{\text{or}})$

$\underline{\mathsf{P}^*_{\text{or}}(1^n, \text{st}_{\text{or}}, \text{crs}^{\text{or}})}$
$/\!/\ \text{st}_{\text{or}} = (\text{st}, \text{crs}^{\text{aux}})$
$\text{crs} \leftarrow \text{crs}^{\text{or}} || \text{crs}^{\text{aux}}$
$\pi \xleftarrow{\$} \mathsf{P}^*(1^n, \text{st}, \text{crs})$
return π

We first observe that, if P^* always outputs $x \notin \mathcal{L}_{\mathcal{R}}$, then our prover P^*_{or} always outputs $(x, \text{crs}^{\text{aux}}) \notin \mathcal{L}^{\text{or}}_{\mathcal{R}}$. This holds as the string crs^{aux} is pseudorandom such that neither condition of the or-language is satisfied. In addition, P^*_{or} is efficient. Hence, P^*_{or} is also an admissible attacker against non-adaptive/exclusive soundness, this time against $\mathcal{L}^{\text{or}}_{\mathcal{R}}$.

We conclude that, by the soundness of Π^{or}, the success probability of prover P^*_{or} must be negligible. But because P^*_{or} has the same success probability as P^* in G_1 it follows that the winning probability of P^*

in G_1 must also be negligible. Since this success probability is negligibly close to the one of P^* in G_0 by the pseudorandomness of G, we derive that P^* success probability against our derived protocol Π must be negligible.

Adaptive/Exclusive Soundness: The proof in the adaptive case follows exactly as in the non-adaptive case. Only this time P^* chooses $x \notin \mathcal{L}_\mathcal{R}$ after seeing crs. But both the distinguisher $\mathcal{D}$ against the pseudorandomness $\mathcal{D}$, as well as the prover P^*_{or} against soundness, can assemble the common random string before P^* selects x. It follows as before that the probability of P^*_{or} against adaptive/exclusive soundness of Π^{or} and thus the one of P^* against Π must be negligible.

Zero Knowledge: The simulator ZKSim works as follows: On input 1^n it first generates $\text{crs} = \text{crs}^{\text{or}}||\text{crs}^{\text{aux}}$, where $\text{crs}^{\text{or}} \xleftarrow{\$} \mathsf{Setup}^{\text{or}}(1^n)$ and crs^{aux} is sampled as

$$\text{crs}^{\text{aux}} \leftarrow G(s) \oplus 0^{|s|}\|t$$

for s, t chosen uniformly from $\{0,1\}^n$. Note that since f is a permutation this CRS has the same distribution as a truly random string. If $t = 0^{|s|}$ then the simulator immediately aborts. Else it outputs crs as the common random string and (s,t) as state $\text{st}_{\mathsf{ZKSim}}$. When receiving a (valid) theorem $x \in \mathcal{L}_\mathcal{R}$ the simulator runs the prover P^{or} for $\mathcal{R}^{\text{or}}$ on input $1^n, (x, \text{crs}^{\text{aux}}), \text{crs}^{\text{or}}$ and witness (s,t) to generate a proof π^{or}. The state remains unchanged.

By assumption, Π^{or} is single-theorem statistical zero knowledge (either adaptively or non-adaptively secure). Further, by Lemma 2.4 it is single-theorem statistical witness indistinguishable, and by Lemma 2.5 also multi-theorem statistical witness indistinguishable for the same level of adaptiveness. Therefore, whenever ZKSim is able to find a valid $t \neq 0^{|s|}$, the statistical distance between genuine proofs by P^{or} (for witness ω) and proofs by ZKSim resp. P^{or} (with witness (s,t)) is given by a negligible term $\epsilon(n)$ for any distinguisher requesting at most q proofs. As ZKSim fails to derive $t \neq 0^{|s|}$ with probability 2^{-n}, the overall statistical distance is therefore at most $\epsilon(n) + 2^{-n}$ and thus negligible. Thus, $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ is multi-theorem statistical zero knowledge. We note that the protocol inherits the notion of zero-knowledge adaptiveness from Π^{or}. □

We remark that the transformation also preserves adaptive/culpable soundness. For this notion the distinguisher against the pseudorandom generator in the soundness part can check efficiently if the prover's choice x is in the language or not with the help of the witness ω_{guilt} which the prover needs to output, too.

4.2 Adaptive Perfect Zero-Knowledge under Expected Polynomial Time

The construction in the previous section displays a small error in the simulation, even if we would start with a perfect zero-knowledge or witness-indistinguishable argument. The reason is that our simulator may not generate a valid pair (s,t) with $t \neq 0^{|s|}$. However, to preserve perfect zero-knowledge the simulator cannot simply discard such bad pairs, else outputs of the form $G(s)$ would not be hit by the simulator (while a uniformly chosen string may actually be in the range of G).

The solution in the single-theorem case is to use the fact that the event of picking bad t's is very unlikely, namely, 2^{-n}. We will now decrease the probability further such that we can safely search for the actual witness ω for the x part in this rare case, without violating polynomial run time on the average. For this let $p_\mathcal{R}$ denote the polynomial which bounds the witness length of relation $\mathcal{R}$. Then we use a pseudorandom generator $G(s)$ as before, but we iterate the one-way permutation f for $p_\mathcal{R}(n)$ steps. Now the probability of picking some input $(s,t) \in \{0,1\}^n \times \{0,1\}^{p_\mathcal{R}(n)}$ with $t = 0^{p_\mathcal{R}(n)}$ is $2^{-p_\mathcal{R}(n)}$. Given that this happens we let the simulator (later, after having obtained the input x) search through all potential witnesses $w \in \{0,1\}^{\leq p_\mathcal{R}(n)}$ and each time check in polynomial time $q_\mathcal{R}(n)$ if $(x,w) \in \mathcal{R}$. The run time of the simulator for the exhaustive search is then bounded from above by $2 \cdot 2^{p_\mathcal{R}(n)} \cdot q_\mathcal{R}(n)$. But since

this step is only executed with probability at most $2^{-p_{\mathcal{R}}(n)}$ the overall run time of the simulator remains polynomial in expectation.

If we assume that the original argument system Π^{or} is perfectly witness indistinguishable for non-adaptively chosen statements, then the derived protocol is perfectly zero-knowledge, with as simulator running in expected polynomial time and holding either a witness s, t for the auxiliary part or a witness for x to compute the proof. As in the statistical case, the protocol still preserves non-adaptive/exclusive or adaptive/exclusive soundness.

The next step is to extend the above idea to multiple theorems. If we have polynomial many statements $x_1, \ldots, x_{\mathsf{q}}$ then we would have to search for all witnesses to simulate the proofs if $t = 0 \ldots 0$. But the time to search for all these witnesses by brute force is additive and requires at most $2\mathsf{q} \cdot q_{\mathcal{R}}(n) \cdot 2^{p_{\mathcal{R}}(n)}$ many steps. Hence, the expected run time is still polynomial.

We finally remark that our simulator only attains the simple notion of expected polynomial where we average the number of steps over the randomness of the algorithm. It is not known if one can modify the simulator to achieve more robust notions, such as Levin's average-time complexity.

5 A Lattice-Based Construction

The main drawbacks of the previous constructions based on general assumptions is that they are not directly applicable to lattice-based problems because they require a one-way permutation.In this section we therefore present a multi-theorem extension in the common random string using dual-mode commitments, based on the Learning-With-Errors (LWE) and the Shortest-Integer-Solution (SIS) assumptions. Here, a dual-mode commitment scheme is a commitment scheme that can be either statistically hiding or perfectly binding, depending on how the public key is generated. Further, without the knowledge of a secret key, it is computationally indistinguishable whether a given public key belongs to the statistically hiding or to the perfectly binding version.

5.1 Dual-Mode Commitment Schemes based on Lattices

A (non-interactive) commitment scheme consists of a probabilistic polynomial-time algorithm to generate a public key and another probabilistic polynomial-time algorithm which allows to commit to a message under a public key. The scheme can be statistically-hiding (and computationally-binding), or it can be perfectly-binding (and computationally-hiding). A dual-mode scheme has now two key generation algorithms, one for the statistically-hiding and one for the perfectly-binding case. Furthermore, the output of the two key generation algorithms is computationally indistinguishable:

Definition 5.1 (Dual-mode Commitment Scheme) *A non-interactive commitment scheme*

$$\Gamma = (\mathsf{Gen}_H, \mathsf{Gen}_B, \mathsf{Com})$$

is called a dual-mode commitment scheme *if,*

Statistically-Hiding Mode: *The scheme* $(\mathsf{Gen}_H, \mathsf{Com})$ *is a statistically-hiding, computationally-binding commitment scheme.*

Perfectly-Binding Mode: *The scheme* $(\mathsf{Gen}_B, \mathsf{Com})$ *is a perfectly-binding, computationally-hiding commitment scheme.*

Indistinguishability of Modes: *The random variables* Gen_H *and* Gen_B *are computationally indistinguishable.*

For the dual-mode commitments, we will use two homomorphic trapdoor functions defined by Gorbunov et al. [GVW15]. As pointed out in [CH19], these two trapdoor functions give rise to a dual-mode commitment scheme. It has been shown in [CH19] that it can be used together with a non-interactive witness-indistinguishable proof system for bounded distance decoding to build non-interactive *designated-verifier* computational zero-knowledge arguments. We will describe this dual-mode commitment scheme now in detail and provide proof sketches based on the security proofs in [GVW15].

The construction of the commitment scheme in [GVW15] itself is based on the SIS problem [Ajt96], stating that for parameters $n, m = \text{poly}(n), q$ and β_{SIS} it is hard to find a short non-zero integer vector u (of length at most β_{SIS}) to a given random $n \times m$-matrix A over $\mathbb{Z}_q$ such that $Au = 0$. The noteworthy property is that there is also a method to generate an $n \times m$ matrix A over $\mathbb{Z}_q$ together with a trapdoor in a secure way. This is implemented by an algorithm $\mathsf{TrapGen}$, taking $1^n, 1^m$ and q as input. Furthermore, there exists an algorithm $\mathsf{Sam}(1^m, 1^m, q)$ which outputs a "small" matrix $U \in \mathbb{Z}_q^{m \times m}$. As discussed in [GVW15] it holds that A generated by $\mathsf{TrapGen}(1^n, 1^m, q)$ is statistically close to uniform, and that A and $A \cdot U$ (sampled according to Sam) are statistically close to A and a uniform matrix V'.

The final ingredient is a fixed and easy to compute matrix $G \in \mathbb{Z}_q^{n \times m}$ for the given parameter which allows us to build the commitment scheme. We can then commit to a value $x \in \mathbb{Z}_q$ for matrix A by computing $A \cdot U + x \cdot G$. Note that since $A \cdot U$ is statistically close to a uniform matrix V' we obtain that x is statistically hidden. We describe the scheme more formally in the following construction:

Construction 5.2 (Hiding-Mode Commitment Scheme)

Key Generation Gen_H: *We generate $(A, td) \leftarrow \mathsf{TrapGen}(1^n, 1^m, q)$, where $A \in \mathbb{Z}_q^{n \times m}$ and $\mathsf{TrapGen}$ is defined in [GVW15, Lemma 2.2]. We then set $pk \leftarrow A$ and discard the trapdoor td.*

Commitment Com: *For input pk and $x \in \mathbb{Z}_q$, we sample $U \leftarrow \mathsf{Sam}(1^m, 1^m, q)$ and return $pk \cdot U + x \cdot G$. To open the commitment, we reveal x and U (or the randomness used to sample U).*

Proposition 5.3 *Assuming the $SIS(n, m, q, \beta_{SIS})$-assumption holds, Construction 5.2 is a statistically-hiding and computationally-binding commitment scheme.*

Proof. We will first show that Construction 5.2 is *statistically hiding*. As shown in [GVW15], we have that the following two tuples are statistically close:

$$(\text{pk}, x, \text{pk} \cdot U + x \cdot G) \equiv_s (\text{pk}, x, V')$$

where $U \leftarrow \mathsf{Sam}(1^m, 1^m, q)$ and $V' \leftarrow \mathbb{Z}_q^{n \times m}$, i.e., the commitment is statistically indistinguishable from a random matrix.

The *computationally-binding* property of the construction follows directly from the claw-freeness of the trapdoor function, which is proven in [GVW15]. For any probabilistic polynomial-time adversary $\mathcal{A}$ we have

$$\text{Prob}_{(U,U',x,x') \xleftarrow{\$} \mathcal{A}(1^n, \text{pk})} \left[\mathsf{Com}(\text{pk}, x; U) = \mathsf{Com}(\text{pk}, x'; U') \right] \leq \text{negl}(n)$$

for $x \neq x'$, where the probability is over $\text{pk} \xleftarrow{\$} \mathsf{Gen}_H(1^n)$ and $\mathcal{A}$'s randomness.

Next we recall from [GVW15] how we can switch to a perfectly-binding mode by assuming the hardness of LWE. This problem states that given a matrix A and $As + e$ for a small error vector e sampled from a distribution χ, recovering s is hard [Reg05].

Construction 5.4 (Binding-Mode Commitment Scheme)

Key Generation Gen_B: *We sample* $A' \leftarrow \mathbb{Z}_q^{(n-1)\times m}$ *uniformly and* $s' \xleftarrow{\$} \mathbb{Z}_q^{n-1}$ *and set*

$$pk \leftarrow \begin{pmatrix} A' \\ s'A' + e \end{pmatrix},$$

where e *is a short "noise vector" sampled from* χ*.*

Commitment Com: *The commitment is identical to the one in Construction 5.2.*

Proposition 5.5 *Assuming the LWE*(q, χ)*-assumption and the SIS*(n, m, q, β_{SIS})*-assumption holds, Construction 5.4 is a perfectly-binding and computationally-hiding commitment scheme.*

Proof. To show this construction is *perfectly binding*, it suffices to show that we can uniquely recover x using s. Indeed, if we know s', we can set $s = (-s', 1)$ and $z = (0, \ldots, 0, r)$ and calculate

$$s\,(\mathrm{pk} \cdot U + x \cdot G)\,G^{-1}(z) = e \cdot U \cdot G^{-1}(z) + x \cdot \langle s, z \rangle = x \cdot r + e'.$$

Note that G^{-1} is a polynomial-time algorithm whose existence is guaranteed by Lemma 2.2 in [GVW15]. For correctly chosen parameters r and e, this lets us recover x uniquely. Now, as s does not depend on x or U, if for two pairs (x, U) and (x', U')

$$pk \cdot U + x \cdot G = pk \cdot U' + x' \cdot G,$$

holds, then we have $x = x'$.

For *computationally hiding* we argue with the indistinguishability of the public keys. As the hiding-mode commitment scheme is statistically hiding and the public keys are computationally indistinguishable, this construction must be computationally hiding. □

Proposition 5.6 *Assuming the LWE*(q, χ)*- and SIS*(n, m, q, β_{SIS})*-assumptions hold, Constructions 5.2 and 5.4 together form a dual-mode commitment scheme.*

Proof. We have already shown that Construction 5.2 is a statistically-hiding, computationally-binding commitment scheme and that Construction 5.4 is a perfectly-binding, computationally-hiding commitment scheme. Therefore, all that is left to prove is that the public keys of both schemes are computationally indistinguishable.

First, note that all but the last column of matrix A are generated uniformly random (or statistically close to that) for both public keys. Therefore, the problem is equivalent to distinguish between $A's + e$ and v' given A', where $v' \in \mathbb{Z}_q^n$ is a uniformly random vector and s and e are sampled as described in the scheme. However, this is exactly the decisional LWE problem. By our assumption, the two public keys are therefore indistinguishable. □

5.2 SZK-FLS-Transformation based on Lattices

We will now define our multi-theorem transformation based on the dual-mode commitment scheme in the previous section. As before, we will use the FLS-type transform, therefore we only need to define a sampling algorithm for the auxiliary CRS $\mathrm{crs}^{\mathrm{aux}}$ and an augmented or-relation $\mathcal{R}^{\mathrm{or}}$ for this string.

The sampling algorithm $\mathsf{Setup}^{\mathrm{aux}}$ to generate $\mathrm{crs}^{\mathrm{aux}}$ will just generate uniformly random values representing a public key pk and a commitment c:

$$\mathrm{crs}^{\mathrm{aux}} = (\mathrm{pk}, c) \leftarrow U_{nmq} \times U_{nmq}.$$

Note that a random public key corresponds to the hiding-mode public key (with overwhelming probability), as Gen_H produces a public key that is statistically close to uniform random [GVW15].

Technically the public key and the commitment in $\mathrm{crs}^{\mathrm{aux}}$ are matrices over $\mathbb{Z}_q$, and not uniform strings as required by the common random string model. However, we can generate random elements in $\mathbb{Z}_q$ from uniform strings by interpreting a random string of length $|q| + n$ as an integer and mapping it to the residue mod q. The statistically distance to a uniform element from $\mathbb{Z}_q$ is then exponentially small. We stress that we can also go "backwards" with this technique. Given a random value $v \in \mathbb{Z}_q$ we can add a random multiple $i \cdot q$ to v for $i \xleftarrow{\$} \{0, 1, \ldots, 2^{n-1}\}$ to get an (almost) uniform $|q|+n$ bit string which would map to v again. Hence, from now on we switch between random matrices from $\mathbb{Z}_q$ and uniformly random string whenever convenient.

Our relation will now ask for a given public key pk of the commitment scheme and commitment c, both found in the common random string, if there is a matrix $U \leftarrow \mathsf{Sam}(1^m, 1^m, q)$ resp. randomness u such that $U = \mathsf{Sam}(1^m, 1^m, q; u)$, such that the commitment opens to 1:

$$((\mathrm{pk}, c), u) \in \mathcal{R}^{\mathrm{or}} :\Longleftrightarrow U = \mathsf{Sam}(1^m, 1^m, q; u) \wedge c = \mathsf{Com}(\mathrm{pk}, 1; U).$$

Given these two properties we can now use the same construction as for the one-way permutation, only that we use the relation above and the sampler $\mathsf{Setup}^{\mathrm{aux}}$ to generate $\mathrm{crs}^{\mathrm{aux}}$. In fact the construction is otherwise identical to the one in Figure 5:

Construction 5.7 (SZK-FLS-Dual-Mode-Transformation) *Let $\mathcal{R}$ be an $\mathcal{NP}$-relation. Further, let $\Gamma = (\mathsf{Gen}_H, \mathsf{Gen}_B, \mathsf{Com})$ be a non-interactive dual-mode commitment scheme and suppose that $\Pi^{or} = (\mathsf{Setup}^{or}, \mathsf{P}^{or}, \mathsf{V}^{or})$ be a multi-theorem non-interactive statistical witness-indistinguishable argument for the $\mathcal{NP}$-relation $\mathcal{R}^{or}$. We construct a multi-theorem non-interactive statistical zero knowledge argument $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ for $\mathcal{R}$ as in Figure 5 with the following exception:*

CRS: *We define the sampling algorithm $\mathsf{Setup}(1^n)$ for the common random string crs for our construction as*

$$\mathsf{Setup}(1^n) = \mathsf{Setup}^{or}(1^n) \| \mathsf{Setup}^{aux}(1^n).$$

The prover algorithm P and verifier algorithm V are as before.

Theorem 5.8 *Let $\mathcal{R}$ be an $\mathcal{NP}$-relation. Assuming that $\Pi^{or} = (\mathsf{Setup}^{or}, \mathsf{P}^{or}, \mathsf{V}^{or})$ is a non-interactive statistical single-theorem zero-knowledge argument for $\mathcal{R}^{or}$ and that $\Gamma = (\mathsf{Gen}_H, \mathsf{Gen}_B, \mathsf{Com})$ is a dual-mode non-interactive commitment scheme, the non-interactive argument $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ in Construction 5.7 is a multi-theorem statistical zero-knowledge argument. Furthermore, if the underlying protocol Π^{or} is (non-adaptively resp. adaptively) exclusively sound, then so is the derived protocol Π; if Π^{or} is adaptive resp. non-adaptive zero-knowledge, then so is Π.*

Proof. The proof is very close to the one of Theorem 4.2 such that we only sketch the main differences here.

(Perfect) Completeness: It follows as in the one-way permutation case that the honest verifier accepts proofs generated by P for $x \in \mathcal{L}_{\mathcal{R}}$.

Exclusive Soundness: To show exclusive soundness (in the non-adaptive or adaptive case) we first switch the auxiliary string to a randomly sampled binding key $\mathrm{pk} \xleftarrow{\$} \mathsf{Gen}_B(1^n)$ and a 0-commitment $\mathsf{Com}(\mathrm{pk}, 0; U)$, instead of using uniformly random values. Note that we can use two game hops to show that this is computationally indistinguishable from genuine common random strings, in the first hop we replace the random key component in $\mathrm{crs}^{\mathrm{aux}}$ by a key $\mathrm{pk} \xleftarrow{\$} \mathsf{Gen}_H(1^n)$, which is even statistically close. Then we

replace the random commitment component in $\mathrm{crs}^{\mathrm{aux}}$ by a random commitment to 0, $\mathsf{Com}(\mathrm{pk}, 0; U)$. This is again statistically indistinguishable.

And finally we switch to a binding key $\mathrm{pk} \xleftarrow{\$} \mathsf{Gen}_B(1^n)$ and a 0-commitment under this key. This is computationally indistinguishable by the indistinguishability of the dual-mode key generation. (The additional 0-commitment can be computed easily given a hiding or binding key.) This is where we again use exclusive soundness to turn a malicious prover into a distinguisher against the dual-mode scheme, analogously to the distinguisher against the pseudorandomness of the generator in the one-way permutation case.

We now have an auxiliary string which contains a binding key and a 0-commitment, such that the or-part in the $\mathcal{R}^{\mathrm{or}}$ cannot be satisfied. It follows now as before that soundness of the constructed protocol follows from the soundness of the original non-interactive argument.

Zero-Knowledge: For adaptive multi-theorem zero-knowledge we remark that the simulator ZKSim can create the key part in the auxiliary string as a hiding key $\mathrm{pk} \xleftarrow{\$} \mathsf{Gen}_H(1^n)$ and the commitment part as a 1-commitment under pk. Since the key pk and the 1-commitment are statistically close to a uniform strings, the simulator's string $\mathrm{crs}^{\mathrm{aux}}$ is statistically close to a uniform string. For this string $\mathrm{crs}^{\mathrm{aux}}$ the simulator can use the randomness of the commitment as a witness. The remaining steps in the proof are identical to the ones in the proof of Theorem 4.2. □

6 Conclusion

We have shown how to apply the idea of the FLS transformation also for statistical zero-knowledge arguments. Our solution follows the FLS approach of using an or-language to give the simulator the leverage to compute a witness, showing that finding the right auxiliary languages in the statistical case is possible. Let us highlight two important aspects of our transformations.

First, our transformations based on one-way permutations and on lattices work in the common *random* string model and does not require any structure of the CRS. Common *reference* strings have the inherent disadvantage that they have some structure and that one needs to trust the party which generates the string. A prominent example is the discussion about the trustworthiness of the Zcash reference string and follow-up suggestions to use common random strings instead, e.g., [FMMO19]. Of course, a party generating a common *random* string may also impose some trust assumption, as our lattice-based solution shows. But several measures to thwart attacks can be implemented much easier than for structured strings. This includes the computation of the string as the output of a hash function, or by xoring common random strings from several sources.

The other aspect we would like to emphasize that our transformations preserve adaptive security for both zero-knowledge and soundness. While this does not conflict with black-box impossibility result for such statistical zero-knowledge arguments directly [AF07, Pas16], because it may still be that there are no single-theorem statistical zero-knowledge arguments with both adaptive properties, and that our transformation is void for such cases. Yet, in the course of showing adaptive soundness we have, in passing, encountered a possibility to bypass the impossibility results. A key observation is that one may be able to achieve adaptive soundness and zero-knowledge if one switches to the notion of exclusive soundness. This adaptive/exclusive soundness implies adaptive/culpable soundness and thus suffices for many practical applications.

Acknowledgments

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – SFB 1119 – 236615297.

References

[AB20] Vivek Arte and Mihir Bellare. Dual-mode NIZKs: Possibility and impossibility results for property transfer. Cryptology ePrint Archive, Report 2020/629, 2020. `https://eprint.iacr.org/2020/629`. (Cited on pages 2, 6, and 13.)

[AF07] Masayuki Abe and Serge Fehr. Perfect NIZK with adaptive soundness. In Salil P. Vadhan, editor, *TCC 2007: 4th Theory of Cryptography Conference*, volume 4392 of *Lecture Notes in Computer Science*, pages 118–136, Amsterdam, The Netherlands, February 21–24, 2007. Springer, Heidelberg, Germany. (Cited on pages 3, 4, 5, and 22.)

[Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *28th Annual ACM Symposium on Theory of Computing*, pages 99–108, Philadephia, PA, USA, May 22–24, 1996. ACM Press. (Cited on page 19.)

[BCC88] Gilles Brassard, David Chaum, and Claude Crépeau. Minimum disclosure proofs of knowledge. *J. Comput. Syst. Sci.*, 37(2):156–189, 1988. (Cited on page 1.)

[BDMP91] Manuel Blum, Alfredo De Santis, Silvio Micali, and Giuseppe Persiano. Noninteractive zero-knowledge. *SIAM Journal on Computing*, 20(6):1084–1118, 1991. (Cited on pages 2 and 10.)

[BFM88] Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications (extended abstract). In *20th Annual ACM Symposium on Theory of Computing*, pages 103–112, Chicago, IL, USA, May 2–4, 1988. ACM Press. (Cited on page 1.)

[BHK15] Mihir Bellare, Dennis Hofheinz, and Eike Kiltz. Subtleties in the definition of IND-CCA: When and how should challenge decryption be disallowed? *Journal of Cryptology*, 28(1):29–48, January 2015. (Cited on pages 2 and 10.)

[BM84] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. Comput.*, 13(4):850–864, 1984. (Cited on pages 5 and 14.)

[CCH+19] Ran Canetti, Yilei Chen, Justin Holmgren, Alex Lombardi, Guy N. Rothblum, Ron D. Rothblum, and Daniel Wichs. Fiat-Shamir: from practice to theory. In Moses Charikar and Edith Cohen, editors, *51st Annual ACM Symposium on Theory of Computing*, pages 1082–1090, Phoenix, AZ, USA, June 23–26, 2019. ACM Press. (Cited on page 3.)

[CCRR18] Ran Canetti, Yilei Chen, Leonid Reyzin, and Ron D. Rothblum. Fiat-Shamir and correlation intractability from strong KDM-secure encryption. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018, Part I*, volume 10820 of *Lecture Notes in Computer Science*, pages 91–122, Tel Aviv, Israel, April 29 – May 3, 2018. Springer, Heidelberg, Germany. (Cited on page 3.)

[CG15] Pyrros Chaidos and Jens Groth. Making sigma-protocols non-interactive without random oracles. In Jonathan Katz, editor, *PKC 2015: 18th International Conference on Theory and Practice of Public Key Cryptography*, volume 9020 of *Lecture Notes in Computer Science*,

pages 650–670, Gaithersburg, MD, USA, March 30 – April 1, 2015. Springer, Heidelberg, Germany. (Cited on page 11.)

[CH19] Geoffroy Couteau and Dennis Hofheinz. Designated-verifier pseudorandom generators, and their applications. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019, Part II*, volume 11477 of *Lecture Notes in Computer Science*, pages 562–592, Darmstadt, Germany, May 19–23, 2019. Springer, Heidelberg, Germany. (Cited on page 19.)

[FL16] Prastudy Fauzi and Helger Lipmaa. Efficient culpably sound NIZK shuffle argument without random oracles. In Kazue Sako, editor, *Topics in Cryptology – CT-RSA 2016*, volume 9610 of *Lecture Notes in Computer Science*, pages 200–216, San Francisco, CA, USA, February 29 – March 4, 2016. Springer, Heidelberg, Germany. (Cited on page 11.)

[FLS90] Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *31st Annual Symposium on Foundations of Computer Science*, pages 308–317, St. Louis, MO, USA, October 22–24, 1990. IEEE Computer Society Press. (Cited on pages 2, 4, 14, and 26.)

[FLS99] Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple noninteractive zero knowledge proofs under general assumptions. *SIAM J. Comput.*, 29(1):1–28, 1999. (Cited on pages 2, 4, and 26.)

[FLSZ17] Prastudy Fauzi, Helger Lipmaa, Janno Siim, and Michal Zajac. An efficient pairing-based shuffle argument. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017, Part II*, volume 10625 of *Lecture Notes in Computer Science*, pages 97–127, Hong Kong, China, December 3–7, 2017. Springer, Heidelberg, Germany. (Cited on page 11.)

[FMMO19] Prastudy Fauzi, Sarah Meiklejohn, Rebekah Mercer, and Claudio Orlandi. Quisquis: A new design for anonymous cryptocurrencies. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology – ASIACRYPT 2019, Part I*, volume 11921 of *Lecture Notes in Computer Science*, pages 649–678, Kobe, Japan, December 8–12, 2019. Springer, Heidelberg, Germany. (Cited on page 22.)

[FS90] Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *22nd Annual ACM Symposium on Theory of Computing*, pages 416–426, Baltimore, MD, USA, May 14–16, 1990. ACM Press. (Cited on pages 2 and 9.)

[GL07] Jens Groth and Steve Lu. A non-interactive shuffle with pairing based verifiability. In Kaoru Kurosawa, editor, *Advances in Cryptology – ASIACRYPT 2007*, volume 4833 of *Lecture Notes in Computer Science*, pages 51–67, Kuching, Malaysia, December 2–6, 2007. Springer, Heidelberg, Germany. (Cited on page 11.)

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM J. Comput.*, 18(1):186–208, 1989. (Cited on page 1.)

[Gol06] Oded Goldreich. *Foundations of Cryptography: Volume 1.* Cambridge University Press, USA, 2006. (Cited on page 10.)

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect non-interactive zero knowledge for NP. In Serge Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 339–358, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Heidelberg, Germany. (Cited on pages 3, 4, and 5.)

[GOS12] Jens Groth, Rafail Ostrovsky, and Amit Sahai. New techniques for noninteractive zero-knowledge. *J. ACM*, 59(3):11:1–11:35, 2012. (Cited on pages 2, 3, 4, 10, 11, and 13.)

[GVW15] Sergey Gorbunov, Vinod Vaikuntanathan, and Daniel Wichs. Leveled fully homomorphic signatures from standard lattices. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *47th Annual ACM Symposium on Theory of Computing*, pages 469–477, Portland, OR, USA, June 14–17, 2015. ACM Press. (Cited on pages 5, 19, 20, and 21.)

[HL18] Justin Holmgren and Alex Lombardi. Cryptographic hashing from strong one-way functions (or: One-way product functions and their applications). In Mikkel Thorup, editor, *59th Annual Symposium on Foundations of Computer Science*, pages 850–858, Paris, France, October 7–9, 2018. IEEE Computer Society Press. (Cited on page 3.)

[LNPT19] Benoît Libert, Khoa Nguyen, Alain Passelègue, and Radu Titiu. Simulation-sound arguments for LWE and applications to KDM-CCA2 security. Cryptology ePrint Archive, Report 2019/908, 2019. `https://eprint.iacr.org/2019/908`. (Cited on pages 3 and 4.)

[LPWW20] Benoît Libert, Alain Passelègue, Hoeteck Wee, and David J. Wu. New constructions of statistical NIZKs: Dual-mode DV-NIZKs and more. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III*, volume 12107 of *Lecture Notes in Computer Science*, pages 410–441. Springer, 2020. (Cited on pages 3, 4, and 5.)

[Pas16] Rafael Pass. Unprovable security of perfect NIZK and non-interactive non-malleable commitments. *Comput. Complex.*, 25(3):607–666, 2016. (Cited on pages 5, 10, 13, and 22.)

[Ps05] Rafael Pass and Abhi shelat. Unconditional characterizations of non-interactive zero-knowledge. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 118–134, Santa Barbara, CA, USA, August 14–18, 2005. Springer, Heidelberg, Germany. (Cited on page 2.)

[PS19] Chris Peikert and Sina Shiehian. Noninteractive zero knowledge for NP from (plain) learning with errors. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019, Part I*, volume 11692 of *Lecture Notes in Computer Science*, pages 89–114, Santa Barbara, CA, USA, August 18–22, 2019. Springer, Heidelberg, Germany. (Cited on pages 3 and 4.)

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th Annual ACM Symposium on Theory of Computing*, pages 84–93, Baltimore, MA, USA, May 22–24, 2005. ACM Press. (Cited on page 19.)

[SW14] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In David B. Shmoys, editor, *46th Annual ACM Symposium on Theory of Computing*, pages 475–484, New York, NY, USA, May 31 – June 3, 2014. ACM Press. (Cited on pages 3 and 4.)

[Yao82] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science*, pages 80–91, Chicago, Illinois, November 3–5, 1982. IEEE Computer Society Press. (Cited on pages 5 and 14.)

A Multi-theorem NIPZK in the Common Reference String Model

As mentioned in the introduction, there exists a folklore transformation from single-theorem to multi-theorem non-interactive perfect zero-knowledge based on the FLS construction [FLS90, FLS99], which however requires a non-uniform common reference string. We will provide here a formal description according to our terminology for sake of completeness.

The original FLS transformation is not statistical (nor perfect) zero-knowledge, as the simulator always chooses crs to be a image of a pseudo-random generator, which can only have up to 2^n images in $\{0,1\}^{3n}$. It is therefore not statistically close to uniformly random values. The idea of the folklore transformation is to always use an image of the pseudo-random generator as crs. Then, the crs generated by the simulator is identically distributed to the real crs. Obviously, now even for an honestly-generated crs the second condition of the augmented language ("Is $\mathrm{crs}^{\mathrm{aux}}$ in the image of the PRG?") is always true. However, as the malicious prover is computationally bounded we can safely replace the string by a truly random string which is most likely not in the range of the generator. We will now define the setup algorithm for the scheme as well as the augmented language.

Let $G : \{0,1\}^n \to \{0,1\}^{3n}$ be a pseudo-random generator. The sampling algorithm $\mathsf{Setup}^{\mathrm{aux}}$ to generate $\mathrm{crs}^{\mathrm{aux}}$ calls the PRG G on a uniformly random seed:

$$\mathrm{crs}^{\mathrm{aux}} \leftarrow G(U_n).$$

Note that crs is indeed not statistically close to a uniformly random string of length $3n$. Each string of length $3n$ only has a negligible probability of having a pre-image under G. It is, however, computationally indistinguishable from a uniformly random string due to the security properties of the pseudo-random generator. The augmented language is now defined identically to the one in the original FLS construction:

$$y \in \mathcal{R}^{\mathrm{or}} :\Longleftrightarrow \exists x \in \{0,1\}^n : G(x) = y.$$

Construction A.1 (Folklore-SKZ-FLS) *Let $\mathcal{R}$ be an $\mathcal{NP}$-relation. Further, let G be a pseudo-random generator stretching n-bit inputs to $3n$-bit outputs for each n, and suppose that $\Pi^{or} = (\mathsf{Setup}^{or}, \mathsf{P}^{or}, \mathsf{V}^{or})$ is a multi-theorem non-interactive perfect witness-indistinguishable argument for the $\mathcal{NP}$-relation $\mathcal{R}^{or}$. We construct a multi-theorem non-interactive perfect zero knowledge argument $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ in the common* reference *string model for $\mathcal{R}$ as in Figure 5 with the following exception:*

CRS: *We define the sampling algorithm $\mathsf{Setup}(1^n)$ for the common* reference *string crs for our construction as*

$$\mathsf{Setup}(1^n) = \mathsf{Setup}^{or}(1^n) \| \mathsf{Setup}^{aux}(1^n).$$

The prover algorithm P and verifier algorithm V are as before.

Theorem A.2 *Let $\mathcal{R}$ be an $\mathcal{NP}$-relation. Assuming that $\Pi^{or} = (\mathsf{Setup}^{or}, \mathsf{P}^{or}, \mathsf{V}^{or})$ is a non-interactive perfect single-theorem zero-knowledge argument for $\mathcal{R}^{or}$ and that $G : \{0,1\}^n \to \{0,1\}^{3n}$ is a pseudo-random generator, the non-interactive argument $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ in Construction A.1 is a multi-theorem perfect zero-knowledge argument. Furthermore, if the underlying protocol Π^{or} is (non-adaptively resp. adaptively) exclusively sound, then so is the derived protocol Π; if Π^{or} is adaptive resp. non-adaptive zero-knowledge, then so is Π.*

Proof. Again, the proof is very close to the one of Theorem 4.2 and of course to the original proof in [FLS90, FLS99] such that we only sketch the main differences here.

(Perfect) Completeness: It follows as in the one-way permutation case that the honest verifier accepts proofs generated by P for $x \in \mathcal{L}_{\mathcal{R}}$.

Exclusive Soundness: In the original FLS construction, the proof for soundness argues that the probability for crs to be an image of the PRG G is negligible and therefore the malicious prover cannot find a proof for it. In our case, this does not work, as we guarantee the (honestly-generated) crs to be in the domain of G. We can, however, first use a game hop to replace crs with a uniformly random string. As the malicious prover is computationally bounded, the security of the pseudo-random generator guarantees that both games are indistinguishable for the prover.

The rest of the proof is then identical to the proof of the original FLS construction. Here, if we ask for non-adaptive soundness, i.e., x is chosen before the (now completely random) crs, then the underlying protocol only needs to be non-adaptive sound as well. If we demand adaptive soundness and x is chosen after the crs, then we also require adaptive soundness of the underlying protocol.

Zero-Knowledge: The proof for (adaptive/non-adaptive) zero-knowledge is identical to the original proof of the FLS construction, with the only difference being that the crs chosen by the simulator is now by construction identically distributed to the honestly-generated one. □

Note that non-adaptive/penalizing soundness for the protocol follows from our equivalence result in Section 3.2 for non-uniform provers.