IACR Communications in Cryptology
ISSN 3006-5496, Vol. 1, No. 4, 48 pages.

https://doi.org/10.62056/a66ce0iuc

On Quantum Simulation-Soundness

Behzad Abdolmaleki[1], Céline Chevalier[2,3], Ehsan Ebrahimi[4], Giulio Malavolta[5,6] and Quoc-Huy Vu[7]

[1] University of Sheffield, Sheffield, UK
[2] CRED, Université Panthéon-Assas, Paris II, Paris, France
[3] DIENS, École normale supérieure, PSL University, CNRS, INRIA, Paris, France
[4] University of Luxembourg, Luxembourg
[5] Bocconi University, Milan, Italy
[6] Max-Planck Institute in Security and Privacy, Bochum, Germany
[7] Léonard de Vinci Pôle Universitaire, Research Center, Paris La Défense, France

Abstract. Non-interactive zero-knowledge (NIZK) proof systems are a cornerstone of modern cryptography, but their security has received little attention in the quantum settings. Motivated by improving our understanding of this fundamental primitive against quantum adversaries, we propose a new definition of security against quantum adversary. Specifically, we define the notion of *quantum simulation soundness* (SS-NIZK), that allows the adversary to access the simulator in superposition.
We show a separation between post-quantum and quantum security of SS-NIZK, and prove that Sahai's construction for SS-NIZK (in the CRS model) can be made quantumly-simulation-sound. As an immediate application of our new notion, we prove the security of the Naor-Yung paradigm in the quantum settings, with respect to a strong *quantum* IND-CCA security notion. This provides the quantum analogue of the classical *dual key approach* to prove the security of encryption schemes. Along the way, we introduce a new notion of quantum-query advantage functions, which may be used as a general framework to show classical/quantum separation for other cryptographic primitives, and it may be of independent interest.

1 Introduction

The rapid progress in quantum computing and the dramatic effects of a full-scale quantum computer on cryptography [Sho94] have prompted the community to re-examine traditional assumptions and security model in cryptography. Even though replacing the quantumly broken computational assumptions with quantum-hard assumptions is essential, a number of works (e.g., [Wat06, Unr12, DFNS14, KLLN16]) has shown attacks against cryptographic schemes that could be proven secure classically. Even worse, many of the current security models fail to capture the quantum nature of the adversary, thus only provide weak, or over-optimistic guarantees.

This has motivated a fruitful line of research, where new security definitions have been developed in the *superposition-access model* [KM10, KM12, Zha12a, DFNS14, BZ13a, BZ13b, ATTU16, KLLN16, AMRS20, BBC+21, CETU21, ABKM22, EvW22, CEV22] and many cryptographic schemes have been proven (in)secure in this model. In the superposition-access model, a quantum adversary may be able to run the cryptographic primitive in superposition of inputs. This is motivated by modeling the attacker's power realistically, e.g., when they are given the description of a hash function modeled as a

E-mail: behzad.abdolmaleki@sheffield.ac.uk (Behzad Abdolmaleki), celine.chevalier@ens.fr (Céline Chevalier), eebrahimi.pqc@gmail.com (Ehsan Ebrahimi), giulio.malavolta@hotmail.it (Giulio Malavolta), quoc.huy.vu@ens.fr (Quoc-Huy Vu)

Received: 2024-10-05 Accepted: 2024-12-03

random oracle, or simply as a worst-case guarantee, where one wants to minimize physical assumptions regarding the underlying hardware. Superposition security also becomes relevant further, when cryptographic primitives are used as part of a (possibly quantum) protocol.

Non-Interactive Zero-Knowledge. Non-Interactive Zero-Knowledge proofs (NIZKs) [GMR85, BFM88] are a cornerstone of modern cryptography and one of the most well-studied primitives in the field. A NIZK allows a party to prove the validity of an arbitrary NP statement in a single message (the proof) in a publicly verifiable way, without revealing anything beyond its validity. NIZKs find applications in basic cryptographic primitives and are extensively used in the design of privacy-preserving systems (such as credentials and digital currencies) [BG90, CL01, BCC+09, CKL+16, FHS19] as well as group signatures [Cv91, BBS04, BCC+16] and verifiable computation [GGP10, GGPR13, BCG+18].

The notion of *simulation-soundness* is a strong property satisfied by NIZKs [Sah99a] which has proven very useful for proofs of protocols in complex security models, such as the universal composability [Can01]. Informally, we say that a NIZK proof system is simulation-sound (SS-NIZK) if a malicious prover that has observed simulated proofs for a polynomial number of (possibly false) statements is not able to convince the verifier on either a new false statement or on an old false statement with a new proof. We also note that simulation soundness implies non-malleability, where an accepting proof cannot be modified into a different one without knowing the witness. Indeed, simulation-soundness is arguably a property that one requires when composing NIZK with other protocols to attain provably security guarantees in many cryptosystem, such as IND-CCA encryption schemes via the Naor-Yung paradigm [NY90, LNPT20] or signature schemes [Gro06, Unr15, KLS18]. Furthermore, schemes lacking this property have been attacked in practice (see examples in [DG23]).

1.1 NIZK in a Quantum World

In this work, we initiate the study of NIZK secure against superposition attacks, and in particular we introduce the notion of *quantum simulation-soundness* to address the lack of a rigorous definition of quantum security in the literature. Informally, quantum simulation-soundness implies that the adversary cannot maul NIZK proofs, even if given superposition access to a prover. Our motivations are:

(i) On the application side, quantum simulation-soundness is needed to prove the security of signature schemes [Gro06, Unr15, KLS18], with respect to stronger unforgeability definitions developed in [BZ13b, AMRS20].

(ii) In addition, such a definition would be needed if one wants to prove the security of the Naor-Yung paradigm [NY90] with respect to a quantum IND-CCA notion that allows quantum challenge queries [CEV22]. Traditional simulation-sound definitions (including Unruh's post-quantum definition [Unr15]) would fail to prove the security in these settings, since a CCA adversary would indeed have some degree of superposition access to the NIZK simulator. Indeed, we note that the Naor-Yung paradigm is a powerful technique, which is used in many cryptographic constructions and different cryptographic setting, for example, key-dependent message [CCS09] and key-leakage attacks [NS09]. Thus, having a quantum security analysis of the Naor-Yung paradigm would be important for many applications.

(iii) Our work is well-aligned with the general motivation for studying security of classical cryptosystems against superposition attacks. Indeed, one of the main motivations for the study of *post-quantum* cryptography is the threat of retroactive attacks, the

so-called *"store now, decrypt later attacks"*. This threat is certainly relevant in the context of encryption, and only partially effective against digital signatures and other cryptographic protocols. We argue that if post-quantum security is cryptographically fully relevant (i.e., when universal quantum computers become available), it makes sense to say that *quantum* security (against superposition attacks) is also relevant. One reason is that, once quantum computers are available, they should be used in a hybrid computation model together with classical computers.[1] In this setting, our motivation boils down to the final argument below.

(iv) Finally, from a theoretical standpoint, we view this notion as a conservative version of security for NIZK, where one places minimal assumptions on the underlying hardware: In the world where end users run quantum computers, the use of "classical computers" boils down to the assumption on the hardware to ensure that there are no quantum effects in the classical computers running the algorithms. This is similar to the line of research on device-independent quantum key distribution (QKD) [VV19], where one does not want to make any assumption on the hardware implementing QKD.

To address these issues, we propose a definition of *quantum simulation-soundness* for NIZKs, where an adversary is allowed quantum access to the oracle that is generating simulated proofs. This definition cannot be cast as a simple translation of the classical variant, because of the no-cloning theorem [WZ82]: The main challenge is that quantumly we cannot keep track of the quantum queries (since it could be just impossible to copy such states) and therefore it is not clear how to rule out the "trivial" adversary that simply forwards a simulated proof that they received from the simulator as the output of a query. To bypass this hurdle, we first resort to a more restricted security model, where the simulator is a classical algorithm. Our approach leverages the ability to track the randomness used in the security definition, as the challenger controls this randomness, which is classical and can be recorded. We believe this insight provides a useful framework for addressing the quantum recording barrier in similar contexts. However, we acknowledge that our definition is limited to classical simulators, which restricts its applicability in certain scenarios. Extending this approach to a more general setting, where the simulator itself can be quantum, remains an open problem.

1.2 Organization

The rest of the paper is organized as follows.

First, in Section 2, we discuss the difficulty of defining a sensible quantum simulation-soundness definition when the quantum adversary has superposition access to a classical simulator. We explain why the existing approaches for a similar scenario do not help to achieve our goal. To remedy this, we take a totally different approach and we use the fact that the randomness used to generate a simulated proof can be classical and therefore can be recorded. The formal definitions, along with some technical discussions, are presented in Section 4.

Then, we show that our definition is strictly stronger that the classical definition by giving two separation results between post-quantum and quantum security in Section 5 and Appendix F. As a side result, we introduce a new notion of *quantum-query advantage functions*, which are functions that demonstrate advantages of quantum queries over classical queries. We believe that our notion of quantum-query advantage functions can be used to provide a generic framework to show separations of cryptographic primitives between the classical-query setting and the quantum-query setting.

[1] After all, it is hard to justify that we build quantum computers only to break the current classical cryptography.

Next, we show that our quantum simulation-soundness is achievable. More precisely, we prove the quantum security with respect to our quantum simulation-soundness definition of Sahai's NIZK protocol [Sah01, DDO+01] in the CRS model in Appendix E, as well as the quantum security of Fiat-Shamir construction [FS87] in Section 6.

Finally, we discuss how a quantum simulation-sound NIZK protocol can be used to construct a quantum IND-CCA secure encryption scheme using the Naor-Yung construction [NY90, Sah99a] in Section 7.

2 Technical Overview

2.1 Definitions: Quantum Simulation-Soundness

Informally, a non-interactive zero-knowledge proof system is said to be *simulation-sound* if it has the property that an adversary cannot provide a convincing proof for any false statement, even if it has seen *simulated proofs* of arbitrary statements (including false statements).

In the traditional experiments, classical simulation-soundness is defined with respect to a zero-knowledge simulator. The adversary can obtain a polynomial number of simulated proofs, some of which possibly on false statements. The simulator would keep a list of statements and simulated proofs, and the adversary is asked to output a new pair of statement and proof that is outside the list, meaning that it wins if it manages to give either a proof on a new false statement, or a new proof of an old statement. Definition for simulation-soundness thus inherently implies recording and comparison of queries.

Translating this definition into the quantum setting is thus highly non-trivial because of several technical obstacles related to recording and comparison of quantum queries, which are linked to quantum no-cloning and the destructiveness of quantum measurements.[2] We note that the same barrier usually appears when one wants to define quantum security of classical primitives under superposition attacks, for example, in the case of message-authentication code [BZ13a], signatures [BZ13b] and encryption [BZ13b, CEV22]. Since one might find some similarities between the notion of simulation-soundness and the notion of "existential unforgeability under a quantum chosen message attack" for message authentication codes and signature schemes, we present below the road that we have taken to reach the final definition start by first giving some discussions on the existing approaches for defining quantum-secure unforgeability.

(n+1) approach. To define quantum-secure unforgeability, Boneh and Zhandry overcame the recording barrier by allowing the adversary to make n quantum queries to its oracle (respectively the MAC oracle or the signing oracle) and requiring it to output $n+1$ distinct valid classical (respectively (message, tag) or (message, sign)) pairs as an output [BZ13a, BZ13b]. At first glance, one may think that this technique can be used to define a quantum simulation-soundness. Namely, the adversary would make n quantum queries to the simulator and at the end it would be required to output $n+1$ distinct valid classical (statement, proof) pairs. We call this definition $(n+1)$-definition.

However, this is not a sensible definition because even when restricted to classical queries, it deviates from the classical simulation-soundness definition. In more details, a quantum adversary that makes n classical queries and obtains an extra valid pair (true statement, proof) would break the $(n+1)$-definition but not the classical simulation-soundness definition since the last pair is a proof of a true statement. To remedy this problem, one naive solution is that we allow the adversary to make n quantum queries on *true* statements and m quantum queries on *false* statements. At the end, the adversary would be required to return n valid pairs with true statements and $m+1$ valid pairs

[2] Zhandry had a partiel success on recording quantum queries for random oracles in [Zha19].

with false statements. We would call this definition $(n+m+1)$-definition. However, a polynomially-bounded simulator would not be able to distinguish whether a quantum query is a superposition of true statements, or a superposition of false statements, or else a mix of both true and false statements, without disturbing the query state. Indeed, the adversary would always be able to win. For example, the adversary could make $n-1$ queries on *true* statements and $m+1$ queries on *false* statements without being detected and use them to break the $(n+m+1)$-definition.

Overall, the conclusion is that Boneh-Zhandry's approach does not work for defining quantum simulation-soundness due to a subtle difference with their setting. This is because, in our case the simulator gets as inputs both true and false statements, but the adversary wins if it outputs a different valid pair with a false statement. In contrast, the inputs to a MAC or signing oracle all belong to a single message space and a new pair of (message, tag) or (message, signature) is a break of the security.

Strong blind-unforgeability approach. Another approach that might help is to use a notion similar to "blind-unforgeability" [AMRS20] that is defined for message authentication codes. In a nutshell, in the blind-unforgeability notion, the oracle blinds a random subset of the domain for the adversary. The adversary wins if at the end of the execution it returns a forgery for a blinded message. In more details, at the beginning of the game the oracle chooses a blinded set B randomly by putting a message m in B with probability ϵ. Then, for any message in B it returns $\perp$, otherwise, it answers with the MAC oracle. The adversary wins if it returns a forgery for a message in B. Then the strong blind-unforgeability is defined similarly with a minor difference: the blinded set is chosen from the set of all valid pairs (message, tag).

However, this approach also has some limitations. First, since the blinded set is chosen at the beginning of the protocol before any query has been made and this set is fixed during the execution, in a sense, the adversary is not able to adaptively influence this set based on the answers it gets. In addition, we are not certain a bounded reduction adversary can sample a blinded set when the message space is exponential-size. Of course this is not problematic when we restrict ourselves to classical adversaries — as proven in [AMRS20] that restricted to classical queries, the blind-unforgeability notion is equivalent to "EUF-CMA" (existential unforgeability under a chosen message attack) notion — since for each query the oracle can toss a biased coin with the probability ϵ being blind. Then, it either returns $\perp$ or answers with the MAC oracle. But, when quantum queries are allowed, the blinded set has to be determined at the beginning of the execution since a single superposition query can contain all the messages.

Another crucial problem with a quantum simulation-soundness definition in the blind-unforgeability style is the inability to use such a definition in applications like constructing an IND-CCA secure encryption scheme from a simulation-sound NIZK protocol and an IND-CPA secure public-key encryption scheme (the Naor-Yung paradigm) [Sah99b]. In particular, the protocol in [Sah99b] encrypts a message m using two different public keys to get two ciphertexts c_0, c_1. Then it invokes a NIZK protocol to prove in π that c_0, c_1 are indeed the encryptions of the same message. In the CCA security proof, when reducing to the CPA security of the underlying public-key encryption, the reduction adversary has to use only one secret key to simulate the decryption oracle. This simulation is possible because the adversary can not generate a new valid pair $((c_0, c_1), \pi)$ where (c_0, c_1) is not in the language (or where c_0, c_1 are the encryptions of two different messages) by the simulation-soundness property of the NIZK protocol. However, a definition in the blind-unforgeability style might not fit here because such a definition does not prevent an adversary that returns a valid pair $((c'_0, c'_1), \pi') \notin B$ (where B is the blinded set) where (c'_0, c'_1) is a false statement and it is different from the challenge ciphertext. Note that the actual decryption oracle on (c'_0, c'_1) returns $\perp$ but a simulated decryption oracle that uses

one secret key returns a message. Therefore, the reduction adversary encounters obstacles with a quantum simulation-soundness definition in the blind-unforgeability style.

Our approach. In this paper, we follow a totally different approach. In the following discussion, we only consider zero-knowledge systems with classical simulators. We note that most of known post-quantum construction of zero-knowledge systems in the CRS model are proven with classical simulators against quantum adversaries.[3] We then take advantage of the fact that the randomness to compute a proof is chosen by the oracle and can be classical. For each query, we thus ask the oracle to record the randomness used to respond the query in a list R.

Then, the first idea would be that when the adversary outputs a valid pair (x, π) with a false statement x, the simulator $\mathcal{S}$ would compute proofs for this statement x using all the randomnesses recorded in R. That is, for any $r_i \in R$, $\pi_i \leftarrow \mathcal{S}(x, r_i)$. If none of these computed proofs π_i is the same as the proof given by the adversary, the oracle would declare that the adversary has won.

However, this definition does not work since we observe that restricted to the classical queries, this definition is weaker than the classical definition. Indeed, if an adversary breaks this definition, it can break the classical definition as well since the given proof is a new proof that has not been obtained from the simulator. But the other direction does not always hold. If an adversary breaks the classical definition by outputting a valid pair (x, π) where x is a new false statement and there exists a recorded randomness that matches with this pair (that is, $\exists\ r \in R$ such that $\mathcal{S}(x, r) = \pi$), we can not use this pair to reject this definition.

Final definition. Lastly, we present our final definition to overcome the issue discussed above. Similarly, the oracle records the randomnesses in a list R and we assume that it picks a fresh randomness for each query. At the end, the adversary returns two pairs $(x_1, \pi_1),\ (x_2, \pi_2)$ and wins if either of the two following statements is true:

1. None of the randomnesses in R matches with one of the pairs $(x_1, \pi_1),\ (x_2, \pi_2)$ with x_1 or x_2 being a false statement.

2. There exists a randomness $r \in R$ that matches with these two pairs and one of the statements x_1 or x_2 is a false statement.

Intuitively, the first case means that the adversary managed to give a new proof of a false statement. Since we assume that the simulator picks a fresh randomness for each query, the second case implies either a malleable attack, or that the adversary managed to give a proof of a new false statement. Restricted to classical queries, this definition is stronger than the classical simulation-soundness definition. In more details, an adversary that breaks the classical definition outputs a new pair (x, π) where x is a false statement. If x has not been queried before, the reduction adversary can break one of the two cases above. If x has been queried before but π is a new proof, the randomness to generate π should not be in R and the reduction adversary breaks case 1. A formal proof of this claim is given in Lemma 1.

Finally, we conclude this section with some discussion on our choice of security model. We argue that a definition where the simulator is classical, while the adversary (or distinguisher) is quantum, is stronger than the standard quantum zero-knowledge definition, as it imposes stricter computational constraints on the simulator while still capturing the essence of security against quantum adversaries. Our approach relies essentially on the fact that in the classical setting, one can de-randomize classical algorithms However, this is not possible in the quantum setting, and we leave this interesting question of considering

[3] In general, non-rewinding simulators are typically PPT.

the general definition with quantum simulators (which captures many arguably practical constructions in the quantum random oracle model) for future work.

2.2 Separation

Separation I. The core of idea is to introduce a new notion of *quantum-query advantage functions*, which are functions that demonstrate advantages of quantum queries over classical queries, motivated by the recent notion of *quantum advantage functions*, introduced by [LMQW22].[4]These objects are then used to show a separation between quantum simulation-sound NIZKs and classical (post-quantumly secure) simulation-sound NIZKs. This separation relies on the hardness assumption of the Learning with Errors problem [Reg05] against quantum computers.

An overview idea of our construction of quantum-query advantage functions is as follows. We start with interactive proof of quantumness protocols [BCM+18]. Very briefly, an interactive proof of quantumness protocol is a demonstration of quantum *unsoundness* and classical soundness: a quantum prover can easily make a classical verifier accepts, but no efficient classical prover can make the verifier accepts. Since the verifier in an interactive proof of quantumness protocol is classical, we note that the communication of the protocol is also classical. Our main observation is that an interactive proof of quantumness protocol (specifically the one given in [BCM+18]) can also be used to show quantum *unsoundness* and semi-quantum soundness: a quantum prover interacting with a quantum verifier using quantum communication can make the protocol accepts, while no efficient quantum prover with only classical communication can make the protocol accepts. We formalize this idea by introducing the notion of *quantum-query advantage* functions, and use it to show separation between the quantum and post-quantum security settings for SS-NIZK. In particular, our separation is constructed by carefully embedding instances of interactive quantum-query advantage into the simulator of the NIZK system. The key conceptual insight is that although we are considering non-interactive proof systems, the security game for simulation-soundness is interactive, allowing us to use a quantum adversary that makes the quantum-query advantage function accept to also break the simulation-soundness: an efficient quantum adversary given *classical* oracle access to the quantum-query advantage function cannot cause it to ever output accept, while it can do so by only making 2 *quantum* queries.

Separation II. We also present an ad-hoc separation construction which holds unconditionally with no extra assumptions (see Appendix F). To this aim, we start with constructing a new pseudorandom function that is periodic with some large, secret period. Classical adversaries will not be able to detect the period, and thus cannot distinguish this new function from random. However, an adversary making quantum queries can detect the period, and thus distinguish our new function from random. In more detail, inspired by the idea of [KZM+15] with some modifications, our second separation's construction makes use of a one-time signature scheme. A pair of one-time signing/verification keys are generated for each proof such that in the zero-knowledge proof, a simulator (simulated prover) is required to provide $\mathsf{ct}_{\mathsf{F}} = \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$ (where the function F is a periodic pseudo-random function with a secret key $\mathsf{sk}_{\mathsf{PRF}}$) and $\mathsf{ct}_{\mathsf{pk}} = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk})$ (where f is a pseudo-random function with a secret key $\mathsf{sk}_{\mathsf{PRF}}$ and pk is a public key of an underlying public key encryption scheme). Then we require the prover to sign the statement together with the proof, the cipher-text, $\mathsf{ct}_{\mathsf{pk}}$, and ct_{F}. Then, briefly, due to the security of the signature scheme, the adversary must use a different x and pk from the ones returned from oracle queries. Thus, for a statement to pass the verifier without a proper witness, the classical prover must generate $\mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$ and $\mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk})$ without the knowledge of $\mathsf{sk}_{\mathsf{PRF}}$ (thus breaking the

[4]The notion defined in [LMQW22] demonstrate a quantum advantage with only *classical queries*, showing separations between *classical* security and *post-quantum* security.

pseudo-random function F and f). But given quantum queries and by using Simon's quantum algorithm, the quantum adversary can find the period of the pseudo-random function F and thus obtain $\mathsf{sk}_{\mathsf{PRF}}$ and breaks the simulation sound property.

2.3 Constructions and Applications

For the feasibility, in the common reference string model (CRS), we consider Sahai's construction of unbounded simulation-sound NIZK [Sah01, DDO+01], based on Naor commitment scheme [Nao90], and we show that when instantiated with quantum-secure one-time signature scheme (Definition 10), this scheme is also quantum simulation-sound.

Finally, we present and prove quantum security of a simple modification of the classical Naor-Yung scheme [NY90, Sah99a]. In particular, we give a construction of building quantum chosen-ciphertext secure encryption schemes from quantum chosen-plaintext secure schemes and quantum simulation-sound NIZK proof systems. It is a combination of quantum-secure $\mathsf{IND\text{-}CPA}$ encryption schemes and a family of invertible pseudorandom functions.

3 Preliminaries

Notation. Throughout this paper, λ denotes the security parameter. The notation $\mathsf{negl}(\lambda)$ denotes any function f such that $f(\lambda) = \lambda^{-\omega(1)}$, and $\mathsf{poly}(\lambda)$ denotes any function f such that $f(\lambda) = \mathcal{O}(\lambda^c)$ for some $c > 0$. For $a, b \in \mathbb{R}$, $[a, b] \coloneqq \{\, x \in \mathbb{R} \mid a \leq x \leq b \,\}$ and $[\![a, b]\!] \coloneqq \{\, x \in \mathbb{Z} \mid a \leq x \leq b \,\}$ will denote the closed real and integer interval with endpoints a and b. With an abuse of notation, we will write $[\![n]\!]$ as shorthand for $[\![0, n-1]\!]$. For a set $I = \{i_1, \ldots, i_\ell\} \subseteq [\![n]\!]$ and a n-bit string $x \in \{0,1\}^n$, we write $x|_I \coloneqq x_{i_1} \cdots x_{i_\ell}$. When sampling uniformly at random a value a from a set $\mathcal{U}$, we employ the notation $a \xleftarrow{\$} \mathcal{U}$. When sampling a value a from a probabilistic algorithm $\mathcal{A}$, we employ the notation $a \leftarrow \mathcal{A}$. Let $|\cdot|$ denote either the length of a string, or the cardinal of a finite set, or the absolute value. By PPT we mean a polynomial-time non-uniform family of probabilistic circuits, and by QPT we mean a polynomial-time family of quantum circuits. For a probabilistic algorithm f, we write $f(x; r)$ to denote the computation of f on input x with randomness r drawn uniformly at random. We sometimes omit the randomness and just write $f(x)$.

3.1 Quantum Computation

We assume familiarity with quantum information and computation, and refer to [NC11] and Section A.1 for the definition of basic concepts.

3.2 Non-interactive Zero-knowledge Proof Systems

For a NP relation $\mathcal{R} \subseteq \times\mathcal{W}$, we let $\mathcal{L}(\mathcal{R}) \coloneqq \{x : \exists w, (x, w) \in \mathcal{R}\}$. As discussed in the introduction, in this work, we only consider zero-knowledge systems with PPT simulators.

Definition 1. A non-interactive zero-knowledge (NIZK) proof system for an NP relation $\mathcal{R}$ in the common reference string (CRS) model consists of three PPT algorithms $\mathsf{NIZK} = \langle \mathsf{Setup}, \mathcal{P}, \mathcal{V} \rangle$:

- $\mathsf{crs} \leftarrow \mathsf{Setup}(1^n, 1^\lambda)$. On input a statement of length n and the security parameter λ, the setup algorithm Setup outputs a common reference string crs.
- $\pi \leftarrow \mathcal{P}(\mathsf{crs}, x, w)$. On input the common reference string crs, an instance x and a witness w such that $(x, w) \in \mathcal{R}$, the proving algorithm $\mathcal{P}$ outputs a proof π.

- $b \leftarrow \mathcal{V}(\mathsf{crs}, x, \pi)$. On input the common reference string crs, an instance x and a proof π, the verification algorithm $\mathcal{V}$ outputs a bit $b \in \{0,1\}$. If $b = 1$, we say that $\mathcal{V}$ $\mathsf{accepts}$, otherwise we say that $\mathcal{V}$ $\mathsf{rejects}$.

The proof system NIZK must satisfy the following requirements for all $\lambda \in \mathbb{N}$.

- **Completeness.** For every $(x, w) \in \mathcal{R}$, we have that

$$\Pr\Big[\mathcal{V}(\mathsf{crs}, x, \mathcal{P}(\mathsf{crs}, x, w)) = 1 \mid \mathsf{crs} \leftarrow \mathsf{Setup}(1^{|x|}, 1^\lambda)\Big] = 1,$$

  where the probability is taken over the randomness of Setup and $\mathcal{P}$.

- **Statistical soundness.** There exists a negligible function $\mathsf{negl}(\lambda)$ such that for any $n \in \mathbb{N}$,

$$\Pr_{\mathsf{crs} \leftarrow \mathsf{Setup}(1^n, 1^\lambda)}[\exists (x, \pi^*) \text{ s.t. } \mathcal{V}(\mathsf{crs}, x, \pi^*) = 1 \wedge x \notin \mathcal{L}] \leq \mathsf{negl}(\lambda).$$

- **(Adaptive) post-quantum computational zero-knowledge.** There exists a PPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ such that for all QPT malicious verifier $\mathcal{V}^* = (\mathcal{V}_1^*, \mathcal{V}_2^*)$, for every $n \in \mathbb{N}$,

$$\left|\Pr\left[\begin{array}{c|c} \mathcal{V}_2^*(\mathsf{crs}, x, \pi, \zeta) = 1 \wedge x \in \mathcal{L} & \begin{array}{c}\mathsf{crs} \leftarrow \mathsf{Setup}(1^n, 1^\lambda) \\ (x, w, \zeta) \leftarrow \mathcal{V}_1^*(\mathsf{crs}) \\ \pi \leftarrow \mathcal{P}(\mathsf{crs}, x, w)\end{array}\end{array}\right]\right.$$
$$\left.-\Pr\left[\begin{array}{c|c} \mathcal{V}_2^*(\mathsf{crs}, x, \pi, \zeta) = 1 \wedge x \in \mathcal{L} & \begin{array}{c}(\mathsf{crs}, \mathsf{td}) \leftarrow \mathcal{S}_1(1^\lambda) \\ (x, w, \zeta) \leftarrow \mathcal{V}_1^*(\mathsf{crs}) \\ \pi \leftarrow \mathcal{S}_2(\mathsf{td}, x)\end{array}\end{array}\right]\right| \leq \mathsf{negl}(\lambda),$$

  where ζ is the internal state of $\mathcal{V}^*$.

Definition 2 (Unbounded Simulation-Soundness)**.** A zero-knowledge proof system is said to be (unbounded) *simulation-sound* if it has the property that an adversary cannot provide a convincing proof for any false statement, even if it has seen *simulated proofs* of arbitrary statements (including false statements). More precisely, an NIZK proof is simulation sound if for all QPT adversaries $\mathcal{A}$, we have:

$$\Pr\left[\begin{array}{c|c} (x_i, \pi_i) \notin Q \wedge\ x \notin \mathcal{L} & (\mathsf{crs}, \mathsf{td}) \leftarrow \mathcal{S}_1(1^\lambda) \\ \wedge\ \mathcal{V}(\mathsf{crs}, x, \pi) = 1 & (x, \pi) \leftarrow \mathcal{A}^{\mathcal{S}_2(\mathsf{td}, \cdot)}(\mathsf{crs})\end{array}\right] \leq \mathsf{negl}(\lambda),$$

where Q is the list of simulation queries and responses (x_i, π_i).

3.3 Pseudorandom Functions

A family of pseudorandom functions (PRFs) consists of two polynomial time classical algorithms $\langle \mathsf{KeyGen}, \mathsf{PRF} \rangle$. KeyGen is a randomized procedure that takes as input the security parameter and outputs a random key $\mathsf{k} \in \mathcal{K}$. PRF takes as input a key $\mathsf{k} \in \mathcal{K}$ and an input $x \in$, and deterministically outputs a classical string $y \in \mathcal{Y}$. In this paper, we also consider the notion of invertible pseudorandom function which is an injective PRF whose inverse function PRF^{-1} can be computed efficiently (given the secret key). We recall the formal security definition of (invertible) PRFs in Section A.2.

3.4 One-time Signatures

A signature scheme consists of three polynomial time classical algorithms $\mathsf{Sig} = \langle \mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verif} \rangle$. KeyGen is a randomized procedure that takes as input the security parameter and produces a secret key and public key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$. Sign takes as input the secret key and a message m, and produces a signature $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$. Finally, Verif takes as input the public key, a message m, and a supposed signature σ on m, and either accepts or rejects. We recall the formal security definition of one-time signatures in Section A.3.

4 Quantum Security Definition for Simulation-Sound Non-Interactive Zero-Knowledge

In this section, following the ideas described in Section 2, we present our definition of quantum simulation-soundness, and give some discussions on these definitions.

4.1 Quantum-Secure Zero-Knowledge

We give below the definition for quantum-secure zero-knowledge in the common reference string model. Our definition is a quantum counterpart of the classical definition for post-quantum zero-knowledge: the only difference is that now the adversary can query the simulator in superposition.

Definition 3. Let $\mathcal{L}$ be a language in NP. A proof system $\mathsf{NIZK} \coloneqq \langle \mathsf{Setup}, \mathcal{P}, \mathcal{V} \rangle$ for $\mathcal{L}$ is (adaptive) quantum-secure zero-knowledge if there exists a PPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ such that for all QPT distinguisher $\mathcal{D}^* = \{\mathcal{D}^*_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$, for every $n \in \mathbb{N}$, and for every $\lambda \in \mathbb{N}$:

$$\Big| \Pr\Big[\mathcal{D}^*_\lambda(\rho_\lambda, \mathsf{crs})^{|\mathcal{P}^{\mathcal{O}}(\mathsf{crs},\cdot,\cdot)\rangle} = 1 \;\Big|\; \mathsf{crs} \leftarrow \mathsf{Setup}(1^n, 1^\lambda) \Big] - \Pr\Big[\mathcal{D}^*_\lambda(\rho_\lambda, \mathsf{crs})^{|\mathcal{S}_2(\mathsf{td},\cdot)\rangle} = 1 \;\Big|\; (\mathsf{crs}, \mathsf{td}) \leftarrow \mathcal{S}_1(1^\lambda) \Big] \Big| \leq \mathsf{negl}(\lambda),$$

where

- $\mathcal{D}^*$ can make quantum queries to the oracles.
- $\mathcal{P}(\mathsf{crs}, \cdot, \cdot)$ is the prover algorithm and $\mathcal{S}_2(\mathsf{td}, \cdot)$ only acts on its private trapdoor td, the input statement $x \in \mathcal{L}_{yes} \cap \{0,1\}^n$ and its private random tape.

Randomness. We recall the following discussion from [BZ13b]. If an oracle $\mathcal{O}$ implements a classical randomized algorithm, there are several choices for how the randomness is used in each query if the oracle is queried in superposition. One option is to choose fresh randomness for each message in the superposition. Another option is to choose a single randomness value for each query, and generate output in the superposition with that randomness. We note that there is a simple transformation that converts an oracle requiring independent randomness for every message into a scheme that is secure when a single randomness value is used for an entire query: for each query, choose a fresh random key k for a quantum pseudorandom function (QPRF) ([Zha12a]). This will be the single per-query randomness value. Each message m in the superposition will be answered using randomness obtained by applying the QPRF to m using the key k. From the adversary's point of view, this is indistinguishable from choosing independent randomness for each message. Indeed, Zhandry [Zha12b] shows that we can replace the QPRF with a function drawn from a pairwise independent function family, which allows us to achieve perfect simulability. For this reason, requiring global randomness per query does not change the

oracle from the adversary's point of view, but greatly simplifies its implementation. In this work, we choose the second approach and all randomized oracles are implemented this way.

Construction. We note that any *perfect* adaptive post-quantum zero-knowledge proof system is also (perfect) quantum-secure zero-knowledge proof system. In Appendix D, we also give a brief discussion on the feasibility of achieving *computationally* quantum-secure zero-knowledge proof systems from known constructions.

4.2 Quantum Simulation-Soundness

Definition 4 (Quantum Simulation-Sound NIZK)**.** Let $\mathcal{L}$ be a language in NP. Consider a proof system $\langle \mathsf{Setup}, \mathcal{P}, \mathcal{V}\rangle$ for $\mathcal{L}$ with PPT zero-knowledge simulator $\mathcal{S} \coloneqq (\mathcal{S}_1, \mathcal{S}_2)$. In each query, $\mathcal{S}_2$ stores the randomness used to answer the query in a list R. A QPT adversary $\mathcal{A}$ after making polynomial numbers of quantum queries to $\mathcal{S}_2$ outputs two pairs $\{(x_i, \pi_i)\}_{i=1}^2$. The adversary $\mathcal{A}$ wins if either of the following two cases hold:

1. There exists $i \in [\![1,2]\!]$ such that $x_i \notin \mathcal{L}$, for all $r \in R$, $\mathcal{S}_2(x_i, r) \neq \pi_i$ and $\mathcal{V}(\mathsf{crs}, x_i, \pi_i) = 1$.
2. There exists a randomness $r \in R$ such that $\mathcal{S}_2(x_1, r) = \pi_1$ and $\mathcal{S}_2(x_2, r) = \pi_2$ and at least one of x_1 or x_2 is not in $\mathcal{L}$.

Formally, we say an NIZK proof is *quantum simulation-sound* if for all $\lambda \in \mathbb{N}$, for all QPT adversaries $\mathcal{A}$, $i, j \in [\![1,2]\!]$ we have:

$$\Pr\left[\begin{array}{c|c} \mathcal{V}(\mathsf{crs}, x_i, \pi_i) = 1 \,\forall\, i \,\wedge & \\ \exists\, i : \Big(\big(x_i \notin \mathcal{L}\big) \,\wedge & (\mathsf{crs}, \mathsf{td}) \leftarrow \mathcal{S}_1(1^\lambda) \\ \big((\mathcal{S}_2(\mathsf{td}, x_i, r) \neq \pi_i \,\forall\, r \in R) \vee & \{(x_i, \pi_i)\}_{i=1}^2 \leftarrow \mathcal{A}^{|\mathcal{S}_2(\mathsf{td},\cdot)\rangle}(\mathsf{crs}) \\ (\exists\, r \in R : \mathcal{S}_2(\mathsf{td}, x_j, r) = \pi_j \,\forall\, j)\big)\Big) & \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Next, we show that our quantum simulation-soundness (with respect to only classical queries) implies the standard classical simulation-soundness.

Lemma 1. *The classical restriction of our quantum simulation-soundness (where the adversary can only make classical queries to the simulator) implies the standard classical simulation-soundness (Definition 2).*

Proof. We show that if there is an adversary $\mathcal{A}$ that breaks the standard classical simulation-soundness (Definition 2), we can use $\mathcal{A}$ to build another adversary $\mathcal{B}$ that breaks our quantum simulation-soundness restricted to classical queries.

$\mathcal{B}$ simply forwards all (classical) queries from $\mathcal{A}$ to its oracle, and keeps a list T of $\mathcal{A}$'s queries and responses (since now they are all classical). When $\mathcal{A}$ outputs a pair (x, π), $\mathcal{B}$ picks at random a pair (x', π') from T and outputs (x, π) and (x', π'). There are two cases:

- If $x \in T$, then (x, π) breaks the case 1 of our quantum simulation-soundness. This is because, in this case π cannot be an answer from the oracle: that is, $\pi \notin T$ (this is by definition of the standard classical simulation soundness). Then the pair (x, π) would pass the check of the first condition: Note that with the randomness r and input a statement x, the proof generation function (computed by the simulator) is a deterministic function of r and x, so if the output proof π is not the one generated by the simulator, the randomness used to generate π must not also be in the list of randomness kept by the simulator.

- If $x \notin T$, this breaks either one of the two cases. If there is no collision on the randomness used to compute (x, π) with the recorded list R of the simulator's randomness, this breaks case 1. Otherwise, with probability $\frac{1}{q}$ (where q is the number of the queries), (x, π) and (x', π') would be computed with the same randomness and this breaks case 2. We note that in the latter case, there is a security loss growing with the number of queries. □

4.2.1 Some Technical Discussions

We now remark on a few details on the notion of quantum simulation-soundness. First, it might seem that our definition does not capture all possible *quantum* attacks. Consider the following adversary $\mathcal{A}$. $\mathcal{A}$ makes a quantum query to the simulator and obtains a superposition of statements and proofs as $\sum_{x \neq x_0, y} \alpha_{x,y} |x, y \oplus \pi\rangle$, where $x_0 \notin \mathcal{L}$. We assume that the simulator answered the query with a classical randomness r, that is hidden from $\mathcal{A}$. (Note that, if r is not hidden, there is a trivial attack.) $\mathcal{A}$ then performs some quantum computation to come up with a proof for x_0, with *the same* randomness r, and during the process, $\mathcal{A}$ also destroys the original state, thus $\mathcal{A}$ cannot procedure two pairs of $\{(x_i, \pi_i)\}_{i=1}^2$ that are computed using the same randomness. Essentially, the adversary makes one or more quantum queries but then must consume the post-query states completely in order to make a single, but convincing, forgery. Obviously, if such an adversary $\mathcal{A}$ exists, this might consider a quantum attack against quantum simulation-soundness, but Definition 4 does not capture this. However, this so-called attack is inherited from the nature of quantum queries and can be applied in similar scenarios, for instance, the $(n+1)$-definition proposed in [BZ13b, BZ13a]. We also note that the blind-unforgeability definition of [AMRS20] is designed to avoid this kind of attacks, however, it is unknown whether blind-unforgeability is stronger than $(n+1)$-definition.[5] The second condition in Definition 4 is thus used to capture classical attacks rather than quantum attacks. We leave this as an open problem, either to find a concrete example for this type of attack, or to show that (in most of the cases) this is not possible.

Secondly, our definition also captures some "malleability" attack that is not captured by the classical definition. In particular, imagine that if the adversary makes a query to the simulator for a statement $x_1 \notin \mathcal{L}$, and outputs a proof for a statement $x_2 \in \mathcal{L}$ with the same randomness used by the simulator. This attack does not violate the classical simulation-soundness, but it is captured by our definition. This is because it is not possible in general to distinguish which statement was queried by the adversary in the quantum setting. We note that the "inverse" case (that is, $x_1 \in \mathcal{L}$ and $x_2 \notin \mathcal{L}$) is obviously an attack and it is captured in both classical and quantum notions.

5 Separation Between Post-Quantum and Quantum Security for SS-NIZK

In this section, we introduce a new notion of *quantum-query advantage functions*, which are functions that demonstrate advantages of quantum queries over classical queries. After recalling the definition of interactive proof of quantumness protocols in Section 5.1, we give the definition and construction for quantum-query advantage functions in Section 5.2, and we use them to show a separation between quantum simulation-sound NIZKs and classical simulation-sound NIZKs in Section 5.3.

[5] The conference version of [AMRS20] claimed that blind-unforgeability implies $(n+1)$-definition, which was removed in [AMRS23].

5.1 Preliminaries: Interactive Proof of Quantumness

We first recall the definition of interactive proof of quantumness protocols with 4 messages in total, which corresponds to the best round complexity known for interactive proofs of quantumness in the plain model [BCM+18].

Definition 5. An interactive proof of quantumness is an interactive protocol Π_{ipq} between a prover $\mathcal{P}$ and a verifier $\mathcal{V}$ using classical communication, with the following properties:

- **Quantum completeness:** there exists a QPT quantum prover $\mathcal{P}$ such that for all $\lambda \in \mathbb{N}$:
$$\Pr\big[(\mathcal{P},\mathcal{V})(1^\lambda)=1\big] \geq 1-\mathsf{negl}(\lambda).$$
- **Classical soundness:** for any PPT classical prover $\mathcal{P}^*$, for all $\lambda \in \mathbb{N}$:
$$\Pr[(\mathcal{P}^*,\mathcal{V})=1] \leq \mathsf{negl}(\lambda).$$

In a 4-round interactive proof of quantumness protocol, the first message is sent by the verifier to the prover. Let v_1, v_2 (resp. p_1, p_2) denote the messages sent by the verifier (resp. the prover) during the execution of an interactive proof of quantumness Π_{ipq}. An interactive proof of quantumness Π_{ipq} can furthermore satisfy the following optional property:

- **Public-coin second verifier message:** the second verifier message v_2 consists of uniformly and independently sampled random coins.
- **Semi-quantum soundness:** for any QPT *quantum* prover $\mathcal{P}^*$, for all $\lambda \in \mathbb{N}$:
$$\Pr\big[(\mathcal{P}^*,\mathcal{V}_{\mathsf{semi}})(1^\lambda)\big] \leq \mathsf{negl}(\lambda),$$
where the verifier $\mathcal{V}_{\mathsf{semi}}$ is defined as follows.
  - Let $\mathcal{P}$ denote the efficient quantum prover for Π_{ipq} such that
  $$\Pr\big[(\mathcal{P},\mathcal{V})(1^\lambda)\big] \geq 1-\mathsf{negl}(\lambda).$$
  We further assume that the first message generation algorithm of $\mathcal{P}$ is a QPT algorithm in the following form: this algorithm runs a classical PPT algorithm P_1 in superposition of inputs (possibly followed by a measurement in the computational basis).
  - $\mathcal{V}_{\mathsf{semi}}$ runs $\mathcal{V}$ to obtain the first verifier message v_1.
  - Whenever $\mathcal{V}_{\mathsf{semi}}$ receives a classical message x from $\mathcal{P}^*$, it runs P_1 on (v_1, x) and obtains a classical message p_1.
  - $\mathcal{P}^*$ is allowed to send a *classical* message x to $\mathcal{V}_{\mathsf{semi}}$ and receive back a tuple of classical message (p_1, v_2) where v_2 is the second verifier message. Then it outputs a classical message p_2.
  - $\mathcal{V}_{\mathsf{semi}}$ outputs the output of $\mathcal{V}$ on (v_1, p_1, v_2, p_2).

  Intuitively, semi-quantum soundness guarantees that no efficient quantum prover can cheat when the first prover message is generated by a *classical* prover.

Lemma 2. *Under the LWE assumption, there exists a 4-message interactive proof of quantumness protocol satisfying: (1) public-coin second verifier message and (2) semi-quantum soundness.*

The proof of this lemma can be found in Appendix B.

5.2 Quantum Advantage with Quantum Query Algorithms

Definition 6 (Quantum-Query Advantage Functions)**.** A *quantum-query* advantage function is a pair of PPT algorithms $\langle \mathsf{Setup}, \mathsf{QAF} \rangle$ with the following properties:

- $(\mathsf{pp}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$. On input a security parameter λ, the setup algorithm Setup outputs a public parameter pp and a secret key sk. Without loss of generality, we will consider that the secret key sk includes the public parameter pp.
- $\mathsf{QAF}(\mathsf{sk}, x)$. On input a secret key sk and a message x, the (randomized) evaluation algorithm QAF outputs either a message y, or a special "accept" symbol denoted accept, or a special "reject" symbol denoted reject. For our applications later, we require that by default $\mathsf{QAF}(\mathsf{sk}, \cdot)$ is stateless.

We additionally require the following properties:

1. q**-Quantum-query easiness.** For any $\lambda \in \mathbb{N}$, there exists a QPT oracle algorithm $\mathcal{A}^{|\mathsf{QAF}(\mathsf{sk},\cdot)\rangle}(\mathsf{pp})$ such that:

$$\Pr\Big[\mathsf{QAF}(\mathsf{sk}, x) = \mathsf{accept} \mid x \leftarrow \mathcal{A}^{|\mathsf{QAF}(\mathsf{sk},\cdot)\rangle}(\mathsf{pp})\Big] = 1 - \mathsf{negl}(\lambda),$$

   where $\mathcal{A}^{|\mathsf{QAF}(\mathsf{sk},\cdot)\rangle}$ makes q *quantum* queries in total to $\mathsf{QAF}(\mathsf{sk}, \cdot)$ before outputting x, and the probability is taken over $(\mathsf{pp}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$.

2. **Classical-query hardness.** For any $\lambda \in \mathbb{N}$, for all QPT oracle algorithm $\mathcal{A}^{\mathsf{QAF}(\mathsf{sk},\cdot)}(\mathsf{pp})$ such that:

$$\Pr\Big[\mathsf{QAF}(\mathsf{sk}, x) = \mathsf{accept} \mid x \leftarrow \mathcal{A}^{\mathsf{QAF}(\mathsf{sk},\cdot)}(\mathsf{pp})\Big] \leq \mathsf{negl}(\lambda),$$

   over $(\mathsf{pp}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$.

Construction. Let Π_{ipq} be a 4-message interactive proof of quantumness, in the form of Definition 5 in Section 5.1 with public-coin second verifier message and semi-quantum soundness properties. Let PRF be a one-wise independent PRF. We define our quantum-query advantage function below.

Construction 1 (A quantum-query advantage function.)**.** *Our construction is as follows.*

- $\mathsf{Setup}(1^\lambda)$*:*
  - *Run the first verifier message for* Π_{ipq} *to obtain* (v_1, r)*, where* v_1 *is the first verifier message and* r *is the private coin of the verifier of* Π_{ipq}*.*
  - *Sample a uniformly random key* $\mathsf{k} \xleftarrow{\$} \{0,1\}^\lambda$ *for* PRF*.*
  - *Set* pp *as an empty string and* $\mathsf{sk} := (\mathsf{pp}, \mathsf{k}, v_1, r)$ *and output* $(\mathsf{pp}, \mathsf{sk})$*.*
- $\mathsf{QAF}(\mathsf{sk}, \cdot)$*: on input a message* x*, we consider several distinguished cases (all cases are considered with appropriate input length):*
  - *If* x *is of the form* $(0\|u)$*: compute the semi-quantum verifier message for* Π_{ipq} *on* (v_1, u) *to obtain* (v_1, p_1, v_2)*, where* $v_2 \leftarrow \mathsf{PRF}(\mathsf{k}, p_1)$*. Output* (p_1, v_2)*.*
  - *If* x *is of the form* $(1\|p_1\|p_2)$*: compute* $v_2 \leftarrow \mathsf{PRF}(\mathsf{k}, p_1)$*. If the verifier of* Π_{ipq} *accepts the transcript* (v_1, p_1, v_2, p_2) *with the secret state* r*, output* accept*, otherwise output* reject*.*
  - *Otherwise output* $\perp$*.*

Theorem 1. *Let Π_{ipq} be a 4-message interactive proof of quantumness satisfying the properties specified in Lemma 2: public-coin second verifier message and semi-quantum soundness. Then there exists a quantum-query advantage function satisfying 2-quantum-query easiness (Definition 6).*

The proof of Theorem 1 follows from Lemma 3 and Lemma 4 stated below.

Lemma 3 (Quantum-query easiness)**.** *Suppose Π_{ipq} satisfies quantum completeness (Definition 5). Then Construction 1 satisfies quantum-query easiness.*

Proof. Let $\mathcal{P}$ denote the efficient quantum prover for Π_{ipq} such that

$$\Pr\left[\,(\mathcal{P},\mathcal{V})(1^\lambda)=1\,\right]\geq 1-\mathsf{negl}(\lambda).$$

Define the following QPT algorithm $\overline{\mathcal{P}}$:

- Make a (quantum) query $\sum_x |0\|x,0\rangle$ to $\mathsf{QAF}(\mathsf{sk},\cdot)$.
- Measure the response register to get a classical string p_1.
- Run $\mathcal{P}$ on p_1, v_2 and the post-measurement state to obtain p_2.
- Output $p_1\|p_2$.

Since PRF is one-wise independent, $\overline{\mathcal{P}}$ perfectly simulates the view of $\mathcal{P}$ in an instance of Π_{ipq}. By the completeness of Π_{ipq}, $\mathsf{QAF}(\mathsf{sk},\cdot)$ outputs accept with probability $1-\mathsf{negl}(\lambda)$. □

Lemma 4 (Classical-query hardness)**.** *Suppose Π_{ipq} has public-coin second verifier messages and has semi-quantum soundness (Lemma 2). Then Construction 1 satisfies classical hardness.*

The proof of this lemma follows the same idea as in [LMQW22]. The only difference is that we reduce to the semi-quantum soundness of Π_{ipq} defined above.

Combining Theorem 1 with Lemma 2, we obtain the following:

Corollary 1. *Assuming the (classical) hardness of LWE, there exists a quantum advantage function satisfying 2-quantum easiness (Definition 6).*

5.3 Separation between Post-Quantum Security and Quantum Security

Following the ideas presented in Section 2, we use our quantum-query advantage functions to give an example of a NIZK proof system that is classically simulation-sound but not quantum simulation-sound.

Construction 2. *Let $\mathcal{L}'$ be a language in NP, with the associated relation R'. Let $\mathcal{L}$ denote the NP language defined in Equation (1). Let $\Pi=\langle\mathsf{Setup},\mathcal{P},\mathcal{V}\rangle$ be a post-quantum simulation-sound non-interactive zero-knowledge proof system for $\mathcal{L}$, and $\langle\mathsf{Setup},\mathsf{QAF}\rangle$ be a quantum-query advantage function. We define the following NIZK proof system $\overline{\Pi}=\langle\overline{\mathsf{Setup}},\overline{\mathcal{P}},\overline{\mathcal{V}}\rangle$ for $\mathcal{L}$ as follows.*

- $\overline{\mathsf{Setup}(1^\lambda)}$ *: Output* $\mathsf{pp}\leftarrow\Pi.\mathsf{Setup}(1^\lambda)$. *(We note that* $\mathsf{pp}=\mathsf{crs}$ *in the CRS model, and* pp *is a token that allow the parties to make quantum queries to the random oracle in the QROM.)*
- $\overline{\mathcal{P}(\mathsf{pp},x,w)}$ *: Compute* $(\mathsf{pp}',\mathsf{sk})\leftarrow\mathsf{Setup}(1^\lambda)$. *Compute* $y\leftarrow\mathsf{QAF}(\mathsf{sk},x)$. *Generate a proof* π *using* Π *for the statement* $(x,y)\in\mathcal{L}$. *Output* $(y\|\pi)$.

- $\overline{\mathcal{V}(\mathsf{pp}, x, \overline{\pi})}$: *Parse* $(y\|\pi) \leftarrow \overline{\pi}$. *Output* $\mathcal{V}(\mathsf{pp}, (x, y), \pi)$.

We define the following augmented language $\mathcal{L}$ of R':

$$\mathcal{L} := \{(x, y) : \exists\, (\mathsf{sk}, r, w) : R'(x, w) = 1 \wedge (y = \mathsf{reject} \vee y = \mathsf{QAF}(\mathsf{sk}, x; r))\}. \tag{1}$$

It is easy to see that completeness and soundness of $\overline{\Pi}$ follow directly from those of Π.

We now construct a simulator $\overline{\mathcal{S}}$ for zero-knowledge property of $\overline{\Pi}$, which is later on also used in the proofs of simulation-soundness. Let $\mathcal{S} := (\mathcal{S}_1, \mathcal{S}_2)$ be a zero-knowledge simulator of Π. The simulator $\overline{\mathcal{S}} := (\overline{\mathcal{S}_1}, \overline{\mathcal{S}_2})$ works as follows.

- $\overline{\mathcal{S}_1}$: Output $\mathcal{S}_1$.
- $\overline{\mathcal{S}_2}$: Initialize an empty list Q. On the input a statement x,
  - For each pair $(\mathsf{pp}'_i, \mathsf{sk}_i) \in Q$, compute $y_i \leftarrow \mathsf{QAF}(\mathsf{sk}_i, x)$.
    - If $y_i = \mathsf{reject}\ \forall\ i$, set $y = \mathsf{reject}$.
    - If there exists an index i such that $y_i = \mathsf{accept}$, set $y = \mathsf{accept}$.
    - Otherwise, compute $(\mathsf{pp}', \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$, store $(\mathsf{pp}', \mathsf{sk})$ in Q and compute $y \leftarrow \mathsf{QAF}(\mathsf{sk}, x)$.
  - Run $\mathcal{S}_2$ on input (x, y) to obtain a simulated proof π.
  - If $y = \mathsf{accept}$, generate a simulated proof π' for a random *false* statement $x' \in \mathcal{L}$ (by sampling $x' \in \{0,1\}^n$ uniformly at random, where n is the length of a statement in $\mathcal{L}$) and output $(\pi, (x', \pi'))$. Otherwise, output $(y\|\pi)$.

Claim. *Assume that* $\langle \mathsf{Setup}, \mathsf{QAF} \rangle$ *satisfies classical-query hardness (Definition 6), then* $\overline{\Pi}$ *is zero-knowledge.*

Proof. We define the following hybrid experiment:

Game G_1: We modify the behavior of the simulator $\mathcal{S}_2$. It computes y and π as normal. However, if $y = \mathsf{accept}$, it aborts. Otherwise, it outputs $y\|\pi$.

For any classical-query QPT adversary $\mathcal{P}^*$, the probability of $\mathcal{P}^*$ making a query with some input x that makes the simulator $\mathcal{S}_2$ abort in G_1 is negligible by classical-query hardness of $\langle \mathsf{Setup}, \mathsf{QAF} \rangle$. Therefore the output of the simulator for $\overline{\Pi}$ is indistinguishable from its output in G_1. Now the zero-knowledge property in G_1 follows directly from the zero-knowledge property of Π, where the reduction samples $(\mathsf{pp}, \mathsf{sk}) \leftarrow \mathsf{Setup}(1^\lambda)$, computes $y \leftarrow \mathsf{QAF}(\mathsf{sk}, x)$ on its own and efficiently generates a proof π for the statement (x, y) for each query. □

Using the simulation $\overline{\mathcal{S}}$, we show that our definition is strictly stronger than the classical one below.

Claim. *Assume that* $\langle \mathsf{Setup}, \mathsf{QAF} \rangle$ *satisfies quantum-query easiness (Definition 6), then* $\overline{\Pi}$ *is not quantum simulation-sound.*

Proof. Let $\mathcal{A}$ be the QPT algorithm associated to the quantum-query easiness of $\langle \mathsf{Setup}, \mathsf{QAF} \rangle$. Define $\mathcal{P}^*$ as follows.

1. Run $\mathcal{A}$ by first making a query to $\overline{\mathcal{S}_2}$. Note that the response registers of the query have two component: one to record the output of QAF, the other to record the output of $\mathcal{S}_2$. The first component is initialized as the all-zero string $|0\rangle$, while the second component is initialized as the uniform superposition state $|+\rangle$ to remove the entanglement between the two registers so that after the query, the second response register can be discarded.

2. Continue the execution of $\mathcal{A}$ (with an input x) and obtain a triple $(\pi, (x', \pi'))$. Output two pairs (x, π) and (x', π').

By definition of $\mathcal{S}_2$ and the quantum-query easiness of $\langle \mathsf{Setup}, \mathsf{QAF} \rangle$, both two pairs output by $\mathcal{P}^*$ are valid, and furthermore x' is a false statement, showing that $\overline{\Pi}$ is not quantum simulation-sound.

Notice that here x can be an arbitrary statement. Our attack breaks the winning condition (1) of Definition 4: the claim is that the pair (x', π') will break this condition. The reason is that since x' is a random string, it cannot make the QAF accept except with negligible probability, thus the re-computation of $\mathcal{S}_2$ on x' will not output π' except with negligible probability. □

Claim. *Assume that* $\langle \mathsf{Setup}, \mathsf{QAF} \rangle$ *satisfies classical-query hardness (Definition 6), then* $\overline{\Pi}$ *is classically simulation-sound.*

Proof. The proof of this claim follows in an almost identical manner as that of Section 5.3. □

6 Constructions of Quantum SS-NIZK in the CRS Model

In this section, we show that in the common reference string model, Sahai's construction of unbounded simulation-sound NIZK [Sah01, DDO+01], when instantiating with quantum-secure one-time signature scheme (Definition 10), is also quantum simulation-sound.

The Naor commitment scheme. We first recall the bit commitment protocol of Naor [Nao90] based on pseudorandom generators, which will be used later in the construction. Let PRG be a pseudorandom generator stretching λ bits to 3λ bits. The Naor commitment procedure commits to a bit b as follows, using randomness $r \in \{0,1\}^{3\lambda}$ and $s \in \{0,1\}^{\lambda}$.

$$\mathsf{Commit}(b; (r, s)) = \begin{cases} (r, \mathsf{PRG}(s)) & \text{if } b = 0, \\ (r, \mathsf{PRG}(s) \oplus r) & \text{if } b = 1. \end{cases}$$

We note that if PRG is post-quantumly secure (against QPT adversaries with classical access to PRG) then the Naor commitment scheme is also post-quantumly computationally hiding and statistically binding.

Sahai's construction. Let PRF be a family of pseudorandom functions mapping $\{0,1\}^*$ to $\{0,1\}^{\lambda}$. Let $\mathsf{Sig} := \langle \mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verif} \rangle$ be a one-time signature scheme. Finally, let Π' be a single-theorem adaptive NIZK systems for a language $\mathcal{L}'$ described below, associated with a QPT simulator $\mathcal{S}' := (\mathcal{S}'_1, \mathcal{S}'_2)$. The construction for a simulation-sound NIZK system Π for some NP language $\mathcal{L}$ is given in Construction 3.

Construction 3.

- ***Common random string.*** *The random reference string consists of three parts* $\mathsf{crs}_1, \mathsf{crs}_2$ *and* crs_3.
  - crs_1 *is of length* $6\lambda^2$*, and breaks up into* λ *pairs* $(r_1, c_1), \ldots, (r_\lambda, c_\lambda)$.
  - crs_2 *is of length* 3λ.
  - crs_3 *is a common random string of* Π'.
- ***Prover.*** *We define the language* $\mathcal{L}'$ *to be the set of tuples* $(x, u, v, \mathsf{crs}_1, \mathsf{crs}_2)$ *such that at least one of the following three conditions hold:*

- $x \in \mathcal{L}$
- crs_1 *consists of commitments to the bits of the* λ *bit string* s*: formally, there exists* $s = s_1 \cdots s_\lambda$ *with* $s_i \in \{0,1\}$ *for all* $i \in [\![1, \lambda]\!]$*, and there exists* $a_1, \ldots, a_\lambda \in \{0,1\}^\lambda$ *such that* $(r_i, c_i) = \mathsf{Commit}(s_i; r_i, a_i)$*. Furthermore,* $u = \mathsf{PRF}(s, v)$*.*
- *There exists* $d \in \{0,1\}^\lambda$ *such that* $\mathsf{crs}_2 = \mathsf{PRG}(d)$*.*

On input a statement x*, a witness* ω *and the common random string* $\mathsf{CRS} := (\mathsf{crs}_1, \mathsf{crs}_2, \mathsf{crs}_3)$*, the prover* $\mathcal{P}$ *does the following:*

1. *Generate a key pair for the one-time signature scheme:* $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{Sig.KeyGen}(1^\lambda)$*.*
2. *Sample a uniformly random* $u \xleftarrow{\$} \{0,1\}^\lambda$*.*
3. *Using* crs_3 *as the common random string and* ω *as the witness, run the prover of* Π' *to generate a proof that* $(x, u, v, \mathsf{crs}_1, \mathsf{crs}_2) \in \mathcal{L}'$*. Denote this proof by* π'*.*
4. *Output* $\pi := (\mathsf{vk}, x, u, \pi', \mathsf{Sig.Sign}(\mathsf{sk}, (x, u, \pi')))$*.*

- ***Verifier.*** *The verification procedure, on input the instance* x*, and a proof* $\pi := (\mathsf{vk}, x, u, \pi', \sigma)$*, with respect to* $\mathsf{CRS} := (\mathsf{crs}_1, \mathsf{crs}_2, \mathsf{crs}_3)$ *does the following:*
  1. *Verify the validity of the one-time signature:* $\mathsf{Sig.Verif}(\mathsf{vk}, (x, u, \pi'), \sigma) = 1$*.*
  2. *Verify that* π' *is a valid proof that* $(x, u, \mathsf{vk}, \mathsf{crs}_1, \mathsf{crs}_2) \in \mathcal{L}'$*.*
- ***Simulator.*** *We now describe the two phases of the simulator* $\mathcal{S} := (\mathcal{S}_1, \mathcal{S}_2)$ *in Figure 1.* $\mathcal{S}_1$ *outputs a reference string* crs *along with some trapdoor information* td*.* $\mathcal{S}_2$ *takes as input this trapdoor information, the reference string, and an instance* x*, and outputs a simulated proof for* x*.*

$\underline{\mathcal{S}_1(1^\lambda)}$

$s \xleftarrow{\$} \{0,1\}^\lambda$
$r_i \xleftarrow{\$} \{0,1\}^{3\lambda}, a_i \xleftarrow{\$} \{0,1\}^\lambda$ for $i \in [\![1, \lambda]\!]$
$g_i \leftarrow \mathsf{Commit}(s_i; r_i, a_i)$ for $i \in [\![1, \lambda]\!]$
$\mathsf{crs}_1 := \{g_1, \cdots, g_\lambda\}$
$\mathsf{crs}_2 \xleftarrow{\$} \{0,1\}^{3\lambda}$
$\mathsf{crs}_3 \leftarrow \Pi'.\mathsf{Setup}(1^\lambda)$
$\mathsf{crs} := (\mathsf{crs}_1, \mathsf{crs}_2, \mathsf{crs}_3)$
$\mathsf{td} := (s, a_1, \cdots, a_\lambda)$
return $(\mathsf{crs}, \mathsf{td})$

$\underline{\mathcal{S}_2(\mathsf{crs}, \mathsf{td}, x)}$

$(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Sig.KeyGen}(1^\lambda)$
$u \leftarrow \mathsf{PRF}(s, \mathsf{vk})$
$\pi' \leftarrow \Pi'.\mathcal{S}_2'\big(\mathsf{crs}_3, (x, u, \mathsf{vk}, \mathsf{crs}_1, \mathsf{crs}_2), (s, a_1, \cdots, a_\lambda)\big)$
$\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}, x, u, \pi')$
return $(\mathsf{vk}, x, u, \pi', \sigma)$

Figure 1: The simulator of Π.

We note that quantum-secure PRFs and the Naor commitment scheme is post-quantumly-secure if quantum-secure one-way functions exist [Zha12a].

Theorem 2. *If* Π' *is a single-theorem* quantum *NIZK proof system for* $\mathcal{L}'$*,* Sig *is a quantum-secure one-time signature scheme and quantum-secure one-way functions exist, the proof system* Π *described above is an unbounded quantum simulation-sound NIZK proof system for* $\mathcal{L}$*.*

The proof of this theorem is given in Appendix E.

7 Application to Quantum-Secure Naor-Yung Construction for CCA Security

In this section, we present and prove quantum security of a simple modification of the classical Naor-Yung scheme [NY90, Sah99a]. In Section 7.1, we give some preliminaries on quantum security of encryption with classical and quantum challenge queries. Then in Section 7.2, we give our construction of building quantum chosen-ciphertext secure encryption schemes from quantum chosen-plaintext secure schemes and quantum simulation-sound NIZK proof systems.

7.1 Preliminaries: Chosen-ciphertext Security with Quantum Challenge Queries

We recall here a succinct definition of chosen-ciphertext security with quantum challenge queries introduced in [CEV22]. Their definition is defined in the real-or-random paradigm. Very informally, in the challenge phase, the adversary receives (in superposition) either encryption of his query, or encryption of random messages (by applying a random function H to the adversary's query first then encrypting). What makes it possible is that by using the compressed random oracle technique introduced by Zhandry [Zha19], the challenger can record the adversary's query in the *random* world. For decryption queries, in the real world, the challenger uses the secret key to answer the decryption queries normally. Only in the random world, by using the recorded database D, the challenger can return the original message if the query is a challenge one. For more details on the notions, we refer the reader to that paper, and refer to Section A.4 for other security notions used in the paper.

Notation. We define the following oracles.

- Let $\mathcal{O}_{\mathsf{Encrypt}(\mathsf{pk},\cdot)}$ the encryption oracle in the real world.
- Let $\mathcal{R}_{\mathsf{Encrypt}(\mathsf{pk},\cdot)}$ the encryption oracle in the random world. This oracle is augmented with a database $D = ((x_1, u_1, y_1), (x_2, u_2, y_2), \ldots, (x_q, u_q, y_q), (\perp, 0, 0), \ldots, (\perp, 0, 0))$, containing the adversary's queries in which x_i is the adversary's query, u_i is a random string in the message space and y_i is an encryption of u_i. D is kept hidden from the adversary and y_i is used to answer the queries.
- Let $\mathcal{O}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)}$ the decryption oracle in the real world.
- Let $\mathcal{R}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)}$ the decryption oracle in the random world, which is defined as follows. We define a classical procedure $\mathsf{FindImage}$ which takes as input a ciphertext $y \in \mathcal{Y}$, and a database D. Then, it looks for a tuple $(x, (u, y)) \in D$. If found, it outputs $(b = 1, w = x)$, otherwise, it outputs $(b = 0, w = 0)$. The oracle $\mathcal{R}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)}$ is defined using $\mathsf{FindImage}$ as follows. It maps the basis state $|y, z\rangle \otimes |D\rangle$ to:

$$\begin{cases} |y, z \oplus \mathsf{Decrypt}(\mathsf{sk}, y)\rangle \otimes |D\rangle & \text{if } \mathsf{FindImage}(y, D) = (0, 0), \\ |y, z \oplus w\rangle \otimes |D\rangle & \text{if } \mathsf{FindImage}(y, D) = (1, w). \end{cases}$$

We define a real-or-random oracle allowing quantum queries and the decryption oracle in the second learning phase as follows.

$$\mathcal{RR}(b) = \begin{cases} \mathcal{O}_{\mathsf{Encrypt}(\mathsf{pk},\cdot)} & \text{if } b = 1, \\ \mathcal{R}_{\mathsf{Encrypt}(\mathsf{pk},\cdot)} & \text{if } b = 0, \end{cases} \quad \mathcal{DEC}(b) = \begin{cases} \mathcal{O}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)} & \text{if } b = 1, \\ \mathcal{R}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)} & \text{if } b = 0. \end{cases}$$

Definition 7 (qIND-qCPA, qIND-qCCA1, qIND-qCCA2)**.** Let $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt})$ be a public-key encryption scheme and let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be a quantum adversary. For $qatk \in [qcpa, qcca1, qcca2]$, we define the following game, where the oracles $\mathcal{O}_1, \mathcal{O}_2$ are defined according to $qatk$:

Experiment $\mathsf{Expt}^{qind\text{-}qatk-b}_{\mathcal{E}}(\lambda, \mathcal{A})$:

1 : $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(\lambda)$
2 : $|\Phi\rangle \leftarrow \mathcal{A}_1^{\mathcal{O}_1}(\mathsf{pk})$
3 : $b' \leftarrow \mathcal{A}_2^{\mathcal{RR}(b),\mathcal{O}_2}(|\Phi\rangle)$
4 : **return** b'

$qatk$	Oracle $\mathcal{O}_1$	Oracle $\mathcal{O}_2$
$qcpa$	$\varnothing$	$\varnothing$
$qcca1$	$\mathcal{O}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)}$	$\varnothing$
$qcca2$	$\mathcal{O}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)}$	$\mathcal{DEC}(b)$

We define $\mathcal{A}$'s advantage by

$$\mathsf{Adv}^{qind\text{-}qatk}_{\mathcal{A},\mathcal{E}}(\lambda) \coloneqq \left|\Pr\left[\mathsf{Expt}^{qind\text{-}qatk-1}_{\mathcal{E}}(\lambda, \mathcal{A}) = 1\right] - \Pr\left[\mathsf{Expt}^{qind\text{-}qatk-0}_{\mathcal{E}}(\lambda, \mathcal{A}) = 1\right]\right|.$$

We say $\mathcal{E}$ is secure in the sense of qIND-qATK if $\mathcal{A}$ being QPT implies that $\mathsf{Adv}^{qind\text{-}qatk}_{\mathcal{A},\mathcal{E}}(\lambda)$ is negligible.

7.2 Construction

Construction 4. *Our construction uses the following ingredients:*

- *Let* $\mathcal{E} = \langle \mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt} \rangle$ *be an* qIND-qCPA *encryption scheme.*
- *Let* $\mathcal{E}' = \langle \mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt} \rangle$ *be an* IND-qCPA *encryption scheme.*
- *Let* iPRF *be a family of invertible pseudorandom functions.*
- *Let* $\Pi = \langle \mathsf{Setup}, \mathcal{P}, \mathcal{V}, \mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2) \rangle$ *be a quantum-simulation-sound NIZK proof system for the language* $\mathcal{L}$ *of consistent pairs of encryptions, defined formally in Equation (2).*

$$\begin{aligned} \mathcal{L} \coloneqq \{(\mathsf{pk}_0, \mathsf{pk}_1, y_0, y_1, y_2) \ : \ \exists\, (x, \mathsf{k}, r_0, r_1) : \\ y_0 = \mathcal{E}.\mathsf{Encrypt}(\mathsf{pk}_0, x; r_0) \\ \wedge\, y_1 = \mathcal{E}'.\mathsf{Encrypt}(\mathsf{pk}_1, \mathsf{k}; r_1) \wedge y_2 = \mathsf{iPRF}(\mathsf{k}, x)\}. \end{aligned} \tag{2}$$

We construct a new encryption scheme $\overline{\mathcal{E}}$ *as follows.*

$\overline{\mathsf{KeyGen}}(1^\lambda)$:

1 : $\mathsf{crs} \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$
2 : $(\mathsf{pk}_0, \mathsf{sk}_0) \xleftarrow{\$} \mathcal{E}.\mathsf{KeyGen}(1^\lambda)$
3 : $(\mathsf{pk}_1, \mathsf{sk}_1) \xleftarrow{\$} \mathcal{E}'.\mathsf{KeyGen}(1^\lambda)$
4 : $\mathsf{pk} = (\mathsf{crs}, \mathsf{pk}_0, \mathsf{pk}_1)$
5 : $\mathsf{sk} = (\mathsf{crs}, \mathsf{sk}_0, \mathsf{sk}_1)$
6 : **return** $(\mathsf{pk}, \mathsf{sk})$

$\overline{\mathsf{Encrypt}}(\mathsf{pk}, x)$:

1 : $\mathsf{k} \leftarrow \mathsf{iPRF.Setup}(1^\lambda)$
2 : $y_0 \leftarrow \mathcal{E}.\mathsf{Encrypt}(\mathsf{pk}_0, x; r_0)$
3 : $y_1 \leftarrow \mathcal{E}'.\mathsf{Encrypt}(\mathsf{pk}_1, \mathsf{k}; r_1)$
4 : $y_2 \leftarrow \mathsf{iPRF}(k_1, x)$
5 : $\pi \leftarrow \Pi.\mathcal{P}(\mathsf{crs}, (y_0, y_1, y_2), (x, \mathsf{k}, r_0, r_1))$
6 : **return** (y_0, y_1, y_2, π)

$\overline{\mathsf{Decrypt}}(\mathsf{sk}, (y_0, y_1, y_2, \pi))$:

1 : $b \leftarrow \Pi.\mathcal{V}(\mathsf{crs}, (y_0, y_1, y_2), \pi)$
2 : **if** $b = 0$ **then**
3 : **return** $\perp$
4 : **return** $\mathcal{E}.\mathsf{Decrypt}(\mathsf{sk}_0, y_0)$

Theorem 3. *The encryption $\overline{\mathcal{E}}$ described in Construction 4 above is* qIND-qCCA2 *secure.*

Proof. Let $\mathcal{A}$ be a QPT adversary. The proof proceeds by a sequence of games where G_0 is defined in which $\mathcal{A}$ can make quantum queries to $\mathcal{S}_2$ (defined in Figure 1), and the winning condition is defined as in Definition 4. For any game G_i, we denote by $\mathsf{Adv}_i(\mathcal{A})$ the advantage of $\mathcal{A}$ in G_i, that is, $\Pr\big[G_i(1^\lambda, \mathcal{A}) = 1\big]$, where the probability is taken over the random coins of G_i and $\mathcal{A}$. The changes in each game are depicted in Figure 2.

Game G_0: This is the real-world experiment. In particular, the challenge encryption oracle and the decryption oracle are implemented as follows.

$$\mathcal{RR}_{\mathsf{Encrypt}(\mathsf{pk},\cdot)}\,|x,y\rangle \mapsto |x, y \oplus \overline{\mathsf{Encrypt}}(\mathsf{pk}, x)\rangle\,,$$

and

$$\mathcal{O}_{\mathsf{Decrypt}(\mathsf{sk},\cdot)}\,|y,x\rangle \mapsto |z, x \oplus \overline{\mathsf{Decrypt}}(\mathsf{sk}, y)\rangle\,.$$

Game G_1: This is identical to G_0, except that now in the decryption oracle, instead of using sk_0, we use sk_1, combining with the fact that iPRF is invertible for the decryption.

Claim. *For any adversary $\mathcal{A}$, $|\mathsf{Adv}_1(\mathcal{A}) - \mathsf{Adv}_0(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.*

Proof. The proof of the claim follows from the correctness of encryption schemes $\mathcal{E}$, $\mathcal{E}'$, the statistical soundness of Π and the fact that iPRF is invertible. In particular, the soundness of Π guarantees that any queried ciphertext is valid (i.e., its components are encryption of the same plaintext), with overwhelming probability. □

Game G_2: This is identical to G_1, except that now in the challenge encryption oracle, we use the simulator $\mathcal{S}$ of Π to generate the proof instead of using the real prover $\mathcal{P}$.

Claim. *For any* QPT *adversary $\mathcal{A}$, $|\mathsf{Adv}_2(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.*

Proof. The indistinguishability between G_1 and G_2 follows from zero-knowledge property of Π. □

Game G_3: This is identical to G_2, except now in the challenge encryption oracle, instead of encrypting using the actual encryption algorithm $\mathcal{E}.\mathsf{Encrypt}$, we use the encryption oracle in the random world of $\mathcal{E}$. Denote this oracle as $\mathcal{R}_{\mathsf{Encrypt}(\mathsf{pk}_0,\cdot)}$.

Claim. *For any* QPT *adversary $\mathcal{A}$, $|\mathsf{Adv}_3(\mathcal{A}) - \mathsf{Adv}_2(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.*

Proof. We note that in G_2 and G_3, the secret key sk_0 is not used at all. The indistinguishability between G_2 and G_3 follows immediately from qIND-qCPA security of $\mathcal{E}$. □

We note that starting from G_3, the challenge encryption oracle can be implemented as a compressed encryption oracle (sine we are now in the random world of $\mathcal{E}$). Concretely, the challenge encryption oracle implements the following map:

$$\sum_{x,y} \alpha_{x,y}\,|x,y\rangle \mapsto \sum_{x,y}\sum_{u} \alpha_{x,y}\,|x, \mathcal{E}.\mathsf{Encrypt}(\mathsf{pk}_0, u) \| \mathcal{E}'.\mathsf{Encrypt}(\mathsf{pk}_1, \mathsf{k}) \| \mathsf{iPRF}(\mathsf{k}, x) \| \pi\rangle \otimes |D\rangle$$

where D is the database of the compressed random encryption oracle for $\mathcal{E}$. In particular, D will be in superposition of tuples (x, u, y_0) (if $D(x) \neq \perp$). Furthermore, we note that if $D(x) \neq \perp$, we can re-compute (y_1, y_2, π) and also store these values in the corresponding slot in D. The reason is that from $x \in D$, these values can be computed with the *classical* randomness used in the challenge encryption oracle of $\overline{\mathcal{E}}$.

Game G_4: This is identical to G_3, except that now instead of using sk_1 in the decryption oracle, we use sk_0 and D.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_4(\mathcal{A}) - \mathsf{Adv}_3(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. We show that if $\mathcal{A}$ can distinguish the two games G_3 and G_4 with non-negligible probability ϵ, the we can construct a QPT adversary $\mathcal{B}$ that runs $\mathcal{A}$ internally as a black-box and breaks the quantum simulation-soundness of Π with non-negligible probability. Notice that the only way $\mathcal{A}$ can distinguish G_3 and G_4 is to submit an "invalid" decryption query in which the proof π if of a false statement but the verification passes.

Formally, $\mathcal{B}$ runs $\mathcal{A}$ and randomly measure one of $\mathcal{A}$'s decryption queries to obtain a tuple $y^* = (y_0^*, y_1^*, y_2^*, \pi^*)$. If $\mathcal{A}$ makes at most q decryption queries, then with probability at least ϵ/q, y^* will be a pair of statement and proof such that the statement is a false statement but π^* passes the verification of Π. Then $\mathcal{B}$ measure its own database D to obtain another tuple $y = (y_0, y_1, y_2, \pi)$ which is supposed to be generated by the simulator of Π. By the definition of $\overline{\mathsf{Decrypt}}$[5], we have that $y \neq y^*$. Thus by outputting (y, y^*), $\mathcal{B}$ breaks the quantum simulation-soundness of Π with probability ϵ/q, which completes the proof of the claim. □

We note that from starting from this game, the secret key sk_1 is not used any more.

Game G_5: This is identical to G_4, except that now in the challenge encryption oracle, we change the encryption $\mathcal{E}'.\mathsf{Encrypt}(\mathsf{pk}_1, \mathsf{k})$ for some random key k by an encryption $\mathcal{E}'.\mathsf{Encrypt}(\mathsf{pk}_1, 0)$.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_5(\mathcal{A}) - \mathsf{Adv}_4(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The indistinguishability between G_4 and G_5 follows immediately from IND-qCPA security of $\mathcal{E}'$. Note that since here the encryption is a classical encryption of a classical random key k (which is independent from $\mathcal{A}$'s query), we only need qCPA security against classical challenge query of $\mathcal{E}'$. □

Game G_6: This is identical to G_5, except that now in the challenge encryption oracle, instead of computing y_2 as $\mathsf{iPRF}(\mathsf{k}, x)$, we compute $y_2 \leftarrow \mathsf{iPRF}(\mathsf{k}, u)$ where u is extracted from the database D (note that $u \in D(x)$). We abuse the notation and write $D(x) = u$. Furthermore, for consistency, we also allow $\mathcal{E}$'s encryption algorithm to take as input the database D.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_6(\mathcal{A}) - \mathsf{Adv}_5(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The indistinguishability between G_5 and G_6 follows immediately from (weak) quantum-pseudorandomness of iPRF. We note that here we only need weak security notion, since iPRF^{-1} is never invoked in the decryption oracle. □

Game G_7: This is identical to G_6, except that now in the challenge encryption oracle, instead of computing y_1 as an encryption of 0, we change it back to encryption of a random key k, that is $y_1 \leftarrow \mathcal{E}'.\mathsf{Encrypt}(\mathsf{pk}_1, \mathsf{k})$.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_7(\mathcal{A}) - \mathsf{Adv}_6(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The indistinguishability between G_6 and G_7 follows immediately from IND-qCPA security of $\mathcal{E}'$. □

Game G_8: This is identical to G_7, except that now in the challenge encryption oracle, we use the real prover $\mathcal{P}$ of Π to generate the proof instead of using the simulator.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_8(\mathcal{A}) - \mathsf{Adv}_7(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The indistinguishability between G_7 and G_8 follows from zero-knowledge property of Π. □

In this final game G_8, we have the challenge encryption oracle implements exactly as the one in the random-world of $\overline{\mathcal{E}}$. Overall, we complete the proof of the theorem. □

8 Acknowledgment

Ehsan Ebrahimi was supported by the Luxembourg National Research Fund under the Junior CORE project QSP (C22/IS/17272217/QSP/Ebrahimi). Céline Chevalier was supported in part by the French ANR project TCS-NISQ (ANR-22-CE47-0004), CryptiQ (ANR-18-CE39-0015). Giulio Malavolta is supported by the European Research Council through an ERC Starting Grant (Grant agreement No. 101077455, ObfusQation).

References

[ABKM22] Gorjan Alagic, Chen Bai, Jonathan Katz, and Christian Majenz. Post-quantum security of the even-mansour cipher. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part III*, volume 13277 of *LNCS*, pages 458–487. Springer, Heidelberg, May / June 2022. doi:10.1007/978-3-031-07082-2_17.

[AGRS24] Behzad Abdolmaleki, Noemi Glaeser, Sebastian Ramacher, and Daniel Slamanig. Circuit-succinct universally-composable nizks with updatable CRS. In *37th IEEE Computer Security Foundations Symposium, CSF 2024, Enschede, Netherlands, July 8-12, 2024*, pages 527–542. IEEE, 2024. doi:10.1109/CSF61375.2024.00006.

[AMRS20] Gorjan Alagic, Christian Majenz, Alexander Russell, and Fang Song. Quantum-access-secure message authentication via blind-unforgeability. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 788–817. Springer, Heidelberg, May 2020. doi:10.1007/978-3-030-45727-3_27.

[AMRS23] Gorjan Alagic, Christian Majenz, Alexander Russell, and Fang Song. Quantum-secure message authentication via blind-unforgeability. *arXiv preprint arXiv:1803.03761v4*, 2023. URL: https://arxiv.org/abs/1803.03761v4.

[ARS20] Behzad Abdolmaleki, Sebastian Ramacher, and Daniel Slamanig. Lift-and-shift: Obtaining simulation extractable subversion and updatable SNARKs generically. In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *ACM CCS 2020*, pages 1987–2005. ACM Press, November 2020. doi:10.1145/3372297.3417228.

[ATTU16] Mayuresh Vivekanand Anand, Ehsan Ebrahimi Targhi, Gelo Noel Tabia, and Dominique Unruh. Post-quantum security of the CBC, CFB, OFB, CTR, and XTS modes of operation. In Tsuyoshi Takagi, editor, *Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016*, pages 44–63. Springer, Heidelberg, 2016. doi:10.1007/978-3-319-29360-8_4.

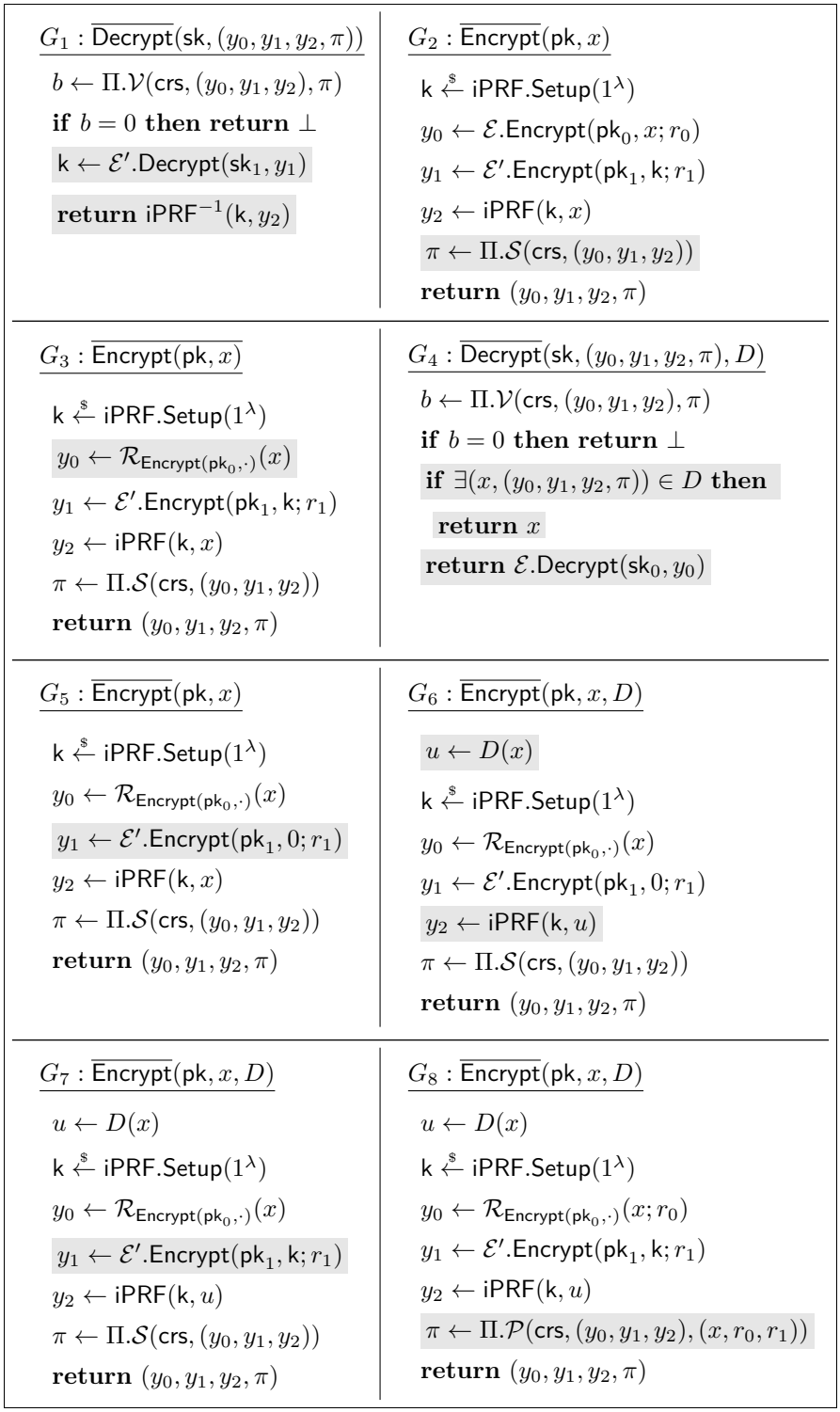

Figure 2: Description of the changes in games G_i for $i \in [\![1, 8]\!]$. In each program, the changes relative to the previous program are highlighted in light gray.

[BBC+21] Ritam Bhaumik, Xavier Bonnetain, André Chailloux, Gaëtan Leurent, María Naya-Plasencia, André Schrottenloher, and Yannick Seurin. QCB: Efficient quantum-secure authenticated encryption. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part I*, volume 13090 of *LNCS*, pages 668–698. Springer, Heidelberg, December 2021. doi:10.1007/978-3-030-92062-3_23.

[BBS04] Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In Matthew Franklin, editor, *CRYPTO 2004*, volume 3152 of *LNCS*, pages 41–55. Springer, Heidelberg, August 2004. doi:10.1007/978-3-540-28628-8_3.

[BCC+09] Mira Belenkiy, Jan Camenisch, Melissa Chase, Markulf Kohlweiss, Anna Lysyanskaya, and Hovav Shacham. Randomizable proofs and delegatable anonymous credentials. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 108–125. Springer, Heidelberg, August 2009. doi:10.1007/978-3-642-03356-8_7.

[BCC+16] Jonathan Bootle, Andrea Cerulli, Pyrros Chaidos, Essam Ghadafi, and Jens Groth. Foundations of fully dynamic group signatures. In Mark Manulis, Ahmad-Reza Sadeghi, and Steve Schneider, editors, *ACNS 16*, volume 9696 of *LNCS*, pages 117–136. Springer, Heidelberg, June 2016. doi:10.1007/978-3-319-39555-5_7.

[BCG+18] Jonathan Bootle, Andrea Cerulli, Jens Groth, Sune K. Jakobsen, and Mary Maller. Arya: Nearly linear-time zero-knowledge proofs for correct program execution. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part I*, volume 11272 of *LNCS*, pages 595–626. Springer, Heidelberg, December 2018. doi:10.1007/978-3-030-03326-2_20.

[BCM+18] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh V. Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In Mikkel Thorup, editor, *59th FOCS*, pages 320–331. IEEE Computer Society Press, October 2018. doi:10.1109/FOCS.2018.00038.

[BFM88] Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications (extended abstract). In *20th ACM STOC*, pages 103–112. ACM Press, May 1988. doi:10.1145/62212.62222.

[BG90] Mihir Bellare and Shafi Goldwasser. New paradigms for digital signatures and message authentication based on non-interactive zero knowledge proofs. In Gilles Brassard, editor, *CRYPTO'89*, volume 435 of *LNCS*, pages 194–211. Springer, Heidelberg, August 1990. doi:10.1007/0-387-34805-0_19.

[BKW17] Dan Boneh, Sam Kim, and David J. Wu. Constrained keys for invertible pseudorandom functions. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part I*, volume 10677 of *LNCS*, pages 237–263. Springer, Heidelberg, November 2017. doi:10.1007/978-3-319-70500-2_9.

[BP15] Nir Bitansky and Omer Paneth. ZAPs and non-interactive witness indistinguishability from indistinguishability obfuscation. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part II*, volume 9015 of *LNCS*, pages 401–427. Springer, Heidelberg, March 2015. doi:10.1007/978-3-662-46497-7_16.

[BZ13a] Dan Boneh and Mark Zhandry. Quantum-secure message authentication codes. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 592–608. Springer, Heidelberg, May 2013. doi:10.1007/978-3-642-38348-9_35.

[BZ13b] Dan Boneh and Mark Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 361–379. Springer, Heidelberg, August 2013. doi:10.1007/978-3-642-40084-1_21.

[Can01] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *42nd FOCS*, pages 136–145. IEEE Computer Society Press, October 2001. doi:10.1109/SFCS.2001.959888.

[CCS09] Jan Camenisch, Nishanth Chandran, and Victor Shoup. A public key encryption scheme secure against key dependent chosen plaintext and adaptive chosen ciphertext attacks. In Antoine Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 351–368. Springer, Heidelberg, April 2009. doi:10.1007/978-3-642-01001-9_20.

[CETU21] Tore Vincent Carstens, Ehsan Ebrahimi, Gelo Noel Tabia, and Dominique Unruh. Relationships between quantum IND-CPA notions. In Kobbi Nissim and Brent Waters, editors, *TCC 2021, Part I*, volume 13042 of *LNCS*, pages 240–272. Springer, Heidelberg, November 2021. doi:10.1007/978-3-030-90459-3_9.

[CEV22] Céline Chevalier, Ehsan Ebrahimi, and Quoc Huy Vu. On security notions for encryption in a quantum world. In Takanori Isobe and Santanu Sarkar, editors, *Progress in Cryptology - INDOCRYPT 2022 - 23rd International Conference on Cryptology in India, Kolkata, India, December 11-14, 2022, Proceedings*, volume 13774 of *Lecture Notes in Computer Science*, pages 592–613. Springer, 2022. doi:10.1007/978-3-031-22912-1_26.

[CKL+16] Jan Camenisch, Stephan Krenn, Anja Lehmann, Gert Læssøe Mikkelsen, Gregory Neven, and Michael Østergaard Pedersen. Formal treatment of privacy-enhancing credential systems. In Orr Dunkelman and Liam Keliher, editors, *SAC 2015*, volume 9566 of *LNCS*, pages 3–24. Springer, Heidelberg, August 2016. doi:10.1007/978-3-319-31301-6_1.

[CL01] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In Birgit Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 93–118. Springer, Heidelberg, May 2001. doi:10.1007/3-540-44987-6_7.

[Cv91] David Chaum and Eugène van Heyst. Group signatures. In Donald W. Davies, editor, *EUROCRYPT'91*, volume 547 of *LNCS*, pages 257–265. Springer, Heidelberg, April 1991. doi:10.1007/3-540-46416-6_22.

[DDO+01] Alfredo De Santis, Giovanni Di Crescenzo, Rafail Ostrovsky, Giuseppe Persiano, and Amit Sahai. Robust non-interactive zero knowledge. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 566–598. Springer, Heidelberg, August 2001. doi:10.1007/3-540-44647-8_33.

[DFNS14] Ivan Damgård, Jakob Funder, Jesper Buus Nielsen, and Louis Salvail. Superposition attacks on cryptographic protocols. In Carles Padró, editor, *ICITS 13*, volume 8317 of *LNCS*, pages 142–161. Springer, Heidelberg, 2014. doi:10.1007/978-3-319-04268-8_9.

[DG23] Quang Dao and Paul Grubbs. Spartan and bulletproofs are simulation-extractable (for free!). In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part II*, volume 14005 of *LNCS*, pages 531–562. Springer, Heidelberg, April 2023. doi:10.1007/978-3-031-30617-4_18.

[DP92] Alfredo De Santis and Giuseppe Persiano. Zero-knowledge proofs of knowledge without interaction (extended abstract). In *33rd FOCS*, pages 427–436. IEEE Computer Society Press, October 1992. doi:10.1109/SFCS.1992.267809.

[EvW22] Ehsan Ebrahimi and Jeroen van Wier. Post-quantum plaintext-awareness. In Jung Hee Cheon and Thomas Johansson, editors, *Post-Quantum Cryptography - 13th International Workshop, PQCrypto 2022, Virtual Event, September 28-30, 2022, Proceedings*, volume 13512 of *Lecture Notes in Computer Science*, pages 260–285. Springer, 2022. doi:10.1007/978-3-031-17234-2_13.

[FHS19] Georg Fuchsbauer, Christian Hanser, and Daniel Slamanig. Structure-preserving signatures on equivalence classes and constant-size anonymous credentials. *Journal of Cryptology*, 32(2):498–546, April 2019. doi:10.1007/s00145-018-9281-4.

[FLS90] Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *31st FOCS*, pages 308–317. IEEE Computer Society Press, October 1990. doi:10.1109/FSCS.1990.89549.

[FS87] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194. Springer, Heidelberg, August 1987. doi:10.1007/3-540-47721-7_12.

[GGP10] Rosario Gennaro, Craig Gentry, and Bryan Parno. Non-interactive verifiable computing: Outsourcing computation to untrusted workers. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 465–482. Springer, Heidelberg, August 2010. doi:10.1007/978-3-642-14623-7_25.

[GGPR13] Rosario Gennaro, Craig Gentry, Bryan Parno, and Mariana Raykova. Quadratic span programs and succinct NIZKs without PCPs. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 626–645. Springer, Heidelberg, May 2013. doi:10.1007/978-3-642-38348-9_37.

[GMR85] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof-systems (extended abstract). In *17th ACM STOC*, pages 291–304. ACM Press, May 1985. doi:10.1145/22145.22178.

[GO93] Shafi Goldwasser and Rafail Ostrovsky. Invariant signatures and non-interactive zero-knowledge proofs are equivalent (extended abstract). In Ernest F. Brickell, editor, *CRYPTO'92*, volume 740 of *LNCS*, pages 228–245. Springer, Heidelberg, August 1993. doi:10.1007/3-540-48071-4_16.

[Gro06] Jens Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 444–459. Springer, Heidelberg, December 2006. doi:10.1007/11935230_29.

[HU19] Dennis Hofheinz and Bogdan Ursu. Dual-mode NIZKs from obfuscation. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part I*, volume 11921 of *LNCS*, pages 311–341. Springer, Heidelberg, December 2019. doi:10.1007/978-3-030-34578-5_12.

[KLLN16] Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part II*, volume 9815 of *LNCS*, pages 207–237. Springer, Heidelberg, August 2016. doi:10.1007/978-3-662-53008-5_8.

[KLS18] Eike Kiltz, Vadim Lyubashevsky, and Christian Schaffner. A concrete treatment of Fiat-Shamir signatures in the quantum random-oracle model. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part III*, volume 10822 of *LNCS*, pages 552–586. Springer, Heidelberg, April / May 2018. doi:10.1007/978-3-319-78372-7_18.

[KM10] Hidenori Kuwakado and Masakatu Morii. Quantum distinguisher between the 3-round feistel cipher and the random permutation. In *IEEE International Symposium on Information Theory, ISIT 2010, June 13-18, 2010, Austin, Texas, USA, Proceedings*, pages 2682–2685. IEEE, 2010. doi:10.1109/ISIT.2010.5513654.

[KM12] Hidenori Kuwakado and Masakatu Morii. Security on the quantum-type even-mansour cipher. In *Proceedings of the International Symposium on Information Theory and its Applications, ISITA 2012, Honolulu, HI, USA, October 28-31, 2012*, pages 312–316. IEEE, 2012. URL: https://ieeexplore.ieee.org/document/6400943/.

[KZM+15] Ahmed Kosba, Zhichao Zhao, Andrew Miller, Yi Qian, Hubert Chan, Charalampos Papamanthou, Rafael Pass, abhi shelat, and Elaine Shi. C∅c∅: A framework for building composable zero-knowledge proofs. Cryptology ePrint Archive, Paper 2015/1093, 2015. https://eprint.iacr.org/2015/1093. URL: https://eprint.iacr.org/2015/1093.

[LMQW22] Alex Lombardi, Ethan Mook, Willy Quach, and Daniel Wichs. Post-quantum insecurity from LWE. In Eike Kiltz and Vinod Vaikuntanathan, editors, *TCC 2022, Part I*, volume 13747 of *LNCS*, pages 3–32. Springer, Heidelberg, November 2022. doi:10.1007/978-3-031-22318-1_1.

[LNPT20] Benoît Libert, Khoa Nguyen, Alain Passelègue, and Radu Titiu. Simulation-sound arguments for LWE and applications to KDM-CCA2 security. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part I*, volume 12491 of *LNCS*, pages 128–158. Springer, Heidelberg, December 2020. doi:10.1007/978-3-030-64837-4_5.

[Nao90] Moni Naor. Bit commitment using pseudo-randomness. In Gilles Brassard, editor, *CRYPTO'89*, volume 435 of *LNCS*, pages 128–136. Springer, Heidelberg, August 1990. doi:10.1007/0-387-34805-0_13.

[NC11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, USA, 10th edition, 2011.

[NS09] Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 18–35. Springer, Heidelberg, August 2009. doi:10.1007/978-3-642-03356-8_2.

[NY90] Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *22nd ACM STOC*, pages 427–437. ACM Press, May 1990. doi:10.1145/100216.100273.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005. doi:10.1145/1060590.1060603.

[RS19] Roy Radian and Or Sattath. Semi-quantum money. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, AFT '19, page 132–146, New York, NY, USA, 2019. Association for Computing Machinery. doi:10.1145/3318041.3355462.

[Sah99a] Amit Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *40th FOCS*, pages 543–553. IEEE Computer Society Press, October 1999. doi:10.1109/SFFCS.1999.814628.

[Sah99b] Amit Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *FOCS 1999*, pages 543–553. IEEE Computer Society, 1999. doi:10.1109/SFFCS.1999.814628.

[Sah01] Amit Sahai. Simulation-sound non-interactive zero knowledge, 2001.

[Sho94] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th FOCS*, pages 124–134. IEEE Computer Society Press, November 1994. doi:10.1109/SFCS.1994.365700.

[Unr12] Dominique Unruh. Quantum proofs of knowledge. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 135–152. Springer, Heidelberg, April 2012. doi:10.1007/978-3-642-29011-4_10.

[Unr15] Dominique Unruh. Non-interactive zero-knowledge proofs in the quantum random oracle model. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 755–784. Springer, Heidelberg, April 2015. doi:10.1007/978-3-662-46803-6_25.

[VV19] Umesh Vazirani and Thomas Vidick. Fully device independent quantum key distribution. *Communications of the ACM*, 62(4):133–133, 2019. doi:10.1145/3310974.

[Wat06] John Watrous. Zero-knowledge against quantum attacks. In Jon M. Kleinberg, editor, *38th ACM STOC*, pages 296–305. ACM Press, May 2006. doi:10.1145/1132516.1132560.

[WZ82] William K Wootters and Wojciech H Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982. URL: https://doi.org/10.1038/299802a0.

[Zha12a] Mark Zhandry. How to construct quantum random functions. In *53rd FOCS*, pages 679–687. IEEE Computer Society Press, October 2012. doi:10.1109/FOCS.2012.37.

[Zha12b] Mark Zhandry. Secure identity-based encryption in the quantum random oracle model. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 758–775. Springer, Heidelberg, August 2012. doi:10.1007/978-3-642-32009-5_44.

[Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 239–268. Springer, Heidelberg, August 2019. doi:10.1007/978-3-030-26951-7_9.

Supplementary Material

A Preliminaries

A.1 Quantum Information

We use $\mathcal{H}$ to denote an arbitrary finite-dimensional Hilbert space, and use indices to differentiate between distinct spaces. We let $|\phi\rangle$ denote an arbitrary pure quantum state, let $|x\rangle$ denote an element of the standard (computational) basis. A mixed state will be denoted by lowercase Greek letters, e.g., ρ. We let $|+\rangle$ denote the uniform superposition, that is $|+\rangle \coloneqq \sum_x |x\rangle$.

A pure state $|\phi\rangle$ can be manipulated by performing a unitary transformation U to the state $|\phi\rangle$, which we denote $U|\phi\rangle$. The identity on a n-bit quantum system is denoted $\mathcal{I}_n$. Given two quantum systems A, B, with corresponding Hilbert spaces $\mathcal{H}_A, \mathcal{H}_B$, let $|\phi\rangle = |\phi_0, \phi_1\rangle$ be a state of the joint system. We write $U^A|\phi\rangle$ to denote that we act with U on register A, and with identity $\mathcal{I}$ on register B, and we write U^{AB} to denote that we act with U on both registers A, B simultaneously, that is $U^{AB} = U^A \otimes U^B$.

Partial Measurement. Given two quantum systems A, B, with corresponding Hilbert spaces $\mathcal{H}_A, \mathcal{H}_B$, let ρ_{AB} be the density matrix of the joint system. We write $\mathrm{Tr}_B(\rho_{AB})$ for the state obtained by tracing out system A.

Quantum Computations. Let Q be a n-bit quantum system over $\mathbb{Z}_q$ for some integer q. The Quantum Fourier Transform (QFT) performs the following operation efficiently:

$$\mathsf{QFT}|x\rangle \coloneqq \frac{1}{\sqrt{q^n}} \sum_{y\in\{0,1\}^n} \omega_q^{x\cdot y}|y\rangle\,,$$

where $\omega_q \coloneqq \exp(\frac{2\pi i}{q})$, and $x \cdot y$ denotes the dot product. In this paper, we usually consider $q = 2$, so that $\omega_q = (-1)$.

Given a function $f : \mathcal{X} \to \mathcal{Y}$, we model a quantum-accessible oracle $\mathcal{O}$ for f as a unitary transformation $\mathcal{O}_f$ acting on three registers X, Y, Z with the property that $\mathcal{O}_f : |x, y, 0\rangle \mapsto |x, y \oplus f(x), 0\rangle$, where $\oplus$ is some involutive group operation (so-called quantum query model). Given an algorithm $\mathcal{A}$, we sometimes write $y \leftarrow \mathcal{A}^{\mathcal{O}_1,\mathcal{O}_2,\cdots}(x)$ for the event that a quantum adversary $\mathcal{A}$ takes x as input, makes quantum queries to $\mathcal{O}_1, \mathcal{O}_2, \ldots$, and finally outputs y.

A.2 Pseudorandom Functions

Definition 8 (Pseudorandom Function)**.** A pseudorandom function (PRF) is a tuple of PPT algorithms $\langle \mathsf{KeyGen}, \mathsf{PRF}\rangle$ such that:

- $\mathsf{k} \leftarrow \mathsf{KeyGen}(1^\lambda)$. The key generation algorithm KeyGen takes a security parameter λ as input and outputs a random key $\mathsf{k} \in \mathcal{K}$.
- $y \leftarrow \mathsf{PRF}(\mathsf{k}, x)$. The PRF algorithm takes as input a key $\mathsf{k} \in \mathcal{K}$ and an input $x \in$, and deterministically outputs a classical string $y \in \mathcal{Y}$.

We require the following properties.

- **Pseudorandomness.** For every QPT adversary $\mathcal{A}$, and every $\lambda \in \mathbb{N}$, the following holds:

$$\left| \Pr_{\mathsf{k} \xleftarrow{\$} \mathsf{KeyGen}(1^\lambda)} \left[\mathcal{A}^{|\mathsf{PRF}(\mathsf{k},\cdot)\rangle}(1^\lambda) = 1 \right] - \Pr \left[\mathcal{A}^{|\mathcal{O}_H\rangle}(1^\lambda) = 1 \right] \right| \leq \frac{1}{2} + \mathsf{negl}(\lambda),$$

  where the probability is taken over the randomness of KeyGen, and $\mathcal{O}_H$ is a random function from $\mathcal{F} \coloneqq \{ F \mid F :\to \mathcal{Y} \}$.

We will optionally require the following property:

- **One-wise Independence.** Let $= \{0,1\}^{n(\lambda)}$ and $\mathcal{Y} = \{0,1\}^{m(\lambda)}$. We say $\langle \mathsf{KeyGen}, \mathsf{PRF} \rangle$ is 1-wise independent if for any input $x \in$ and any $y \in \mathcal{Y}$:

$$\Pr[\, \mathsf{PRF}(\mathsf{k}, x) = y \,] = \frac{1}{2^m},$$

  where the probability is over $\mathsf{k} \leftarrow \mathsf{KeyGen}(1^\lambda)$.

Definition 9 (Invertible Pseudorandom Functions)**.** An invertible pseudorandom function (IPF) with key-space $\mathcal{K}$, domain , and range $\mathcal{Y}$ consists of two functions $\mathsf{iPRF} : \mathcal{K} \times \to \mathcal{Y}$ and $\mathsf{iPRF}^{-1} : \mathcal{K} \times \mathcal{Y} \to \cup\{\bot\}$. An IPF can also include a setup algorithm $\mathsf{iPRF.Setup}(1^\lambda)$ that on input the security parameter λ, outputs a key $\mathsf{k} \in \mathcal{K}$. The functions iPRF and iPRF^{-1} satisfy the following properties:

- Both iPRF and iPRF^{-1} can be computed by deterministic polynomial-time algorithms.
- For all security parameters λ and all keys k output by $\mathsf{iPRF.Setup}(1^\lambda)$, the function $\mathsf{iPRF}(\mathsf{k}, \cdot)$ is an injective function from to $\mathcal{Y}$. Moreover, the function $\mathsf{iPRF}^{-1}(\mathsf{k}, \cdot)$ is the (generalized) inverse of $\mathsf{iPRF}(\mathsf{k}, \cdot)$.
- **(Weak) Quantum Pseudorandomness.** An IPF $\mathsf{iPRF} : \mathcal{K} \times \to \mathcal{Y}$ is secure if for all QPT adversaries $\mathcal{A}$,

$$\left| \Pr_{\mathsf{k} \xleftarrow{\$} \mathsf{iPRF.Setup}(1^\lambda)} \left[\mathcal{A}^{|\mathsf{iPRF}(\mathsf{k},\cdot)\rangle}(1^\lambda) = 1 \right] - \Pr_{R \xleftarrow{\$} \mathsf{InjFuncs}[,\mathcal{Y}]} \left[\mathcal{A}^{|R(\cdot)\rangle}(1^\lambda) = 1 \right] \right| \leq \mathsf{negl}(\lambda),$$

  where $\mathsf{InjFuncs}[, \mathcal{Y}]$ is the set of all *injective* functions from to $\mathcal{Y}$.

A quantum-secure construction of invertible pseudorandom function is given in Appendix C.

A.3 One-time Signatures

A signature scheme consists of three polynomial time classical algorithms $\mathsf{Sig} = \langle \mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verif} \rangle$. KeyGen is a randomized procedure that takes as input the security parameter and produces a secret key and public key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$. Sign takes as input the secret key and a message m, and produces a signature $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$. Finally, Verif takes as input the public key, a message m, and a supposed signature σ on m, and either accepts or rejects.

A signature scheme is correct if Verif accepts signatures outputted by Sign such that

$$\Pr \left[\begin{array}{c|c} \mathsf{Verif}(\mathsf{pk}, m, \sigma) = 1 & \begin{array}{c} (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ \sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, m) \end{array} \end{array} \right] \geq 1 - \mathsf{negl}(\lambda).$$

For security, we will for simplicity only consider one-time signature schemes where the adversary only receives a single superposition of messages. Furthermore, for simplicity we assume that the signing function is a deterministic function of the secret key and message; this can be made without loss of generality by using a pseudorandom function to generate the randomness.

Boneh-Zhandry security. Boneh and Zhandry [BZ13b] give the following definition of security for signatures in the presence of quantum adversaries. Let $\mathcal{A}$ be a quantum adversary, and consider the following experiment between $\mathcal{A}$ and a challenger:

- The challenger runs $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$, and gives pk to $\mathcal{A}$.
- $\mathcal{A}$ makes a quantum superpositions query to the function $\mathsf{Sign}(\mathsf{sk}, \cdot)$ as $|m, u\rangle \mapsto |m, u \oplus \mathsf{Sign}(\mathsf{sk}, m)\rangle$.
- $\mathcal{A}$ outputs two classical message/signature pairs $((m_0, \sigma_0), (m_1, \sigma_1))$.
- The challenger accepts and outputs 1 if and only if (1) $m_0 \neq m_1$, and (2) $\mathsf{Verif}(\mathsf{pk}, m_b, \sigma_b)$ for both $b \in \{0, 1\}$. Denote this output by $\mathsf{W\text{-}BZ\text{-}Exp}_\lambda(\mathcal{A})$.

Definition 10 (Boneh-Zhandry [BZ13b])**.** A signature scheme is one-time weakly BZ-secure if, for any quantum polynomial time adversary $\mathcal{A}$, $\mathsf{W\text{-}BZ\text{-}Exp}_\lambda(\mathcal{A})$ is negligible.

A.4 Public-key Encryption

A public-key cryptosystem $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt})$ consists of three PPT algorithms.

- $\mathsf{KeyGen}(\lambda)$ is a probabilistic key generation algorithm which takes as input the security parameter λ and outputs a pair $(\mathsf{pk}, \mathsf{sk})$ of matching public and secret keys.
- $\mathsf{Encrypt}(\mathsf{pk}, x; r)$ is a probabilistic encryption algorithm which takes as input a public key pk, a plaintext $x \in$ (where is some fixed message space), samples a random coin on each invocation $r \in \mathcal{R}$ (where $\mathcal{R}$ is the randomness space), and outputs a ciphertext y. We sometimes omit the random coin and write $\mathsf{Encrypt}(\mathsf{pk}, x)$.
- $\mathsf{Decrypt}(\mathsf{sk}, y)$ is a deterministic decryption algorithm which takes as input a secret key sk and a ciphertext y, and outputs a message $x \in \cup\{\perp\}$, where $\perp$ is a distinguished symbol indicating decryption failure.

Security Definitions. Similar to the symmetric setting, we first give a Real-or-Random security definition for public-key encryption in the classical setting, then Boneh-Zhandry's definitions. For any subset D of the ciphertext space $\mathcal{C}$, we define the "punctured" decryption oracle $\widetilde{\mathsf{Decrypt}}^{D}(\mathsf{sk}, y)$ which returns $\mathsf{Decrypt}(\mathsf{sk}, y)$ if $y \notin D$, else it returns $\perp$.

Definition 11 (Real-or-Random $\mathsf{IND\text{-}CPA}, \mathsf{IND\text{-}CCA1}, \mathsf{IND\text{-}CCA2}$)**.**
Let $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt})$ be an public-key encryption scheme and let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be a classical adversary. Let $\mathcal{F}$ is the family of all functions over . For $atk \in [cpa, cca1, cca2]$, we define the following game, where the oracles $\mathcal{O}_1, \mathcal{O}_2$ are defined according to atk:

Experiment $\mathsf{Expt}_{\mathcal{E}}^{ind\text{-}atk-b}(\lambda, \mathcal{A})$:

1 : $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(\lambda)$
2 : $(x, \mathsf{state}) \leftarrow \mathcal{A}_1^{\mathcal{O}_{\mathsf{Encrypt}(\mathsf{pk}, \cdot)}, \mathcal{O}_1}(\lambda)$
3 : $h \xleftarrow{\$} \mathcal{F}$
4 : $y^* \leftarrow \mathcal{O}_{\mathsf{Encrypt}(\mathsf{pk}, \cdot)}(h^{1-b}(x))$
5 : $b' \leftarrow \mathcal{A}_2^{\mathcal{O}_{\mathsf{Encrypt}(\mathsf{pk}, \cdot)}, \mathcal{O}_2}(y^*, \mathsf{state})$
6 : **return** b'

atk	Oracle $\mathcal{O}_1$	Oracle $\mathcal{O}_2$
cpa	$\varnothing$	$\varnothing$
$cca1$	$\mathsf{Decrypt}(\mathsf{sk}, \cdot)$	$\varnothing$
$cca2$	$\mathsf{Decrypt}(\mathsf{sk}, \cdot)$	$\mathsf{Decrypt}^*(\mathsf{sk}, \cdot)$

Here, $\mathsf{Decrypt}^*(\mathsf{sk}, y)$ returns x if $y = y^*$, otherwise it decrypts normally.
We define $\mathcal{A}$'s advantage by

$$\mathsf{Adv}^{ind\text{-}atk}_{\mathcal{A},\mathcal{E}}(\lambda) \coloneqq \left|\Pr\left[\mathsf{Expt}^{ind\text{-}atk-1}_{\mathcal{E}}(\lambda,\mathcal{A})=1\right] - \Pr\left[\mathsf{Expt}^{ind\text{-}atk-0}_{\mathcal{E}}(\lambda,\mathcal{A})=1\right]\right|.$$

We say $\mathcal{E}$ is secure in the sense of IND-ATK if $\mathcal{A}$ being PPT implies that $\mathsf{Adv}^{ind\text{-}atk}_{\mathcal{A},\mathcal{E}}(\lambda)$ is negligible.

Definition 12 (IND-qCPA, IND-qCCA1, IND-qCCA2 [BZ13b])**.**
Let $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt})$ be an public-key encryption scheme and let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be a quantum adversary. For $qatk \in [qcpa, qcca1, qcca2]$, we define the following game, where the oracles $\mathcal{O}_1, \mathcal{O}_2$ are defined according to $qatk$:

Experiment $\mathsf{Expt}^{ind\text{-}qatk-b}_{\mathcal{E}}(\lambda,\mathcal{A})$:

1 : $(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{KeyGen}(\lambda)$
2 : $|x_0,x_1\rangle|\phi\rangle \leftarrow \mathcal{A}_1^{\mathcal{O}_1}(\mathsf{pk})$
3 : **if** $|x_0| \neq |x_1|$ **then return** 0
4 : $y^* \leftarrow \mathsf{Encrypt}(\mathsf{pk}, x_b)$
5 : $b' \leftarrow \mathcal{A}_2^{\mathcal{O}_2}(|y^*\rangle|\phi\rangle)$
6 : **return** b'

$qatk$	Oracle $\mathcal{O}_1$	Oracle $\mathcal{O}_2$
$qcpa$	$\varnothing$	$\varnothing$
$qcca1$	$\mathsf{Decrypt}(\mathsf{sk},\cdot)$	$\varnothing$
$qcca2$	$\mathsf{Decrypt}(\mathsf{sk},\cdot)$	$\widetilde{\mathsf{Decrypt}}^{D}(\mathsf{sk},\cdot)$ with $D=\{y^*\}$

We define $\mathcal{A}$'s advantage by

$$\mathsf{Adv}^{ind\text{-}qatk}_{\mathcal{A},\mathcal{E}}(\lambda) \coloneqq \left|\Pr\left[\mathsf{Expt}^{ind\text{-}qatk-1}_{\mathcal{E}}(\lambda,\mathcal{A})=1\right] - \Pr\left[\mathsf{Expt}^{ind\text{-}qatk-0}_{\mathcal{E}}(\lambda,\mathcal{A})=1\right]\right|.$$

We say $\mathcal{E}$ is secure in the sense of IND-qATK if $\mathcal{A}$ being QPT implies that $\mathsf{Adv}^{ind\text{-}qatk}_{\mathcal{A},\mathcal{E}}(\lambda)$ is negligible.

B The Quantum Certification Protocol [BCM+18]

The protocol relies on a post-quantum secure trapdoor claw-free (TCF) family of functions with adaptive hard-core bit property $f_0, f_1 : \{0,1\}^n \to \{0,1\}^m$. A TCF pair is a pair of functions which are injective, with the same image, and satisfy the following property. With knowledge of a secret trapdoor it is possible to efficiently (classically) compute the two pre-images x_0 and x_1 of a given y ($f_0(x_0) = f_1(x_1) = y$), but without the trapdoor, there is no efficient quantum algorithm that can compute such a triple (x_0, x_1, y), referred to as a claw, for any y. The adaptive hardcore bit property states that it is also hard to hold both a single pre-image x_b, as well as a string $d \in \{0,1\}^b \setminus 0^b$ and a bit c such that $c = d \cdot (x_0 \oplus x_1)$.

We note that while the quantum device cannot compute a claw or break the adaptive hard-core bit property, nevertheless it can simultaneously hold an image y as well as a superposition $\frac{1}{\sqrt{2}}(|0,x_0\rangle + |1,x_1\rangle)$ over the two pre-images of y, simply by evaluating f on a uniform superposition over all inputs and measuring the image register y. Then by either measuring the state in the computational basis or the Hadamard basis, the device can obtain either a random pre-image x_b of y, or a pair (c,d) such that $c = d \cdot (x_0 \oplus x_1)$.

A high-level description of the [BCM+18] protocol is given below.

Construction 5.

1. *The verifier generates a TCF pair, along with a trapdoor, and sends just the function pair to the prover.*

2. *The prover returns an image y of the TCF pair.*

3. *The verifier challenges the prover by randomly asking for either a pre-image of y, or a bit c and an n-bit string d such that $d \cdot (x_0 \oplus x_1) = c$.*

4. *The prover measures in the computational or Hadamard basis to return the requested output and the verifier checks the validity by using the trapdoor to compute the two pre-images x_0, x_1 of y.*

Lemma 5 (Lemma 2, restated)**.** *Under the LWE assumption, there exists a 4-message interactive proof of quantumness protocol satisfying: (1) public-coin second verifier message and (2) semi-quantum soundness.*

Proof. The protocol we use in the proof is the n-fold parallel repetition of Construction 5. Construction 5 has soundness error 1/2, and parallel repetition amplifies the soundness of this protocol, which has been shown in [RS19].

By inspecting the [BCM+18] protocol, we note that the verifier's second message is public coin. What remains is to argue that the protocol is also semi-quantum sound: in the [BCM+18] protocol, the crucial point is that the prover can compute a quantum state to obtain its first message p_1 on its own, and later this quantum state will be either measured in the computational basis or the Hadamard basis to answer the challenge from the verifier. In particular, p_1 is an image of the TCF function obtained by running the TCF function in superposition over all input and then measuring in the computational basis. Now consider the security game of semi-quantum soundness: the prover can only compute p_1 via sending a *classical* query to $\mathcal{V}_{\mathsf{semi}}$: that is, the prover sends a classical input x and receives back the image p_1 of x. Since this is a classical query, no efficient quantum prover can give a valid answer in the Hadamard basis for a fixed pair (v_1, p_1). Formally, we can construct a simulator $\mathcal{S}$ which simulates the malicious semi-quantum prover $\mathcal{P}^*$ and plays the role of the prover in the [BCM+18] protocol. $\mathcal{S}$ makes a copy of (v_1, x, p_1) (where x is the input of $\mathcal{P}^*$'s query) and later sends (v_1, p_1, v_2) to $\mathcal{P}^*$. Finally $\mathcal{S}$ outputs whatever $\mathcal{P}^*$ outputs. One can see that now if $\mathcal{P}^*$ breaks the semi-quantum soundness, $\mathcal{S}$ breaks the "adaptive hard-core bit" property of the [BCM+18] protocol. □

C Quantum-Secure Invertible Pseudorandom Functions

We show a construction for invertible pseudorandom functions from standard pseudorandom functions. The construction is the one given in [BKW17]. We first recall the construction and show that if the underlying pseudorandom functions are quantum-secure, then this construction is also weakly quantum-secure.

Construction 6. *Let $\mathsf{PRF}_1 : \mathcal{K}_1 \times \to \mathcal{Y}$ and $\mathsf{PRF}_2 : \mathcal{K}_2 \times \mathcal{Y} \to$ be two pseudorandom functions. Define the following invertible iPRF on domain using a key $\mathsf{k} := (\mathsf{k}_1, \mathsf{k}_2) \in \mathcal{K}_1 \times \mathcal{K}_2$:*

Theorem 4. *Assume that $\mathsf{PRF}_1, \mathsf{PRF}_2$ are quantum-secure (according to Definition 8), then iPRF in Construction 6 is weakly quantum-secure (according to Definition 9).*

Proof. We note that in the weak pseudorandom security, the adversary has only quantum access to an evaluation oracle iPRF, and not an inversion oracle iPRF^{-1}. The proof of the theorem follows from the standard hybrid argument, where we first replace PRF_1 with a truly random function, and then we replace PRF_2 with another truly random function. We omit the details. □

$\underline{\mathsf{iPRF}((\mathsf{k}_1,\mathsf{k}_2),x)}$	$\underline{\mathsf{iPRF}^{-1}((\mathsf{k}_1,\mathsf{k}_2),(y_1,y_2))}$
$y_1 \leftarrow \mathsf{PRF}_1(\mathsf{k}_1,x)$	$x \leftarrow \mathsf{PRF}_2(\mathsf{k}_2,y_1) \oplus y_2$
$y_2 \leftarrow \mathsf{PRF}_2(\mathsf{k}_2,y_1) \oplus x$	**if** $y_1 \neq \mathsf{PRF}_1(\mathsf{k}_1,x)$
return (y_1,y_2)	**return** $\perp$
	else return x

Figure 3: [BKW17]'s invertible pseudorandom functions construction.

D Quantumly Computational Zero-Knowledge Proof Systems

In this section, we briefly show that some known non-interactive zero-knowledge proof systems in literature satisfy the stronger notion of *quantumly computational zero-knowledge*. The construction we consider is the one given in [BP15]. At a high-level overview, their construction is a concrete instantiation of the Goldwasser-Ostrovsky transformation [GO93] which gives NIZKs from *invariant signatures*.

Informally, invariant signatures are digital signatures where all valid signatures of any message are either identical, or share a common property. Concretely, we say that a signature scheme is invariant if there is some efficiently computable property P of signatures such that for any message m^* and any verification key vk there is a unique value $P_{\mathsf{vk}}(m^*)$ such that $P(\sigma) = P_{\mathsf{vk}}(m^*)$ for any valid signature σ with respect to vk. Furthermore, it is required that for every message m^*, for an honestly generated verification key (sampled independently of m^*), the property value $P_{\mathsf{vk}}(m^*)$ is pseudo-random, even given the verification key and a signature oracle on messages $m \neq m^*$. We can also consider a relaxed notion of invariant signatures in the common random string model (CRS).

The Goldwasser-Ostrovsky transformation is based on the construction of Feige, Lapidot and Shamir [FLS90] of NIZKs in the *hidden-bits model*. In this model, a random hidden string is available to the prover but is hidden from the verifier. The prover can reveal to the verifier specific bits of the hidden string in the locations of its choice, but it cannot change the value of these bits. Very briefly, the transformation is as follows: we interpret the CRS (available to both prover and verifier) as containing a CRS for the invariant signature, as well as a sequence of messages $\{m_i\}$ and one-time pad bits $\{s_i\}$ where every (m_i, s_i) will be used to obtain a single hidden bit b_i. The prover will sample keys $(\mathsf{vk},\mathsf{sk})$ for the invariant signature and send the verification key vk to the verifier as part of the proof. The hidden bit b_i is then defined as the bit $P_{\mathsf{vk}}(m_i)$, the property value of the message m_i, XORed with the the one-time pad bit s_i. To reveal the bit b_i, the prover sends to the verifier a signature σ_i on m_i. The verifier can compute b_i by computing $P(\sigma) = P_{\mathsf{vk}}(m_i)$.

The simulator can be defined based on this strategy, where first we run the simulator of the proof system in the hidden model to obtain a proof π and a set of revealing bits $\{b_i\}$. The CRS will be generated exactly as in the real execution, except that the one-time pad bits $\{s_i\}$ are computed as $P_{\mathsf{vk}}(m_i) \oplus b_i$.

The proof of zero-knowledge is essentially based on pseudo-randomness property of the signature, so that each hidden bit in the simulation is computationally indistinguishable from a uniformly random bit as in the real execution.

There are two important points that make the proof also works in the quantum setting:

- A NIZK proof system in the hidden-bit model can achieve both perfect soundness and perfect zero-knowledge [HU19].
- The computational indistinguishability only appears in the proof of the CRS gener-

ation in the real execution and the simulation, which is classical and independent from the adversary's queries.

Since perfect zero-knowledge implies quantum zero-knowledge, the classical proof also carries to the quantum setting, when the building blocks are post-quantumly secure. For more details, we refer the reader to the (classical) proof given in [BP15].

E Sahai's Construction

Proof of Theorem 2. Completeness follows by inspection. Soundness follows by the fact that if crs is chosen uniformly at random, then the probability that crs_1 can be interpreted as a commitment to any string is exponentially small, and likewise the probability that crs_2 is in the image of the pseudorandom generator PRG is exponentially small.

For the proof of adaptive unbounded zero-knowledge, we note that the only difference in the common random string crs between the real protocol and the simulation is crs_1. However, by post-quantum security of the commitment scheme, the two are computationally indistinguishable. (We note that the commitments are classical.) Thus, since the simulator for Π uses only a different witness to prove the same statement, the view of the adversary in the simulator experiment is computationally indistinguishable from the view of the adversary in the modified prover experiment. Thus, adaptive unbounded zero-knowledge follows.

Quantum simulation-soundness proof. The proof of simulation-soundness follows almost identical as the one in the classical setting [Sah01], except for some small modifications on the reductions to quantum security of building blocks. We give the full proof as follows.

Let $\mathcal{A}$ be a QPT adversary. The proof proceeds by a sequence of games where G_0 is defined in which $\mathcal{A}$ can make quantum queries to $\mathcal{S}_2$ (defined in Figure 1), and the winning condition is defined as in Definition 4. For any game G_i, we denote by $\mathsf{Adv}_i(\mathcal{A})$ the advantage of $\mathcal{A}$ in G_i, that is, $\Pr\big[G_i(1^\lambda, \mathcal{A}) = 1\big]$, where the probability is taken over the random coins of G_i and $\mathcal{A}$.

Game G_0: This is the actual adversary experiment, in which $\mathcal{A}$ can make quantum queries to the simulator $\mathcal{S}_2$ and outputs two pairs $\{(x_i, \pi_i)\}_{i=1}^2$. Let R be the list of all classical randomness $\mathcal{S}_2$ used to answer each adversarial query during the experiment. We say $\mathcal{A}$ wins if either of the following holds:

(a) There exists $i \in [\![1,2]\!]$ such that $x_i \notin \mathcal{L}$, for all $r \in R$, $\mathcal{S}_2(x_i, r) \neq \pi_i$ and $\mathcal{V}(\mathsf{crs}, x_i, \pi_i) = 1$.

(b) There exists a randomness $r \in R$ such that $\mathcal{S}_2(x_1, r) = \pi_1$ and $\mathcal{S}_2(x_2, r) = \pi_2$ and at least one of x_1 or x_2 is not in $\mathcal{L}$.

Game G_1: In this game, we change the winning condition. The winning condition is now defined as:

(a) There exists $i \in [\![1,2]\!]$ such that $x_i \notin \mathcal{L}$, for all $r \in R$, $\mathcal{S}_2(x_i, r) \neq \pi_i$ and $\mathcal{V}(\mathsf{crs}, x_i, \pi_i) = 1$.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_0(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. We show that the probability that the adversary wins by the second condition is negligible, otherwise it must be able to break the unforgeability of the one-time signature. Assume that $\mathcal{A}$ wins by the second condition with non-negligible probability ϵ.

Let T be the list of verification keys output by the simulator. We note that since verification keys (as well as signing keys) are *classically* and independently from the adversary's queries, T is well-defined as a list of classical strings. Furthermore, with all but exponentially small probability, these verification keys will all be distinct. First, we note that if the output of the adversary can be computed from the same randomness $r \in R$, it means that the verification keys vk_1 and vk_2 (as parts of the proofs) output by the adversary also in T, and furthermore it must be the case that $\mathsf{vk}_1 = \mathsf{vk}_2$, and at least one of the two proofs is a forgery of the signature scheme. Denote this verification key as vk^*, and the forge as (m, t).

We show how to use $\mathcal{A}$ to break the (weak) unforgeability of Sig (the security game $\mathsf{W\text{-}BZ\text{-}Exp}$ as defined in Definition 10). Specifically, assume that the adversary $\mathcal{A}$ makes at most q queries to the simulator. The reduction algorithm picks a random index $i \in [\![1, q]\!]$ and uses $\mathcal{A}$'s i-th query in the game $\mathsf{W\text{-}BZ\text{-}Exp}$. With probability $1/q$, this verification key returned by the challenger in the game $\mathsf{W\text{-}BZ\text{-}Exp}$ is vk^*. In this case, the reduction just returns $\mathcal{A}$'s output pairs $\{(x_i, \pi_i = (\mathsf{vk}^*, x_i, u_i, \pi'_i, \sigma_i))\}_{i=1}^2$. If follows that with probability ϵ/q, $\{((x_i, u_i, \pi'_i), \sigma_i)\}_{i=1}^2$ are valid forges of Sig (with respect to vk^*). We note that $x_1 \neq x_2$ by the assumption. This probability is non-negligible if ϵ is non-negligible. The proof of the claim follows. □

Game G_2: In this game, we continue changing the winning condition, as follows:

(a) There exists $i \in [\![1, 2]\!]$ such that for all $r \in R$, $\mathcal{S}_2(x_i, r) \neq \pi_i$ and $\mathcal{V}(\mathsf{crs}, x_i, \pi_i) = 1$ and $u = \mathsf{PRF}(s, \mathsf{vk})$ where (u, vk) is parts of the output of the proof π_i and s is a part of the trapdoor information td.

We note that now this game can be implemented in quantum-polynomial-time.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_1(\mathcal{A}) - \mathsf{Adv}_2(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. Since crs_2 is a uniformly random string, there is a string d such that $\mathsf{crs}_2 = \mathsf{PRG}(d)$ with only negligible probability. By the definition of the language $\mathcal{L}'$ and the fact that Π' is a proof system for $\mathcal{L}'$, we conclude that if $x \notin \mathcal{L}$, the only way the adversary's proof can be accepted is if $\mathsf{PRF}(s, \mathsf{vk}) = u$ with overwhelming probability. This is because the adversary never sees a valid proof for a false statement of $\mathcal{L}'$ (the simulator is generating the simulated proofs using the commitment witness), thus any adversary that outputs a valid proof for a false statement of $\mathcal{L}'$ (which means $x \notin \mathcal{L} \wedge \mathsf{PRF}(s, \mathsf{vk}) \neq u$) would break the soundness of Π'. Therefore, the winning conditions in G_1 and G_2 are exponentially close. □

Game G_3: In this game, we make crs_2 to be pseudorandom. That is, instead of sampling crs_2 uniformly at random, we compute crs_2 by using a pseudorandom generator PRG. The change is described in Figure 4.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_2(\mathcal{A}) - \mathsf{Adv}_3(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The indistinguishability between G_2 and G_3 follows directly from post-quantum security of PRG. □

Game G_4: In this game, the trapdoor information also includes the seed d of PRG. Furthermore, the simulator $\mathcal{S}'_2$, instead of using the witness of the commitments (that is, $(s, a_1, \cdots, a_\lambda)$), uses the seed d for crs_2 to generate the proof π'. The change is described in Figure 5.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_3(\mathcal{A}) - \mathsf{Adv}_4(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

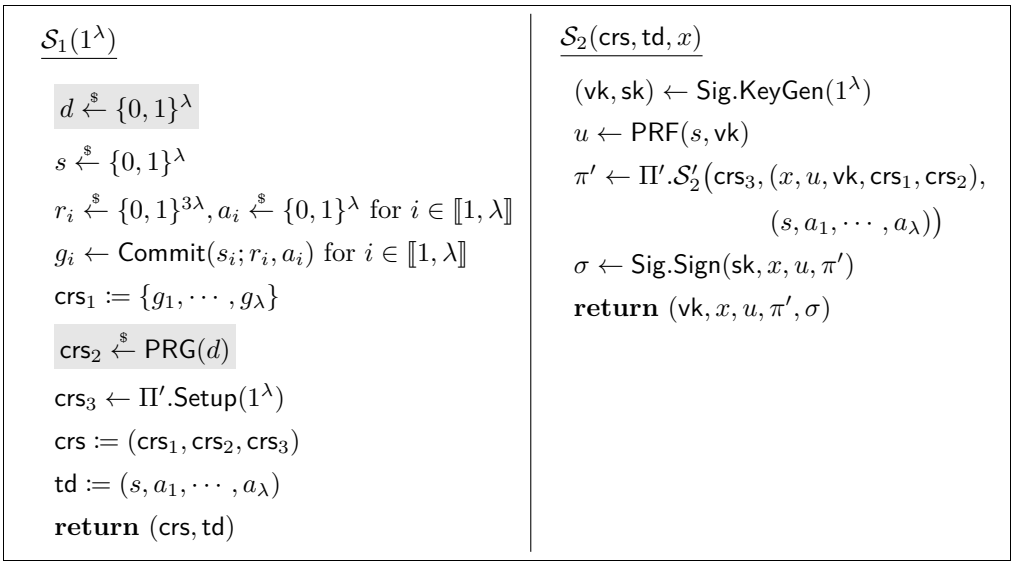

Figure 4: The simulator of game G_3.

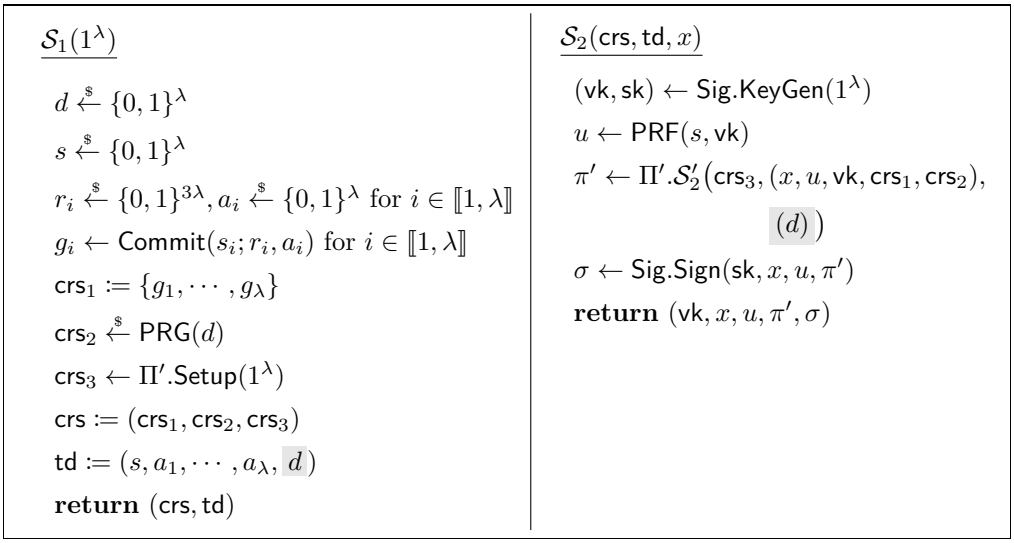

Figure 5: The simulator of game G_4.

Proof. The indistinguishability between G_3 and G_4 follows the *quantum* zero-knowledge property (Definition 3) of Π' (which implies witness-indistinguishability): instead of using witness $(s, a_1, \cdots, a_\lambda)$, we now use witness d to generate the proof. □

Game G_5: In this game, we make crs_1 independent of s: we choose two independent uniformly random strings s, s' and make crs_1 into a commitment to s' rather than s. The change is described in Figure 6.

$\underline{\mathcal{S}_1(1^\lambda)}$	$\underline{\mathcal{S}_2(\mathsf{crs}, \mathsf{td}, x)}$
$d \xleftarrow{\$} \{0,1\}^\lambda$	$(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Sig.KeyGen}(1^\lambda)$
$s, s' \xleftarrow{\$} \{0,1\}^\lambda$	$u \leftarrow \mathsf{PRF}(s, \mathsf{vk})$
$r_i \xleftarrow{\$} \{0,1\}^{3\lambda}, a_i \xleftarrow{\$} \{0,1\}^\lambda$ for $i \in [\![1, \lambda]\!]$	$\pi' \leftarrow \Pi'.\mathcal{S}_2'\big(\mathsf{crs}_3, (x, u, \mathsf{vk}, \mathsf{crs}_1, \mathsf{crs}_2), (d)\big)$
$g_i \leftarrow \mathsf{Commit}(s_i'; r_i, a_i)$ for $i \in [\![1, \lambda]\!]$	$\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}, x, u, \pi')$
$\mathsf{crs}_1 \coloneqq \{g_1, \cdots, g_\lambda\}$	**return** $(\mathsf{vk}, x, u, \pi', \sigma)$
$\mathsf{crs}_2 \xleftarrow{\$} \mathsf{PRG}(d)$	
$\mathsf{crs}_3 \leftarrow \Pi'.\mathsf{Setup}(1^\lambda)$	
$\mathsf{crs} \coloneqq (\mathsf{crs}_1, \mathsf{crs}_2, \mathsf{crs}_3)$	
$\mathsf{td} \coloneqq (s, a_1, \cdots, a_\lambda, d)$	
return $(\mathsf{crs}, \mathsf{td})$	

Figure 6: The simulator of game G_5.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_4(\mathcal{A}) - \mathsf{Adv}_5(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The indistinguishability between G_4 and G_5 follows the computational hiding property of the Naor's commitment scheme. □

Game G_6: In this game, we replace PRF with a truly random function H (lazy-sampling). The change is described in Figure 7.

Claim. *For any* QPT *adversary* $\mathcal{A}$, $|\mathsf{Adv}_5(\mathcal{A}) - \mathsf{Adv}_6(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The indistinguishability between G_5 and G_6 follows pseudorandomnesss of PRF. Note that here since vk is classical, we only need classical pseudorandomess of PRF against quantum adversaries. □

Claim. *For any adversary* $\mathcal{A}$, $\mathsf{Adv}_6(\mathcal{A}) \leq 2^{-\lambda}$.

Proof. Since we only consider the case where $\mathsf{vk}^* \notin T$, for any vk^* output by $\mathcal{A}$, $H(\mathsf{vk}^*)$ will be a uniformly selected value that is totally independent of everything the adversary sees. Denote this value as u'. Then the probability that the proof output by $\mathcal{A}$ having $u = u'$ is exactly $2^{-\lambda}$. The claim follows. □

Overall, we conclude the proof of the theorem. □

$\underline{\mathcal{S}_1(1^\lambda)}$

$d \xleftarrow{\$} \{0,1\}^\lambda$

$s, s' \xleftarrow{\$} \{0,1\}^\lambda$

$r_i \xleftarrow{\$} \{0,1\}^{3\lambda}, a_i \xleftarrow{\$} \{0,1\}^\lambda$ for $i \in [\![1,\lambda]\!]$

$g_i \leftarrow \mathsf{Commit}(s'_i; r_i, a_i)$ for $i \in [\![1,\lambda]\!]$

$\mathsf{crs}_1 := \{g_1, \cdots, g_\lambda\}$

$\mathsf{crs}_2 \xleftarrow{\$} \mathsf{PRG}(d)$

$\mathsf{crs}_3 \leftarrow \Pi'.\mathsf{Setup}(1^\lambda)$

$\mathsf{crs} := (\mathsf{crs}_1, \mathsf{crs}_2, \mathsf{crs}_3)$

$\mathsf{td} := (s, a_1, \cdots, a_\lambda, d)$

return $(\mathsf{crs}, \mathsf{td})$

$\underline{\mathcal{S}_2(\mathsf{crs}, \mathsf{td}, x)}$

$(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Sig.KeyGen}(1^\lambda)$

$u \xleftarrow{\$} \{0,1\}^\lambda$

$\pi' \leftarrow \Pi'.\mathcal{S}'_2\big(\mathsf{crs}_3, (x, u, \mathsf{vk}, \mathsf{crs}_1, \mathsf{crs}_2), (d)\big)$

$\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}, x, u, \pi')$

return $(\mathsf{vk}, x, u, \pi', \sigma)$

Figure 7: The simulator of game G_6.

F Separation II: Simulation Sound NIZK in the CRS Model

In this section, we construct a simulation sound (extractable) NIZK[6] in the CRS model that is secure against a classical adversary. Then we show our separation result that is the simulation sound property of the construction can be broken if the adversary allows making superposition queries. Formally, we prove the following theorems:

Theorem 5. *Assume that the underlying NIZK scheme satisfies perfect completeness, computational soundness, and computational zero-knowledge, that the encryption scheme is semantically secure and perfectly correct, that the pseudo-random function family is secure, that the commitment scheme is perfectly binding and computational hiding, and that the one-time signature scheme is strongly unforgeable. Then the construction Π_{SE} is a zero-knowledge proof system satisfying perfect completeness, computational zero-knowledge, and simulation sound extractability.*

Theorem 6. *If secure simulation (extractable) sound NIZKs exist, then there are standard-secure simulation (extractable) sound NIZKs that are not secure simulation (extractable) sound when an adversary queries in superpositions of the statement.*

Before describing our protocol formally, we first recall some notation and the primitives used in the construction.

Ingredients and notation.

- An encryption scheme $\mathcal{E}$ is semantically secure and perfectly correct.
- A pseudo-random function family F is standard-secure PRFs.
- A pseudo-random permutation family PRP is standard-secure PRPs.
- A commitment scheme Com is perfectly binding and computational hiding.

[6]The stronger notion of simulation sound is called simulation sound extractability [Gro06], which is a combination of the definitions of simulation soundness introduced by Sahai [Sah01] and proofs of knowledge from De Santis and Persiano [DP92].

- A one-time signature scheme Sig is strongly unforgeable.
- A NIZK scheme NIZK satisfies perfect completeness, computational soundness, and computational zero-knowledge.

F.1 Periodic PRFs

Our construction makes use of a pseudo-random function that is a PRF to be a standard-secure pseudorandom function with key-space K, domain X, and co-domain Y. We will construct a new pseudorandom function that is periodic with some large, secret period. Classical adversaries will not be able to detect the period, and thus cannot distinguish this new function from random. However, an adversary making quantum queries can detect the period, and thus distinguish our new function from random.

We now construct a new pseudo-random function F as follows,

$$\mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x}) = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x}) \oplus \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x} \oplus \mathsf{p}_{\mathsf{PRF}}),$$

where $\mathsf{sk}_{\mathsf{PRF}} \xleftarrow{\$} \{0,1\}^{\lambda}$ is the secret key of the PRF. The function $\mathsf{f}(.)$ is a PRF function . The value $\mathsf{p}_{\mathsf{PRF}}$ is the perodic of the function F which later we set $\mathsf{p}_{\mathsf{PRF}} = \mathsf{sk}_{\mathsf{PRF}}$.

Let K be the key space, and X and Y are the domain and range. K, X, and Y are implicit functions of the security parameter λ. Assume without loss of generality that Y contains at least $N/2$ elements (if not, we can construct a new pseudorandom function with a smaller domain but larger range in a standard way). In theorem 7, we prove that the function F is a pseudorandom function : $K \times X \to Y$.

Theorem 7. *Assume a function* f *is a standard-secure PRF. Then the pseudo-random function* F *is standard-secure PRFs and periodic.*

Proof. Pseudo-randomness property: it directly follows from the pseudo-random function f. *Perodic peroperty*: The function F is periodic with value $\mathsf{p}_{\mathsf{PRF}}$ iff $\mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x}) = \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x} \oplus \mathsf{p}_{\mathsf{PRF}})$. We observe that $\mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x} \oplus \mathsf{p}_{\mathsf{PRF}}) = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x} \oplus \mathsf{p}_{\mathsf{PRF}}) \oplus \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(x \oplus \mathsf{p}_{\mathsf{PRF}} \oplus \mathsf{p}_{\mathsf{PRF}}) = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x} \oplus \mathsf{p}_{\mathsf{PRF}}) \oplus \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(x) = \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$. □

F.2 Construction

Before describing our protocol formally, to help the exposition, we first give a brief intuition of the construction.

Intuition. For the simulation extractable property, a prover must always provide encryption of a witness. Inspired by the idea of [AGRS24, ARS20, KZM+15] with some modifications, our construction makes use of a one-time signature scheme. A pair of one-time signing/verification keys are generated for each proof such that in the zero-knowledge proof, a simulator (simulated prover) is required to provide $\mathsf{ct}_{\mathsf{F}} = \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$ (where the function F is the pseudo-random function introduced in section F.1) and $\mathsf{ct}_{\mathsf{pk}} = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk})$ (where f is a pseudo-random function). Then we require the prover to sign the statement together with the proof, the cipher-text, $\mathsf{ct}_{\mathsf{pk}}$, and ct_{F}. Then, briefly, due to the security of the signature scheme, the adversary must use a different x and pk from the ones returned from oracle queries. Thus, for a statement to pass the verifier without a proper witness, the prover must generate $\mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$ and $\mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk})$ without the knowledge of $\mathsf{sk}_{\mathsf{PRF}}$ (thus breaking the pseudo-random function F and f).

Construction. Let $\mathcal{L}$ be a NP language for a relation R. Define a language $\mathcal{L}'$ to be the language that $(\mathsf{x}, , \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r, r_0, \mathsf{w}, a, b) \in \mathsf{R}_{\mathcal{L}}$ iff

$$\begin{aligned} \mathsf{ct} = \mathcal{E}_{\mathsf{pk}_e}(\mathsf{w}, r_0) \wedge (\mathsf{x}, \mathsf{w}) \in \mathcal{L} \vee (\mathsf{ct}_{\mathsf{F}} = \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x}) \wedge \mathsf{ct}_{\mathsf{pk}} = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk}_s) \wedge \\ \mathsf{ct}_c = \mathsf{Com}(\mathsf{sk}_{\mathsf{PRF}}, r) \wedge \mathsf{ct}_r = \mathsf{PRP}_{\mathsf{sk}_{\mathsf{PRF}}}(r)), \end{aligned}$$

where the function F is a pseudo-random function introduced in section F.1. PRP is a pseudo-random permutation function with the key $\mathsf{sk}_{\mathsf{PRF}}$.

Now, we show the construction from NIZKs to simulation (extractable) sound NIZKs. The construction is similar to the construction in [KZM+15] but with some modification in a way that.

- $\mathsf{crs}.\mathsf{KGen}(1^\lambda, \mathcal{L})$: On input a security parameter λ and a language $\mathcal{L}$ compute,
  - $\mathsf{NIZK.crs} \leftarrow \mathsf{NIZK.KGen}(1^\lambda, \mathcal{L}')$;
  - $\mathsf{pk}_e, \mathsf{sk}_e) \leftarrow \mathcal{E}.\mathsf{KGen}(1^\lambda)$;
  - $\mathsf{sk}_{\mathsf{PRF}}, r \xleftarrow{\$} \{0,1\}^\lambda$, set the periodic value $\mathsf{p}_{\mathsf{PRF}} = \mathsf{sk}_{\mathsf{PRF}}$ (that is the periodic value of the PRF function F). Compute $\mathsf{ct}_c = \mathsf{Com}(\mathsf{sk}_{\mathsf{PRF}}, r)$ and $\mathsf{ct}_r = \mathsf{PRP}_{\mathsf{sk}_{\mathsf{PRF}}}(r)$.

  Set $\mathsf{crs}' := (\mathsf{NIZK.crs}, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r)$.

- $\mathcal{P}(\mathsf{crs}', \mathsf{x}, \mathsf{w})$: On input a CRS crs', a statement x, and a witness w,
  - Parse $\mathsf{crs}' = (\mathsf{NIZK.crs}, \mathsf{pk}_2, \mathsf{ct}_c)$. If $(\mathsf{x}, \mathsf{w}) \notin \mathsf{R}_{\mathcal{L}}$ then abort;
  - Run $(\mathsf{pk}_s, \mathsf{sk}_s) \leftarrow \mathsf{Sig.KGen}(1^\lambda)$ and sample $z_0, z_0', z_1, z_2, r_0 \xleftarrow{\$} \{0,1\}^\lambda$. Set $\mathsf{ct}_{\mathsf{F}} := z_0$ and $\mathsf{ct}_{\mathsf{pk}} := z_0'$
  - Compute $\mathsf{ct} \leftarrow \mathcal{E}_{\mathsf{pk}_e}(\mathsf{w}, r_0)$;
  - Compute $\mathsf{NIZK}.\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{NIZK.crs}, (\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r_0, z_1, \mathsf{w}, z_2))$;
  - Compute $\sigma \leftarrow \mathsf{Sig}_{\mathsf{sk}_s}(\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi)$;

  Return $\pi := (\mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi, \mathsf{pk}_s, \sigma)$.

- $\mathcal{V}(\mathsf{crs}', \mathsf{x}, \pi)$: On input a CRS crs', a statement x, and a proof π, check,
  - Parse $\mathsf{crs}' := (\mathsf{NIZK.crs}, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r)$ and $\pi := (\mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi, \mathsf{pk}_s, \sigma)$;
  - Abort if $0 \leftarrow \mathsf{Sig}.\mathcal{V}(\mathsf{pk}_s, (\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi), \sigma)$;
  - Abort if $0 \leftarrow \mathsf{NIZK}.\mathcal{V}(\mathsf{NIZK}.crs, (\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), \mathsf{NIZK}.\pi)$.

- $\mathcal{S}(1^\lambda, \mathcal{L})$: It contains two algorithms $\mathcal{S}_1$ and $\mathcal{S}_2$. The algorithm $\mathcal{S}_1(1^\lambda, \mathcal{L})$ runs the $\mathsf{crs}.\mathsf{KGen}(1^\lambda, \mathcal{L})$ and outputs crs', sets a trapdoor $\mathsf{td} := (\mathsf{sk}_{\mathsf{PRF}}, r)$, and an extraction key $\mathsf{ek} := \mathsf{sk}_c$. The algorithm $\mathcal{S}_2(\mathsf{crs}', \mathsf{x}, \mathsf{td})$ works as follows:
  - Parse $\mathsf{crs}' = (\mathsf{NIZK.crs}, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r)$ and $\mathsf{td} := (\mathsf{sk}_{\mathsf{PRF}}, r)$;
  - Run $(\mathsf{pk}_s, \mathsf{sk}_s) \leftarrow \mathsf{Sig.KGen}(1^\lambda)$ and sample $z_3, r_0 \xleftarrow{\$} \{0,1\}^\lambda$. Compute $\mathsf{ct}_{\mathsf{F}} \leftarrow \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$ and $\mathsf{ct}_{\mathsf{pk}} \leftarrow \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk}_s)$.
  - Compute $\mathsf{ct} \leftarrow \mathcal{E}_{\mathsf{pk}_e}(z_3, r_0)$;
  - Compute $\mathsf{NIZK}.\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{NIZK.crs}, (\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r_0, z_3, \mathsf{td} = (r, \mathsf{sk}_{\mathsf{PRF}})))$;
  - Compute $\sigma \leftarrow \mathsf{Sig}_{\mathsf{sk}_s}(\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi)$;
  - Return $\pi := (\mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi, \mathsf{pk}_s, \sigma)$.

$\underline{\mathsf{Exp}(1^\lambda, \mathcal{L})}$

$\mathsf{NIZK.crs} \leftarrow \mathsf{NIZK.KGen}(1^\lambda, \mathcal{L}')$;

$(\mathsf{pk}_e, \mathsf{sk}_e) \leftarrow \mathcal{E}.\mathsf{KGen}(1^\lambda)$;

$\mathsf{sk}_{\mathsf{PRF}}, r \xleftarrow{\$} \{0,1\}^\lambda$;

$\mathsf{ct}_c = \mathsf{Com}(\mathsf{sk}_{\mathsf{PRF}}, r)$;

$\mathsf{ct}_r = \mathsf{PRP}_{\mathsf{sk}_{\mathsf{PRF}}}(r)$;

$\mathsf{crs}' := (\mathsf{NIZK.crs}, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r)$

$\{(x_i, \pi_i)\}_{i=1}^2 \leftarrow \mathcal{A}^{\mathsf{O}(.)}(\mathsf{crs}')$;

if $\mathcal{V}(\mathsf{crs}, x_i, \pi_i) = 1 \,\forall\, i \wedge$

$(\exists i : x_i \notin \mathcal{L}); \wedge$

$\Big(\mathsf{O}(x_i, r) \neq \pi_i \,\forall\, r \in R)\Big) \vee$

$(\exists r \in R : \mathsf{O}(x_j, r) = \pi_j \,\forall\, j)$ for $i, j \in [\![1, 2]\!]$ **return** 1

else return 0

$\underline{\mathsf{O(x)}}$

Parse $\mathsf{crs}' = (\mathsf{NIZK.crs}, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r)$ and $\mathsf{td} := (\mathsf{sk}_{\mathsf{PRF}}, r)$;

$(\mathsf{pk}_s, \mathsf{sk}_s) \leftarrow \mathsf{Sig.KGen}(1^\lambda)$

$z_3, r_0 \xleftarrow{\$} \{0,1\}^\lambda$;

$\mathsf{ct}_{\mathsf{F}} \leftarrow \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$; $\mathsf{ct}_{\mathsf{pk}} \leftarrow \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk}_s)$;

$\mathsf{ct} \leftarrow \mathcal{E}_{\mathsf{pk}_e}(z_3, r_0)$;

$\mathsf{NIZK}.\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{NIZK.crs}, (\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r_0, z_3, \mathsf{td}))$;

$\sigma \leftarrow \mathsf{Sig}_{\mathsf{sk}_s}(\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi)$;

return

$\pi := (\mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi, \mathsf{pk}_s, \sigma)$.

Figure 8: Experiment $\mathsf{Exp}(1^\lambda, \mathcal{L})$ for the simulation-sound proof of Theorem 5.

F.3 Security Proof

Next, we prove Theorem 5. The *completeness* is straight forward from the construction. Next we prove it is also *simulation sound (extractable)* and *zero-knowledge*.

Proof. **Simulation sound (extractable).** We prove it in several games. We recall the experiment for simulation sound extractable in Fig. 8 and we highlight changes by pointing to the line numbers in the experiment or the oracle.

Game G_0: This is the original experiment in Fig. 8.

Game G_1: This game is the same as G_0 but we relax the return condition as follows: let T be the set of verification keys generated by $\mathsf{O}(.)$. The experiment Exp_1 outputs 1 iff:

- $\mathcal{V}(\mathsf{crs}, x_i, \pi_i) = 1 \,\forall\, i \;\wedge\; \exists i : (\mathsf{O}(x_i, r) \neq \pi_i \,\forall\, r \in R) \vee (\exists r \in R : \mathsf{O}(x_j, r) = \pi_j \,\forall\, j)$ for $i, j \in [\![1, 2]\!]$
- $\wedge\ \mathsf{pk}_s \notin T \wedge \mathsf{ct}_{\mathsf{pk}} = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk}_s) \wedge ct_{\mathsf{F}} = \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$.

Claim. *If the underlying one-time signature scheme is strongly unforgeable, and that the underlying NIZK is sound, then we have any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_0(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. We know that if (i) x and π are not queried before and (ii) pk_s has been generated by the oracle $\mathsf{O}(\mathring{\mathsf{u}})$, then the $(\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi)$ is a valid message/signature pair. Hence by the unforgeability of the signature scheme, we know that (i) and (ii) happen with negligible probability, thus, we focus on $\mathsf{pk}_s \notin T$. Furthermore, if some witness (the decrypted w is unique for all valid witnesses) is valid for $\mathcal{L}$ and that $(\mathsf{x}, \mathsf{w}) \notin \mathsf{R}_{\mathcal{L}}$, then it has to be the case that there exists some $\mathsf{sk}'_{\mathsf{PRF}}$, such that ct_c is a valid commitment of $\mathsf{sk}'_{\mathsf{PRF}}$ and that $\mathsf{ct}_{\mathsf{pk}} = \mathsf{f}_{\mathsf{sk}'_{\mathsf{PRF}}}(\mathsf{pk}_s)$, which implies $\mathsf{ct}_{\mathsf{pk}} = \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk}_s)$, by the perfectly binding property. □

Game G_2: This game is the same as G_1 but for generating the $\mathsf{NIZK}.\pi$, instead of the $\mathsf{NIZK}.\mathcal{P}$ we run the simulator $\mathsf{NIZK}.\mathcal{S}$ of the underlying NIZK. Thus, the experiment Exp_2 has the following changes:

- $(\mathsf{NIZK.crs}, \mathsf{td}_{\mathsf{NIZK.crs}}) \leftarrow \mathsf{NIZK.KGen}(1^\lambda, \mathcal{L}')$;
- $\underline{\mathsf{O}(\mathsf{x})}$: $\mathsf{NIZK}.\pi \leftarrow \mathsf{NIZK}.\mathcal{S}(\mathsf{NIZK.crs}, (\mathsf{x}, \mathsf{ct}, \mathsf{ct}_\mathsf{F}, \mathsf{ct}_\mathsf{pk}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r_0, z_3, \mathsf{td}_{\mathsf{NIZK.crs}}))$.

Claim. *If the underlying NIZK is zero-knowledge, then we have that for any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_2(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The proof follows directly from the zero-knowledge property of the underlying NIZK. □

Game G_3: This game is the same as G_2 but we use different sk_PRF in ct_c than the one used in ct_F, ct_pk, and ct_r. Thus, the experiment Exp_3 has the following changes:

- $\mathsf{sk}'_\mathsf{PRF}, \mathsf{sk}_\mathsf{PRF}, r \xleftarrow{\$} \{0,1\}^\lambda$;
- $\mathsf{ct}_c = \mathsf{Com}(\mathsf{sk}'_\mathsf{PRF}, r)$;

Claim. *If the underlying commitment scheme is computationally hiding, then we have any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_3(\mathcal{A}) - \mathsf{Adv}_2(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. By the hiding property, no polynomial algorithm can distinguish the commitment of two elements. □

Game G_4: This game is the same as G_3 but we replace PRP with a true random permutation PRP'. Thus, the experiment Exp_4 has the following changes:

- $\mathsf{ct}_r = \mathsf{PRP}'_{\mathsf{sk}_\mathsf{PRF}}(r)$;

Claim. *If the underlying* PRP *scheme is secure, then we have any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_4(\mathcal{A}) - \mathsf{Adv}_3(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The proof follows directly from the security of the PRP scheme. □

Game G_5: This game is the same as G_4 but we replace PRF with a truly random function f'. Thus, the experiment Exp_5 has the following changes:

- $\underline{\mathsf{O}(\mathsf{x})}$: $\mathsf{ct}_\mathsf{F} \leftarrow \mathsf{f}'(\mathsf{x})$; $\mathsf{ct}_\mathsf{pk} \leftarrow \mathsf{f}'(\mathsf{pk}_s)$;
- $\underline{\text{Outputs 1 iff:}}$ $\wedge\ \mathsf{pk}_s \notin T \wedge \mathsf{ct}_\mathsf{pk} = \mathsf{f}'(\mathsf{pk}_s) \wedge ct_\mathsf{F} = \mathsf{f}'(\mathsf{x})$.

Claim. *If the underlying PRF is secure, then we have that for any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_5(\mathcal{A}) - \mathsf{Adv}_4(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. As the PRF is secure, the outputs of f' are indistinguishable from f and F. This completes the proof. □

Finally, since $\mathsf{pk}_s \notin T$ then $\mathsf{f}(pk_s)$ has not been queried before. Thus, we may view $\mathsf{f}(\mathsf{pk}_s)$ as newly generated random bits independent from ct_pk. So, it concludes that $Pr[\mathsf{Exp}_5] \leq \mathsf{negl}(\lambda)$. This completes the proof.

Zero-knowledge. We prove this by showing a series of indistinguishable hybrids. We recall the experiment for zero-knowledge in Fig. 9 and we highlight changes by pointing to the line numbers in the experiment or the oracle.

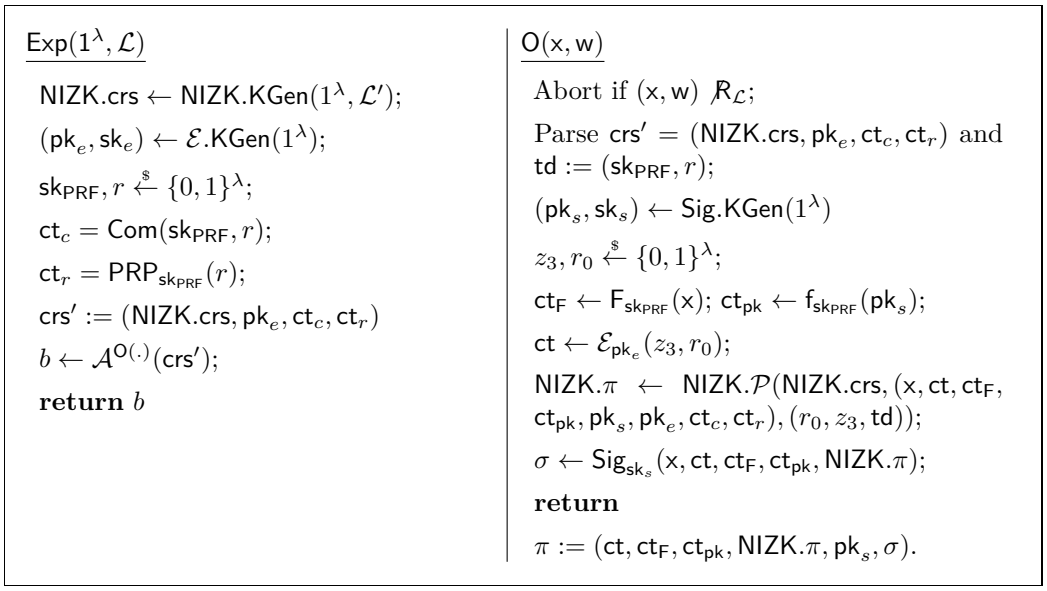

Figure 9: Experiment $\mathsf{Exp}(1^\lambda, \mathcal{L})$ for the zero-knowledge proof of Theorem 5.

Game G_0: This is the original experiment in Fig. 9.

Game G_1: This game is the same as G_0 but for generating the $\mathsf{NIZK}.\pi$, instead of the $\mathsf{NIZK}.\mathcal{P}$ we run the simulator $\mathsf{NIZK}.\mathcal{S}$ of the underlying NIZK. Thus, the experiment Exp_1 has the following changes:

- $(\mathsf{NIZK.crs}, \mathsf{td}_{\mathsf{NIZK.crs}}) \leftarrow \mathsf{NIZK.KGen}(1^\lambda, \mathcal{L}')$;
- $\underline{\mathsf{O}(\mathsf{x})}$: $\mathsf{NIZK}.\pi \leftarrow \mathsf{NIZK}.\mathcal{S}(\mathsf{NIZK.crs}, (\mathsf{x}, \mathsf{ct}, \mathsf{ct_F}, \mathsf{ct_{pk}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r_0, z_3, \mathsf{td}_{\mathsf{NIZK.crs}}))$.

Claim. *If the underlying NIZK is zero-knowledge, then we have that for any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_1(\mathcal{A}) - \mathsf{Adv}_0(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The proof follows directly from the zero-knowledge property of the underlying NIZK. □

Game G_2: This game is the same as G_1 but the oracle O encrypts the true witness. The experiment Exp_2 has the following changes:

- $\underline{\mathsf{O}(\mathsf{x})}$: $\mathsf{ct} \leftarrow \mathcal{E}_{\mathsf{pk}_e}(\mathsf{w}, r_0)$

Claim. *If the underlying encryption scheme is semantically secure, then we have any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_2(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The proof follows directly from the semantical security of the encryption scheme. □

Game G_3: In his game we only use different $\mathsf{sk_{PRF}}$ in ct_c than the one used in $\mathsf{ct_F}$, $\mathsf{ct_{pk}}$, and ct_r. Thus, the experiment Exp_3 has the following changes:

- $\mathsf{sk}'_{\mathsf{PRF}}, \mathsf{sk_{PRF}}, r \xleftarrow{\$} \{0,1\}^\lambda$;
- $\mathsf{ct}_c = \mathsf{Com}(\mathsf{sk}'_{\mathsf{PRF}}, r)$;

Claim. *If the underlying commitment scheme is computationally hiding, then we have any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_3(\mathcal{A}) - \mathsf{Adv}_2(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The proof follows directly from the hiding property of the commitment scheme. □

Game G_4: This game is the same as G_3 but we only replace PRP with a true random permutation PRP'. Thus, the experiment Exp_4 has the following change:

- $\mathsf{ct}_r = \mathsf{PRP}'_{\mathsf{sk}_{\mathsf{PRF}}}(r)$;

Claim. *If the underlying* PRP *scheme is secure, then we have that for any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_4(\mathcal{A}) - \mathsf{Adv}_3(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The proof follows directly from the security of the PRP scheme. □

Game G_5: This game is the same as G_4 but we replace PRF with a truly random function f'. Thus, the experiment Exp_5 has the following changes:

- $\underline{\mathsf{O}(\mathsf{x})}$: $\mathsf{ct}_{\mathsf{F}} \leftarrow \mathsf{f}'(\mathsf{x})$; $\mathsf{ct}_{\mathsf{pk}} \leftarrow \mathsf{f}'(\mathsf{pk}_s)$;

Claim. *If the underlying PRF is secure, then we have that for any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_5(\mathcal{A}) - \mathsf{Adv}_4(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. As the PRF is secure, the outputs of f' are indistinguishable from f and F. This completes the proof. □

Game G_6: In this game, we replace the real prover $\mathcal{P}(\mathsf{w}, \mathsf{x})$ to generate the proof $\pi := (\mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi, \mathsf{pk}_s, \sigma)$. Thus, the experiment Exp_6 has the following changes in the $\underline{\mathsf{O}(\mathsf{x})}$:

- Sample$z_0, z'_0, z_1, z_2, r_0 \xleftarrow{\$} \{0,1\}^\lambda$; Set $\mathsf{ct}_{\mathsf{F}} := z_0$ and $\mathsf{ct}_{\mathsf{pk}} := z'_0$
- $\mathsf{NIZK}.\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{NIZK.crs}, (\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r_0, z_1, \mathsf{w}, z_2))$;

Claim. *If the underlying NIZK is zero-knowledge, then we have that for any PPT adversary* $\mathcal{A}$, $|\mathsf{Adv}_6(\mathcal{A}) - \mathsf{Adv}_5(\mathcal{A})| \leq \mathsf{negl}(\lambda)$.

Proof. The last experiment is exactly the definition. This concludes the proof for the zero-knowledge part. □

This concludes the proof.

□

F.4 Separation Result: Non-Secure SS-NIZK Against Superpositions Query

In this section, we show our separation result in Theorem 6 that is the standard-secure simulation (extractable) sound NIZKs that are not secure simulation (extractable) sound when an adversary query in superpositions of the statement. We prove Theorem 6 as follows:

Proof. Let a quantum adversary $\mathcal{A}$ against the pseudo-random function F. Let F be a standard-secure pseudorandom function (introduced in Section F.1). By using Simon's quantum algorithm, given quantum queries, the quantum adversary $\mathcal{A}$ can find the period of the pseudo-random function F and thus obtain $\mathsf{p}_{\mathsf{PRF}} = \mathsf{sk}_{\mathsf{PRF}}$. Let $\mathcal{B}$ be a quantum adversary against simulation (extractable) sound property with allowing quantum queries on the statement x. The adversary $\mathcal{B}(\mathsf{crs}', \pi)$ for $\mathsf{x}' \notin \mathcal{L}$ works as follows:

- Run the adversary $\mathcal{A}(\mathsf{ct}_{\mathsf{F}})$ and return back the PRF's secret key $\mathsf{sk}_{\mathsf{PRF}}$. Extract the randomness r from ct_r. Set the trapdoor $\mathsf{td} = (\mathsf{sk}_{\mathsf{PRF}}, r)$.
- Run $(\mathsf{pk}_s, \mathsf{sk}_s) \leftarrow \mathsf{Sig.KGen}(1^\lambda)$ and sample $z_3, r_0 \xleftarrow{\$} \{0,1\}^\lambda$. Compute $\mathsf{ct}_{\mathsf{F}} \leftarrow \mathsf{F}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{x})$ and $\mathsf{ct}_{\mathsf{pk}} \leftarrow \mathsf{f}_{\mathsf{sk}_{\mathsf{PRF}}}(\mathsf{pk}_s)$;
- Compute $\mathsf{ct} \leftarrow \mathcal{E}_{\mathsf{pk}_e}(z_3, r_0)$;
- Compute $\mathsf{NIZK}.\pi \leftarrow \mathsf{NIZK}.\mathcal{P}(\mathsf{NIZK.crs}, (\mathsf{x}', \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{pk}_s, \mathsf{pk}_e, \mathsf{ct}_c, \mathsf{ct}_r), (r_0, z_3, \mathsf{td} = (r, \mathsf{sk}_{\mathsf{PRF}})))$;
- Compute $\sigma \leftarrow \mathsf{Sig}_{\mathsf{sk}_s}(\mathsf{x}, \mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi)$;
- Return $\pi := (\mathsf{ct}, \mathsf{ct}_{\mathsf{F}}, \mathsf{ct}_{\mathsf{pk}}, \mathsf{NIZK}.\pi, \mathsf{pk}_s, \sigma)$.

This concludes the proof. □