Non-Interactive Zero-Knowledge Proofs with Certified Deletion *

Kasra Abbaszadeh[1] and Jonathan Katz[2]

[1] University of Maryland
kasraz@umd.edu
[2] Google and University of Maryland
jkatz2@gmail.com

Abstract. We introduce the notion of non-interactive zero-knowledge (NIZK) proofs with *certified deletion*. Our notion enables the recipient of a quantum NIZK proof for a (quantumly hard) NP statement to delete the proof and collapse it into a classical deletion certificate. Once this certificate is successfully validated, we require the recipient of the proof to lose their ability to find any accepting inputs to NIZK verification.
We formally define this notion and build several candidate constructions from standard cryptographic assumptions. In particular, we propose a primary construction from classical NIZK for NP and one-way functions, albeit with two limitations: (i) deletion certificates are only privately verifiable, and (ii) both prover and verifier are required to be quantum algorithms. We resolve these hurdles in two extensions that assume the quantum hardness of the learning with errors problem. The first one achieves *publicly verifiable* certificates and the second one requires merely *classical communication* between classical provers and quantum verifiers. Our results have applications to the *revocable signatures of knowledge* and *revocable anonymous credentials*, which we define and construct.

Keywords: quantum cryptography, revocable cryptography, certified deletion, non-interactive zero-knowledge proofs, anonymous credentials.

* Work supported in part by NSF award CNS-2154705.

Contents

1 Introduction

Recent advances in quantum computing have enabled cryptographic capabilities that are impossible in a classical world, including secure key-agreement [BB14], quantum money [Wie83], copy-protection [Aar09], certified deletion [BI20,BK23], unclonable encryption [Got03,BL20]. A standard technique commonly used here is to encode a piece of secret information into a quantum state and protect the information from being copied by leveraging the *no-cloning* principle.

In this framework, Aaronson [Aar09] introduced quantum copy protection. This primitive enables the encoding of an unlearnable function f into a quantum state such that no adversary with access to a single copy can generate two states, both of which can compute f. Ananth et al. [ALP21] introduced a similar but weaker notion of secure software leasing. Here, we enable the encoding of software into a state that can be leased, and the lessee can later return it provably. Once this return is validated, the lessee no longer can execute the leased software.

Building on these foundations, several recent works has explored various cases of revocable cryptographic functionalities. Ananth et al. [APV23] and Agrawal et al. [AKN+23] proposed revocable public-key encryption protocols. Here, a decryption key can be temporarily leased. Morimae et al. [MPY24] studied the case of digital signatures with revocable signing keys and signatures; the former enables leasing signing keys, while the latter allows leasing signatures.

We follow this line of research and introduce an analogous notion in the case of non-interactive zero-knowledge (NIZK) [GMR85]. An NIZK enables a prover holding the witness to a hard NP problem to convince a verifier of the truth of the statement without leaking any additional information about the witness. A fundamental barrier of NIZKs in the classical world is that once a recipient obtains a proof, they can verify and reuse it evermore. Therefore, one can simply observe that classical NIZK proofs are inherently undeniable even if we rely on the random oracle or common reference string models [Pas03]. In some scenarios, however, it might be desirable to revoke the recipient's access to proofs after verification, preventing any future reuse for unintended purposes.

In this work, we take a step forward to address this limitation in the quantum world, and we introduce NIZK with *certified deletion* (NIZK-CD). Here, we allow the recipient of a quantum NIZK proof for a (quantumly) hard NP statement to verify the proof multiple times and, at any point in the future, collapse the proof into some classical deletion certificate. Once this certificate is validated, the security of the primitive guarantees that the recipient looses the access to the proof and can no longer find an accepting input to NIZK verification algorithm.

Consider the following scenario as an illustrative application of NIZK-CD. Suppose a government agency holds a classified document relevant to a court case. Instead of revealing the document's sensitive contents, the agency could provide a NIZK proof attesting to the document's authenticity, satisfying legal requirements without compromising the privacy of the information. Moreover, the certified deletion property of a NIZK-CD allows the court to immediately delete the proof after verification. This ensures that no unauthorized parties can access and misuse the provided information at some point in the future.

1.1 Our Results

We formally define the notion of NIZK-CD as a novel cryptographic primitive. We then propose a primary construction assuming the existence of post-quantum classical NIZK for NP and post-quantum one-way functions. This construction, however, has two limitations: (i) the deletion certificate is privately verifiable and can only be validated by the initial prover, and (ii) while the deletion certificate and its validation are classical, the NIZK proof itself includes quantum states. Therefore, the prover and the verifier are inherently quantum algorithms.

We further address these limitations by proposing two extensions assuming the hardness of learning-with-errors (LWE) [Reg09]. In our first extension, we leverage compute-and-compare program obfuscation [WZ17] to achieve public verifiability for deletion certificates. In the second extension, we show how the prover can be entirely classical and remotely and securely prepare the required quantum piece of information included in NIZK proofs in the verifier's device.

As a natural application, we also construct *revocable signature of knowledge* from NIZK-CD. This can address the long-standing problem of having revocable anonymous credentials from standard assumptions, without the need to rely on conventional blocklisting and time-expiring approaches [BCC+09,AN11,CKS10].

1.2 Related Works

We review prior work on certified deletion and copy-protection.

Certified deletion and revocable cryptography. Unruh [Unr14] introduced quantum revocable encryption, where the recipient of a quantum ciphertext can return the state and, hence, lose all information about the encrypted message. Broadbent et al. [BL20] studied one-time pad encryption with certified deletion, where quantum ciphertexts can be collapsed into classical deletion certificates. Several recent works [HMNY21,Por23,BK23,BGK+24,BKM+23] extended the idea to advanced functionalities, e.g., public-key encryption and attribute-based encryption. Kitagawa et al. [KNY23], and Bartusek et al. [BKM+23] replaced privately verifiable deletion certificates with publicly verifiable ones from one-way functions and one-way state generators [MY22]. Certified deletion has also been investigated for revoking cryptographic keys [KN22,AKN+23,APV23,CGJL23], digital signatures [MPY24], secret sharing [BR24], and obfuscation [BGK+24].

Moreover, the revocation idea has been considered in the case of general programs by Ananth et al. [ALP21]. However, their proposed scheme relies on post-quantum indistinguishability obfuscation (iO), where concrete realizations from standard cryptographic assumptions are not known yet. In this work, we consider the case of NIZKs, which allows constructions from weaker assumptions.

Copy-protection and unclonable primitives. Aaronson [Aar09] introduced quantum copy-protection, which enables encoding a functionality into a quantum state that cannot be cloned, and proposed a scheme for any unlearnable Boolean functions relative to a quantum oracle. Later, it was shown that any unlearnable functionality could be copy-protected relative to some classical oracle [ALL+21].

Coladangelo et al. [CMP22] constructed copy-protection for (multi-bit) point functions and compute-and-compare functions in the QROM. Copy-protection for decryption schemes and pseudorandom functions has been realized from iO, compute-and-compare obfuscation for the class of unpredictable distributions, and one-way functions [CLLZ21]. Liu et al. [LLQZ22] constructed bounded collusion-resistant copy-protection for several functionalities, such as decryption, signatures, and pseudorandom functions from iO and the LWE assumption.

Goyal et al. [GMR23], and Jawale et al. [JK23] studied copy-protected NIZKs, also referred to as unclonable NIZKs, and showed that this notion is equivalent to public-key quantum money [AC12], which is currently only known under the iO assumption. In contrast, we consider here a weaker notion of NIZK-CD. This relaxation allows us to rely on standard assumptions and, furthermore, achieve publicly verifiable deletion or solely use classical communication channels.

1.3 Paper Organization

Section 2 provides a high-level overview of our results and techniques. Section 3 covers preliminaries and background on cryptographic primitives and lemmas. We present formal definition of NIZK-CD in Section 4. In Section 5, we propose NIZK-CD from one-way functions. Section 6 presents the extensions from LWE. Section 7 defines revocable signature of knowledge and revocable anonymous credentials and further presents their candidate constructions from NIZK-CD.

2 Technical Overview

In this section, we give a high-level overview of our techniques.

2.1 Definitions

We define NIZK-CD, a tuple of algorithms $\langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify}, \mathsf{Delete}, \mathsf{Certify} \rangle$. The first three algorithms, Setup, Prove, and Verify, are defined similarly to the standard NIZK. In particular, Setup is used to generate the common reference string (CRS) and potentially its corresponding trapdoor. Prove and Verify are used to generate and verify a quantum NIZK proof π, respectively. The only difference is that Prove additionally outputs a classical certification key ck. We can later exploit this key to validate the deletion certificate. Delete is executed on the proof π and collapses it into a classical deletion certificate cert. We can validate deletion by running a classical algorithm Certify on input cert and ck. We note that the certification key ck can be a public or private key. Moreover, the proving algorithm Prove might be an interactive algorithm between the prover and verifier, where it remotely prepares the NIZK proof π in the verifier's device.

The primitive satisfies the basic security requirements of a standard NIZK, i.e., completeness, computational soundness, and computational zero knowledge. One can naturally strengthen single-theorem zero knowledge to a multi-theorem variant [FLS90] and also soundness to simulation extractibility [Sah99,SCO+01].

We can achieve the security goal of deletion by two definitions. The first, followed by simplicity, relies on the concept of efficiently samplable (quantumly) hard distributions over NP instance-witness pairs. Consider $(\mathcal{X}, \mathcal{W})$ as such a distribution where $\mathcal{X}$ and $\mathcal{W}$ correspond to instance and witness, respectively. Let $(x, w) \leftarrow (\mathcal{X}, \mathcal{W})$ be a hard instance-witness pair, and π be a NIZK-CD proof generated on inputs x and w. We require that no efficient quantum adversary given the instance x and the proof π can return both a proof π^* and a certificate cert such that π^* passes Verify and cert passes $\mathsf{Certify}$ algorithms successfully.

However, this definition does not capture the case of the adversary receiving more than one proofs for potentially different instances, and it also does not support any guarantees against malleability attacks. To address this limitation, we strengthen the definition to ensure that from any adversary that returns both an accepting proof and a valid certificate, one can extract the witness, even if the adversary is given oracle access to prover and can query it to various instances.

2.2 Construction from OWF

We propose a generic compiler to transform any classical post-quantum NIZK for NP to a NIZK-CD construction. Here, the only additional assumption we would need is the existence of a classical post-quantum non-interactive bit-commitment scheme. Such commitment scheme can be realized from post-quantum one-way functions in the common reference string model [Nao91,DCIO98,HILL99].

In more details, let $c \leftarrow \mathsf{Com}(m, r)$ be the commitment function for a message m under randomness r. Given an NP relation R, we generate a NIZK-CD proof for an instance-witness pair $(x, w) \in R$ as follows. The prover samples random strings r_0 and r_1 and generates commitments $c_b = \mathsf{Com}(b, r_b)$ for all $b \in \{0, 1\}$. The prover produces a classical NIZK σ of $(x, w) \in R \bigvee \wedge_b c_b = \mathsf{Com}(1 - b, r)$. Given σ, c_0, and c_1, one can validate whether the instance x is satisfied if they have at least one r_b, which can invalidate the commitment part of the statement.

The prover generates the state $|\psi\rangle := |0\rangle|r_0\rangle + (-1)^u|1\rangle|r_1\rangle$ for a uniformly sampled bit u and sends the NIZK-CD proof $\pi := (|\psi\rangle, \sigma, \{c_b\}_{b\in\{0,1\}})$. Then, let $\mathsf{ck} := (u, r_0, r_1)$ be the certification key, and $U^{\mathsf{com}}_{c_0,c_1}$ be a unitary operation defined as

$$|b\rangle|r\rangle|0\rangle \xrightarrow{U^{\mathsf{com}}_{c_0,c_1}} |b\rangle|r\rangle|\mathsf{Cmt} - \mathsf{Cmp}(b, r, c_b)\rangle,$$

where $\mathsf{Cmt} - \mathsf{Cmp}$ is a commit-and-compare function that commits to b using randomness r, returns 1 if the commitment equals to c_b, and returns 0 otherwise. The recipient can verify the proof by applying $U^{\mathsf{com}}_{c_0,c_1}$ to the state $|\psi\rangle$ and measure the commit-and-compare register. If the measured result is one, i.e., there exists at least one r_b such that $c_b = \mathsf{Com}(b, r_b)$; it suffices to infer $\exists w : (x, w) \in R$, if σ also passes the verification algorithm of the classical NIZK. Moreover, for any honestly generated proof, the commit-and-compare register is always measured to 1, and the post-measurement state remains the same as $|\psi\rangle$. Therefore, π can be reused for arbitrary times. Deletion can be done by a Hadamard measurement, yielding a string d where $d \cdot (r_0 \oplus r_1) = u$. As the state collapses, r_0 and r_1 are lost. The certificate is $\mathsf{cert} := d$, and the prover validates it using $\mathsf{ck} = (u, r_0, r_1)$.

More precisely, deletion security follows the adaptive hardcore bit-property of one-way functions, showed by a recent prior works [MPY24]. This property states that given any one-way function f, no efficient adversary on input $y_0 = f(r_0)$ and $y_1 = f(r_1)$ and a superposition $|0\rangle|r_0\rangle + (-1)^u|1\rangle|r_1\rangle$ can output both a preimages r where $f(r) \in \{y_0, y_1\}$ and an string d where $d \cdot (r_0 \oplus r_1) = u$ with an advantage more than $1/2$. Since the commitment scheme satisfies hiding and binding, one can view Com as a one-way function, and the adaptive hardcore bit property holds for the commitment function. This implies that the recipient of a NIZK-CD proof cannot output both an accepting deletion certificate cert and a proof π^*; due to the definition of $U^{\mathsf{com}}_{c_0,c_1}$, measuring the quantum state included in π^* in the computational basis should yield a valid commitment randomness r_b for $c_b = \mathsf{Com}(b, r_b)$, and this contradicts the adaptive hardcore bit property. The adversary's advantage can be reduced to negligible via parallel repetition. In particular, given λ as the security parameter and $n = \omega(\lambda)$, for all $i \in [n]$, the prover samples uniform random bits u_i and strings $r_{i,0}$ and $r_{i,1}$. The then prover generates the classical NIZK proof σ for the following amplified statement.

$$(x, w) \in R \quad \bigwedge_{i\in[n]} \quad \vee_{b\in\{0,1\}}\, c_{i,b} = \mathsf{Com}(1-b, r_{i,b}),$$

where for all $i \in [n]$ and $b \in \{0, 1\}$, the commitments are indeed $c_{i,b} = \mathsf{Com}(b, r_{i,b})$. The quantum state is also amplified to $|\psi\rangle = \bigotimes_{i\in[n]}|0\rangle|r_{i,0}\rangle + (-1)^{u_i}|1\rangle|r_{i,1}\rangle$. For verification, the verifier applies the commitment function in superposition as before. For deletion, Hadamard measurements yields a set of strings $\{d_i\}_{i\in[n]}$ where for all $i \in [n]$, we have $d_i \cdot (r_{i,0} \oplus r_{i,1}) = u_i$. Completeness, zero knowledge, and simulation extractability of NIZK-CD are borrowed from the NIZK proof σ.

2.3 Constructions from LWE

The primary construction we outlined above suffers from two limitations. The deletion certificate $\mathsf{cert} = \{d_i\}_{i\in[n]}$ are only privately verifiable, and the prover cannot reveal the certification key $\mathsf{ck} = \{(u_i, r_{i,0}, r_{i,1})\}_{i\in[n]}$ publicly. Moreover, both the prover and verifier algorithms are required to be quantum. We address each problem by presenting an extended construction from LWE assumption.

Public verifiability. We can achieve publicly verifiable deletion certificates by instantiating a certificate validation oracle via compute-and-compare obfuscation which can be realized from LWE assumption [WZ17]. Given a function P, a target value lock, and a message z, compute-and-compare program is defined as

$$\mathsf{CC}[P, \mathsf{lock}, z](x) = \begin{cases} z, & P(x) = \mathsf{lock}, \\ \bot, & \text{Otherwise.} \end{cases}$$

A compute-and-compare obfuscator $\mathsf{CC.Obf}$ takes a program of this form and outputs an obfuscated program $\tilde{P}$. The security of the primitive ensures that if given access to P, no computationally-bounded adversary can find the target value lock, then $\tilde{P}$ and P are indistinguishable, i.e., $\tilde{P}$ hides all details of P.

We observe that the certificate validation algorithm $\mathsf{Certify}$ in our primary construction can be viewed as a compute-and-compare program. In particular, consider a function I with hard-coded strings $\{r_{i,0}, r_{i,1}\}_{i\in[n]}$. On input $\{d_i\}_{i\in[n]}$, the function I computes $w_i = d_i \cdot (r_{i,0} \oplus r_{i,1})$ for all $i \in [n]$ and outputs a string $w = w_0 w_1 \dots w_n$. A certificate $\mathsf{cert} = \{d_i\}_{i\in[n]}$ is valid if given as input to I, the output equals a target value $\mathsf{lock} = u_0 u_1 \dots u_n$. Since each bit u_i is sampled uniformly, lock is unpredictable. Thus, we can securely obfuscate I, denoted by $\tilde{I}$, and reveal $\tilde{I}$ publicly without comprising security of commitment randomnesses $\{r_{i,0}, r_{i,1}\}_{i\in[n]}$. Anyone having access to $\tilde{I}$ can validate the deletion certificate. We note that a recent work has applied a similar technique of using obfuscation to achieve publicly verifiable deletion in the context of secret sharing [KS24].

Classical Communication. One more observation is that the prover only needs quantumness to create the state $|\psi\rangle$ and send it to the recipient. Then, other components of the proof generation and deletion processes solely rely on classical computation and communication. Here, we present an extension of our primary construction that allows a classical prover to securely and remotely prepare $|\psi\rangle$ in the recipient's device while the recipient does not learn any information about the strings r_0, r_1. Then, the rest of the protocol remains as discussed before.

We first recall the notion of *noisy trapdoor claw-free (NTCF)* functions. Given a fixed key k, a pair of functions $f_{k,0}, f_{k,1} : \mathcal{X} \to \mathcal{Y}$ of a NTCF family satisfy the following properties. $f_{k,0}$ and $f_{k,1}$ share the same range. It is computationally hard to find a claw, i.e., a pair (r_0, r_1) where $f_{k,0}(r_0) = f_{k,1}(r_1)$. There exists a trapdoor td that allows to efficiently find two preimages r_0 and r_1 of any image y, i.e., for all $b \in \{0,1\}$ we have $f_{k,b}(r_b) = y$. Moreover, the adaptive hardcore bit property holds, i.e., given y and $|r_0\rangle + |r_1\rangle$, where r_0 and r_1 are preimages of y, no adversary can return both one of the preimages and a string d such that $d \cdot (r_0 \oplus r_1) = 0$. The NTCF family can be constructed under LWE [BCM+18].

Assume that the key k for NTCF functions and its trapdoor td are sampled by the prover. The recipient generates the following state $|\phi\rangle$.

$$|\phi\rangle := \sum_{r\in\mathcal{X}} |0\rangle|r\rangle \ + \ |1\rangle|r\rangle$$

Let $U_{f_{k,0},f_{k,1}}$ be a unitary operation defined as

$$|b\rangle|r\rangle|0\rangle \xrightarrow{U_{f_{k,0},f_{k,1}}} |b\rangle|r\rangle|f_{k,b}(r)\rangle.$$

The recipient applies $U_{f_{k,0},f_{k,1}}$ to $|\phi\rangle$. Then, they measure the register of $f_{k,b}(r)$, and it yields an image y and a post-measurement state $|0\rangle|r_0\rangle + |1\rangle|r_1\rangle$, where (r_0, r_1) form a claw for the image y. We use this as $|\psi\rangle$ for our NIZK-CD, where u is fixed to 0. The recipient sends y to the prover, who can recover the claw (r_0, r_1) using td. Fixed u does not cause any vulnerabilities, and the security is ensured from the adaptive hard-code bit property of NTCFs. However, we cannot then apply obfuscation to achieve public verifiability since the target value is fixed. We leave it as an open question whether we can have a NIZK-CD construction with both publicly verifiable deletion certificates and classical communication.

2.4 Revocable Anonymous Credentials

Signature of knowledge is a specific class of digital signatures, where messages are signed with respect to an NP instance using the corresponding witness as signing key. It requires that if an adversary, given a signature to a message m with respect to an instance x, can output two signatures for m with respect to the same instance x, they *must know* the witness. A revocable signature of knowledge enables the recipient of the signature to delete the signature after being verified. This primitive can be constructed from simulation extractable NIZK-CD, where it just suffices to attach the message m to the proved statement. One can apply any revocable signature of knowledge to build revocable anonymous credentials, where the signed messages represent access tokens. A prior work by Jawale et al. [JK23] constructed revocable anonymous credentials from unclonable NIZKs. Despite our scheme, their solution relies on non-standard assumptions such as post-quantum iO, and further, it inherently requires quantum communication.

3 Preliminaries

We use λ to denote the security parameter. We use negl as a generic negligible function. For a set S, we use $x \leftarrow S$ to indicate that x is sampled uniformly from S. We define $[n] := \{0, 1, \ldots, n-1\}$. The term PPT stands for probabilistic polynomial time, and similarly QPT stands for quantum polynomial time.

Quantum conventions. A register X is a Hilbert space $\mathbb{C}^{2^n}$. An n-qubit pure state on register X is a unit vector $|\psi\rangle \in \mathbb{C}^{2^n}$. A mixed state on register X, described by a density matrix $\rho \in \mathbb{C}^{2^n \times 2^n}$, is a positive semi-definite Hermitian operator with trace 1. Also, a quantum operation F is a completely positive trace-preserving map from a register X to a register Y, i.e., on input a density matrix ρ on register X, the operation F returns $F(\rho)$ on register Y. A unitary operation $U : X \to X$ is a quantum operation that satisfies $U^\dagger U = UU^\dagger = I^X$, where I^X is identity. A projector Π is a Hermitian operator such that $\Pi^2 = \Pi$. A projective measurement is a collection of projectors $\{\Pi_i\}_i$ with $\sum_i \Pi_i = I$.

Densities and distances. Let $\mathcal{X}$ be a finite domain. A density on $\mathcal{X}$ is a function $f : \mathcal{X} \to [0, 1]$ such that $\sum_{x \in \mathcal{X}} f(x) = 1$. $\mathcal{D}_\mathcal{X}$ denotes the set of densities on $\mathcal{X}$. For any $f \in \mathcal{D}_\mathcal{X}$, $\mathsf{Supp}(f) = \{x \in \mathcal{X} : f(x) > 0\}$. Given two densities f_0 and f_1 on $\mathcal{X}$, the Hellinger distance is defined as follows.

$$H^2(f_0, f_1) := 1 - \sum_{x \in \mathcal{X}} \sqrt{f_0(x) f_1(x)}$$

For two density matrices ρ and σ, their trace distance is defined as follows.

$$\|\rho - \sigma\|_{\mathsf{tr}} = \frac{1}{2}\|\rho - \sigma\|_1 = \frac{1}{2}\mathsf{Tr}\left[\sqrt{(\rho - \sigma)^2}\right],$$

where $\|\cdot\|_1$ is the trace norm.

Lemma 3.1. *Let $\mathcal{X}$ be a finite set, $f_0, f_1 \in \mathcal{D}_{\mathcal{X}}$, and $|\psi_b\rangle := \sum_{x\in\mathcal{X}} \sqrt{f_b(x)}\,|x\rangle$ for $b \in \{0,1\}$. We have*

$$\||\psi_0\rangle\langle\psi_0| - |\psi_1\rangle\langle\psi_1|\|_{\mathsf{tr}} = \sqrt{1-(1-H^2(f_0,f_1))^2}.$$

Theorem 3.1. ***(Holevo-Helstrom)*** *[Hel69,Hol73] Consider the experiment in which one of two quantum states ρ and σ is sent to a distinguisher, each with probability $\frac{1}{2}$. The advantage of any distinguisher that can correctly determine which state was sent is at most $\frac{1}{2} + \frac{1}{2}\|\rho-\sigma\|_{\mathsf{tr}}$.*

3.1 The Learning-with-Errors Hardness Assumption

We recall the definition of the learning-with-errors (LWE) problem. For positive real B and integer q, the truncated discrete Gaussian distribution over $\mathbb{Z}_q$ with parameter B is a distribution on $\{x \in \mathbb{Z}_q : \|x\| \leq B\}$ with a density as follows.

$$D_{\mathbb{Z}_q,B}(x) := \frac{e^{\frac{-\pi\|x\|^2}{B}}}{\sum_{x'\in\mathbb{Z}_q,\|x'\|\leq B} e^{\frac{-\pi\|x'\|^2}{B}}}.$$

For some higher dimension d, the truncated discrete Gaussian distribution over $\mathbb{Z}_q^d$ with parameter B is a distribution on $\{x \in \mathbb{Z}_q^d : \|x\| \leq B\sqrt{d}\}$ with the density

$$\forall x = (x_1, x_2, \ldots, x_d) \in \mathbb{Z}_q^d : \quad D_{\mathbb{Z}_q^d,B}(x) = D_{\mathbb{Z}_q,B}(x_1), D_{\mathbb{Z}_q,B}(x_2), \ldots, D_{\mathbb{Z}_q,B}(x_d).$$

We then define LWE that underlies several hardness assumptions in this paper.

Definition 3.1. ***(LWE)*** *Let $n(\lambda), m(\lambda), q(\lambda)$ be polynomials in λ. Moreover, let $\mathcal{X} = \mathcal{X}(\lambda)$ be a distribution over $\mathbb{Z}$. The $\mathsf{LWE}_{n,q,\mathcal{X}}$ problem is to distinguish between the distributions $(A, As+e)$ and (A, Au), where $A \leftarrow \mathbb{Z}_q^{n\times m}$, $s \leftarrow \mathbb{Z}_q^m$, $e \leftarrow \mathcal{X}^m$, $u \leftarrow \mathbb{Z}_q^m$, such that m is, at most, polynomial in $n\log q$.*

We assume no QPT algorithm can solve $\mathsf{LWE}_{n,q,\mathcal{X}}$ with some non-negligible advantage in λ, even when given access to a quantum polynomial-size advice state depending on the parameters n, m, q, and $\mathcal{X}$ of the problem. We refer to this assumption as "the $\mathsf{LWE}_{n,q,\mathcal{X}}$ assumption." It can be shown [Reg09,PRSD17] that for any $\alpha > 0$ such that $\sigma = \alpha q \geq 2\sqrt{n}$, $\mathsf{LWE}_{n,q,D_{\mathbb{Z}_{q,\sigma}}}$ is at least as hard as the shortest independent vector problem to within a factor of $\tilde{O}(n/\alpha)$, where $\tilde{O}$ hides logarithmic factors, in the *worst case* dimension n in lattices. The best-known algorithm to solve the problem runs in time $2^{\tilde{O}(n/log\gamma)}$. We assume the hardness against polynomial-time quantum adversaries where γ is super-polynomial in n. We recall two additional properties of the LWE problem. The first shows that it is possible to generate LWE samples $(A, As+e)$ such that there is a trapdoor that can recover s from the samples. We state this in the following Theorem.

Theorem 3.2. *[MP12] Let $n, m \geq 1$ and $q \geq 2$ be such that $q = \Omega(n\log q)$. There is an efficient randomized algorithm $\mathsf{Gen}(1^n, 1^m, q)$ that returns a matrix*

$A \in \mathbb{Z}_q^{n \times m}$ and a trapdoor td such that the distribution of A is negligibly close to uniform. There is an efficient deterministic algorithm Inv such that on input A, td, and a sample $As + e$ where $\|e\| \leq q/(c\sqrt{n \log q})$ and c is a universal constant outputs vectors s and e with a high overwhelming probability.

The second property is the existence of a "lossy mode" for LWE.

Theorem 3.3. *[AKPW13] Let $\mathcal{X} = \mathcal{X}(\lambda)$ be efficiently sampleable distribution over $\mathbb{Z}_q$. Define a lossy sampler $\tilde{A} \leftarrow \mathsf{LSY}(1^n, 1^m, 1^\ell, q, x)$ by $\tilde{A} = BC + F$, where $B \leftarrow \mathbb{Z}_q^{m \times \ell}$, $C \leftarrow \mathbb{Z}_q^{\ell \times m}$, $F \leftarrow \mathbb{Z}_q^{n \times m}$. Under $\mathsf{LWE}_{\ell,q,\mathcal{X}}$ assumption, the distribution of $\tilde{A}$ is computationally indistinguishable from uniform $\tilde{A} \leftarrow \mathbb{Z}_q^{m \times n}$.*

3.2 Noisy Trapdoor Claw-Free Hash Function Families

We recall the notion of noisy trapdoor claw-free (NTCF) hash function families introduced by [BCM+18]. Given two finite sets $\mathcal{X}$ and $\mathcal{Y}$, a trapdoor claw-free family of functions satisfies the following properties. For each public key k, there exists two functions $\{f_{k,b} : \mathcal{X} \to \mathcal{Y}\}_{b \in \{0,1\}}$ that both are injective and have the same range and are invertible given a trapdoor td, i.e., on input td and an image $y \in \mathcal{Y}$ it is feasible to efficiently output $x_0, x_1 \in \mathcal{X}$ such that $f_0(x_0) = f_1(x_1) = y$. Furthermore, the pair of functions should be claw-free, i.e., it is computationally hard for an attacker to find two preimages x_0, x_1 such that $f_0(x_0) = f_1(x_1)$. The functions should also satisfy an adaptive hardcore bit property, which states it is computationally hard for an attacker to generate a non-trivial tuple (b, d, x_b) such that with a non-negligible probability more than $\frac{1}{2}$ the equation $d \cdot (x_0 \oplus x_1) = 0$ is satisfied, where x_{1-b} is a unique element such that $f_{1-b}(x_{1-b}) = f_b(x_b)$.

Unfortunately, we are not aware of any exact constructions of the trapdoor claw-free functions. Instead, Brakerski et al. [BCM+18] proposed a construction for noisy trapdoor claw-free functions, which relaxes the requirements in the following way. First, the range of functions is not $\mathcal{Y}$, but instead $\mathcal{D}_\mathcal{Y}$, the set of probability densities over $\mathcal{Y}$. The trapdoor injective pair property is then defined according to the support of the output densities. The use of densities as the output of the functions requires considering additional requirements. In this paper, we need a quantum polynomial-time algorithm that efficiently prepares a superposition over the range of the function, i.e., given a function key k and a bit $b \in \{0, 1\}$, the algorithm can prepare the following quantum state.

$$\frac{1}{\sqrt{\mathcal{X}}} \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} \sqrt{f_{k,b}(x)y} |x\rangle |y\rangle$$

The construction proposed in [BCM+18] is unable to exactly produce the above state; however, it is possible to create a superposition with coefficients that $f_{k,b}(x)$ is approximated by another function $f'_{k,b}(x)$ and the resulting state is within negligible trace distance of the desired state. $f'_{k,b}(x)$ supports membership checks efficiently without the need for the trapdoor, and the inversion algorithm

should operate properly on the images in the support of $f'_{k,b}(x)$. The adaptive hardcore bit property needs to also be slightly modified. The set $\mathcal{X}$ might not be a subset of binary strings. We first assume the existence of an injective, efficiently invertible map $J : \mathcal{X} \to \{0,1\}^w$ and consider the adaptive hardcore bit property to hold for a subset of all nonzero string. Below, we present the definition.

Definition 3.2. ***(NTCF)*** *[BCM+18] $\mathcal{F} := \{f_{k,b} : \mathcal{X} \to \mathcal{D}_\mathcal{Y}\}_{k\in\mathcal{K},b\in\{0,1\}}$ is a NTCF hash function family if the following properties are satisfied.*

Key generation: *A* PPT*, $\mathsf{NTCF.Gen}_\mathcal{F}$, samples a key $k \in \mathcal{K}$ and a trapdoor* td*.*

Trapdoor Injective Pair: *For all $k \in \mathcal{K}$, $b \in \{0,1\}$, distinct $x, x' \in \mathcal{X}$, $\mathsf{Supp}(f_{k,b}(x)) \cap \mathsf{Supp}(f_{k,b}(x')) = \emptyset$. An efficient deterministic algorithm $\mathsf{Inv}_\mathcal{F}$ exists such that for all $k \in \mathcal{K}$, $b \in \{0,1\}$, $x \in \mathcal{X}$, and $y \in \mathsf{Supp}(f_{k,b}(x))$, it holds that $\mathsf{Inv}(td, b, y) = x$. Moreover, given a key $k \in \mathcal{K}$, there exists a perfect matching $\mathcal{R}_k \subseteq \mathcal{X} \times \mathcal{X}$ such that $f_{k,0}(x_0) = f_{k,1}(x_1)$ if and only if $(x_0, x_1) \in \mathcal{R}_k$.*

Range Superposition: *For all $k \in \mathcal{K}$, $b \in \{0,1\}$, $f'_{k,b} : \mathcal{X} \to \mathcal{D}_\mathcal{Y}$ exists where:*

- *For any claw $(x_0, x_1) \in \mathcal{R}_k$ with image $y \in \mathsf{Supp}(f'_{k,b}(x_b))$ it holds that $\mathsf{Inv}_\mathcal{F}(\mathsf{td}, b, y) = x_b$ as well as $\mathsf{Inv}_\mathcal{F}(\mathsf{td}, b \oplus 1, y) = x_{b\oplus 1}$.*
- *There exists an efficient deterministic algorithm $\mathsf{Chk}_\mathcal{F}$ where on input $k \in \mathcal{K}$, $b \in \{0,1\}$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$, it outputs 1 if $y \in \mathsf{Supp}(f'_{k,b}(x))$ and 0 otherwise.*
- *For all $k \in \mathcal{K}$, $b \in \{0,1\}$, we have $\mathbb{E}_{x\leftarrow X}[H^2(f_{k,b}(x), f'_{k,b}(x))] \leq \mathsf{negl}(\lambda)$. In addition, there exists a* QPT *algorithm $\mathsf{Samp}_\mathcal{F}$ such that on input $k \in \mathcal{K}$ and $b \in \{0,1\}$ outputs the following quantum state.*

$$|\psi'\rangle = \frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x\in\mathcal{X},y\in\mathcal{Y}} \sqrt{(f'_{k,b}(x))(y)}|x\rangle|y\rangle.$$

Given $|\psi\rangle = \frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x\in\mathcal{X},y\in\mathcal{Y}} \sqrt{(f_{k,b}(x))(y)}|x\rangle|y\rangle$, Lemma 3.1 implies that

$$\||\psi\rangle\langle\psi| - |\psi'\rangle\langle\psi'|\|_{\mathsf{tr}} \leq \mathsf{negl}(\lambda).$$

Adaptive Hardcore Bit: *For all keys $k \in \mathcal{K}$ and a polynomial $\ell(\lambda)$:*

- *For all $b \in \{0,1\}$, $x \in \mathcal{X}$, there exists a subset of strings $\mathcal{G}_{k,b,x} \subseteq \{0,1\}^{\ell(\lambda)}$ such that $\Pr_{d\leftarrow\{0,1\}^{\ell(\lambda)}}[d \notin \mathcal{G}_{k,b,x}] \leq \mathsf{negl}(\lambda)$. Moreover, there exists a* PPT *algorithm to efficiently check membership in $\mathcal{G}_{k,b,x}$ given k, b, x and* td*.*
- *There exists an efficiently computable injection $J : \mathcal{X} \to \{0,1\}^{\ell(\lambda)}$ such that J can also be efficiently inverted on its range, and the following holds. Let*

$$H_k := \{(b, x_b, d, d \cdot (J(x_0) \oplus J(x_1))|(x_0, x_1) \in R_k, d \in \mathcal{G}_{k,0,x} \cap \mathcal{G}_{k,1,x}\},$$
$$\bar{H}_k := \{(b, x_b, d, u \oplus 1)|(b, x_b, d, u) \in H_k\}.$$

For any QPT *algorithm $\mathcal{A}$ it holds that*

$$\left|\Pr_{\mathsf{td},k\leftarrow\mathsf{NTCF.Gen}(1^\lambda)}[A(k) \in H_k] - \Pr_{\mathsf{td},k\leftarrow\mathsf{NTCF.Gen}(1^\lambda)}[A(k) \in \bar{H}_k]\right| \leq \mathsf{negl}(\lambda).$$

Theorem 3.4. *[BCM+18] Assuming the polynomial-time quantum hardness of LWE, there exists an* NTCF *hash function family.*

One can consider an amplified adaptive hardcore bit property such that the adversary cannot return a set $\{(b_i, x_{i,b_i}, d_i, u_i)\}_{i\in[n]}$ where n is polynomial in the security parameter λ, and each tuple satisfies $d_i \cdot (x_{i,b_i} \oplus x_{i,1-b_i}) = u_i$, such that $(x_{i,b_i}, x_{i,1-b_i})$ is a claw. The property is formally defined as follows.

Definition 3.3. ***(Amplified Adaptive Hardcore Bit)*** *An* NTCF *function family $\mathcal{F}$ satisfies the amplified adaptive hardcore bit property if, for any* QPT *algorithm $\mathcal{A}$ and a polynomial $n = \ell(\lambda)$, the following is at most* $\mathsf{negl}(\lambda)$.

$$\Pr\left[\begin{array}{c} \forall i \in [n] : (k_i, \mathsf{td}_i) \leftarrow \mathsf{NTCF.Gen}(1^\lambda) \\ \{(b_i, x_{i,b_i}, d_i, u_i)\}_{i\in[n]} \leftarrow \mathcal{A}(\{k_i\}_{i\in[n]}) \\ \forall \beta \in \{0,1\} : x_{i,\beta} = \mathsf{Inv}(\mathsf{td}_i, \beta, y_i) \end{array} : \begin{array}{c} \forall i \in [n] : d_i \in \mathcal{G}_{k,0,x} \cap \mathcal{G}_{k,1,x} \\ \wedge \\ u_i = d_i \cdot (J(x_{i,0}) \oplus J(x_{i,1})) \end{array}\right]$$

Theorem 3.5. *[RS19,KNY21] Any* NTCF *family of hash functions satisfies the amplified adaptive hardcore bit property.*

We also note that recently Morimae et al. [MPY24] introduced a similar notion of adaptive hardcore bit property for general OWFs, beyond those based on the LWE assumption, and showed the following result.

Theorem 3.6. *[MPY24] Let $\ell(\lambda), \kappa(\lambda), n(\lambda) \in \mathbb{N}$ be polynomials. Given any quantum-secure OWF $f : \{0,1\}^{\ell(\lambda)} \to \{0,1\}^{\kappa(\lambda)}$, for any* QPT *adversary $\mathcal{A}$,*

$$\Pr\left[\begin{array}{c} \forall i \in [n] : x_{i,0}, x_{i,1} \leftarrow \{0,1\}^{\ell(\lambda)}, u_i \leftarrow \{0,1\} \\ \{(x_i, d_i)\}_{i\in[n]} \leftarrow \mathcal{A}(\otimes_{i\in[n]} \frac{|x_{i,0}\rangle + (-1)^{u_i}|x_{i,1}\rangle}{\sqrt{2}}, \{f(x_{i,b})\}_{i,b}) \end{array} : \begin{array}{c} \wedge_{i\in[n]} f(x_i) \in \{f(x_{i,0}), f(x_{i,1})\} \\ \wedge \\ \wedge_{i\in[n]} d_i \cdot (x_{i,0} \oplus x_{i,1}) = u_i \end{array}\right] \leq \mathsf{negl}(\lambda).$$

3.3 Other Useful Cryptographic Primitives and Lemmas

We recall the notion of commitment schemes, which is a central building block for our NIZK-CD constructions throughout this paper.

Definition 3.4. ***(Commitment)*** *Let $n(\lambda), \ell(\lambda), \kappa(\lambda)$ be polynomials. Λ is a post-quantum commitment scheme in the* CRS *model if it satisfies the following.*

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: *on input λ, outputs* crs *and trapdoor* td.
- $c \leftarrow \mathsf{Com}(\mathsf{crs}, m, r)$*: on input* crs*, a message $m \in \{0,1\}^{n(\lambda)}$, commitment randomness $r \in \{0,1\}^{\ell(\lambda)}$, outputs a commitment $c \in \{0,1\}^{\kappa(\lambda)}$.*

Perfectly Binding: *For every security parameter* $\lambda \in \mathbb{N}$, $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$, *and* m, m', r, r' *such that* $m \neq m'$, *it holds that* $\mathsf{Com}(\mathsf{crs}, m, r) \neq \mathsf{Com}(\mathsf{crs}, m', r')$.

Computational Hiding: *For any* QPT *distinguisher* $\mathcal{D}$ *and sufficiently large* $\lambda \in \mathbb{N}$, $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$, $m, m' \in \{0,1\}^{n(\lambda)}$, *the following is, at most,* $\mathsf{negl}(\lambda)$.

$$\left|\Pr\left[\begin{array}{c} r \leftarrow \{0,1\}^{\kappa(\lambda)} \\ c \leftarrow \mathsf{Com}(\mathsf{crs}, m, r) \end{array} : \mathcal{D}(\mathsf{crs}, c) = 1\right] - \Pr\left[\begin{array}{c} r \leftarrow \{0,1\}^{\kappa(\lambda)} \\ c' \leftarrow \mathsf{Com}(\mathsf{crs}, m', r) \end{array} : \mathcal{D}(\mathsf{crs}, c') = 1\right]\right|$$

Theorem 3.7. *[DCIO98,HILL99] Assuming post-quantum one-way functions, there exists a post-quantum, classical, non-interactive, perfectly binding, and computationally hiding bit-commitment scheme in the* CRS *model.*

Theorem 3.8. *[PS19] Assuming polynomial hardness of learning-with-errors, there exists a post-quantum, classical, non-interactive, perfectly binding, and computationally hiding commitment scheme in the* CRS *model.*

We also recall the notion of compute-and-compare obfuscation [CLLZ21], which we use for our NIZK-CD construction with publicly verifiable certificates.

Definition 3.5. ***(Compute-and-Compare Program)*** *Given some function* $P : \{0,1\}^{\ell_{\mathsf{in}}} \to \{0,1\}^{\ell_{\mathsf{out}}}$ *along with a target value* $\mathsf{lock} \in \{0,1\}^{\ell_{\mathsf{out}}}$ *and a message* $z \in \{0,1\}^{\ell_{\mathsf{msg}}}$, *we define the compute-and-compare program as follows.*

$$\mathsf{CC}[P, \mathsf{lock}, z](x) = \begin{cases} z, & P(x) = \mathsf{lock}, \\ \bot, & \text{Otherwise.} \end{cases}$$

Definition 3.6. ***(Unpredictable Distributions)*** $\mathcal{D} = \{\mathcal{D}_\lambda\}_{\lambda \in \mathbb{N}}$ *is a family of distributions where* $\mathcal{D}_\lambda$ *is a distribution over pairs of form* $(\mathsf{CC}[P, \mathsf{lock}, z], \mathsf{aux})$ *where* aux *is a quantum state.* $\mathcal{D}$ *is unpredictable if for all* QPT *algorithms* $\mathcal{A}$,

$$\Pr_{(\mathsf{CC}[P,\mathsf{lock},z],\mathsf{aux}) \leftarrow \mathcal{D}_\lambda}[\mathcal{A}(P, \mathsf{aux}) = \mathsf{lock}] \leq \mathsf{negl}(\lambda).$$

Definition 3.7. ***(Compute-and-Compare Obfuscation)*** *A* PPT *algorithm* $\mathsf{CC.Obf}$ *is an obfuscator for the class of unpredictable distributions if for any family of distributions* $\mathcal{D} = \{\mathcal{D}_\lambda\}$ *belonging to the class, the following holds.*

Functionality Preserving: *For every program* P *in the support of* $\mathcal{D}_\lambda$,

$$\Pr[\tilde{P} \leftarrow \mathsf{CC.Obf}(P) : \forall x, P(x) = \tilde{P}(x)] \geq 1 - \mathsf{negl}(\lambda).$$

Distributional Indistinguishability: *For every program* P *in the support of* $\mathcal{D}_\lambda$, *there exists an efficient simulator* Sim *such that we have*

$$(\mathsf{CC.Obf}(P), \mathsf{aux}) \approx_c (\mathsf{Sim}(P.\mathsf{params}), \mathsf{aux}),$$

where $(P, \mathsf{aux}) \leftarrow \mathcal{D}_\lambda$, *and* $P.\mathsf{params}$ *denotes the input size, output size, and circuit size of* P, *which are not required to be obfuscated.*

Theorem 3.9. *[WZ17] There exists a compute-and-compare obfuscation scheme assuming polynomial quantum hardness of learning-with-errors.*

We recall the notion of non-interactive zero-knowledge (NIZK) arguments in the CRS model. We provide two definitions of NIZK as follows. The first one is a single-theorem definition and ensures that no efficient adversary can output an accepting proof for any unsatisfied instance. The second one is a multi-theorem definition and ensures that from any efficient adversary that can output an accepting proof, one can extract a valid witness to the relation.

Definition 3.8. ***(NIZK for NP)*** *Let any* NP *relation R with a corresponding language L. $\Pi = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify} \rangle$ is a post-quantum NIZK for* NP *in the* CRS *model if it satisfies the following syntax and the security properties.*

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: *on input λ, outputs* crs *and a trapdoor* td*.*
- $\pi \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)$: *on input* crs *and a pair $(x, w) \in R$, outputs a proof π.*
- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, \pi)$: *on input* crs*, x, and π, outputs accept 1 or reject 0.*

Completeness: *For every security parameter $\lambda \in \mathbb{N}$ and $(x, w) \in R$,*

$$\Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ \pi \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)\end{array} : \mathsf{Verify}(\mathsf{crs}, x, \pi)\right] \geq 1 - \mathsf{negl}(\lambda).$$

Adaptive Computational Soundness: *For any* QPT *adversary algorithm $\mathcal{A}$ and sufficiently large security parameter λ,*

$$\Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ x, \pi \leftarrow \mathcal{A}(\mathsf{crs})\end{array} : \mathsf{Verify}(\mathsf{crs}, x, \pi) = 1 \wedge x \notin L\right] \leq \mathsf{negl}(\lambda).$$

Adaptive Computational Zero-Knowledge: *There exists a* QPT *simulator $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ where for every* QPT *adversary $\mathcal{A}$, every* QPT *distinguisher $\mathcal{D}$, and sufficiently large $\lambda \in \mathbb{N}$,*

$$\left|\Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ x, w, \xi \leftarrow \mathcal{A}(\mathsf{crs}) \\ \pi \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)\end{array} : \mathcal{D}(\mathsf{crs}, x, \pi, \xi) = 1\right] - \Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, w, \xi \leftarrow \mathcal{A}(\mathsf{crs}) \\ \pi \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x)\end{array} : \mathcal{D}(\mathsf{crs}, x, \pi, \xi) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

Theorem 3.10. *[PS19] Assuming polynomial quantum hardness of LWE, there exists a post-quantum non-interactive, adaptively computationally sound, and adaptively computationally zero-knowledge argument for NP.*

Definition 3.9. ***(Simulation-Extractable NIZK for NP)*** *Let relation R with language L be any* NP *relation. $\Pi = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify} \rangle$ is a post-quantum non-interactive simulation-extractable, and adaptive multi-theorem computational*

zero-knowledge protocol for NP in the CRS *model if it is satisfies the completeness as in Definition 3.8 and the following additional properties.*

Adaptive Multi-Theorem Computational Zero-Knowledge: *There exists* QPT *simulator* $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$*, such that for any* QPT *adversary algorithm* $\mathcal{A}$ *and sufficiently large* $\lambda \in \mathbb{N}$*,*

$$\Big|\Pr\left[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) : \mathcal{A}^{\mathsf{Prove}(\mathsf{crs},\cdot,\cdot)}(\mathsf{crs}) = 1\right] - \Pr\left[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) : \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) = 1\right]\Big| \leq \mathsf{negl}(\lambda).$$

Simulation Soundness: *Let* $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ *be the simulator given by the adaptive multi-theorem computational zero-knowledge. For every* QPT *algorithm* $\mathcal{A}$ *and sufficiently large* $\lambda \in \mathbb{N}$*,*

$$\Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, \pi \leftarrow \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs})\end{array} : \mathsf{Verify}(\mathsf{crs}, x, \pi) = 1 \wedge x \notin L \wedge x \notin Q\right] \leq \mathsf{negl}(\lambda),$$

where Q *is the list of queries from* $\mathcal{A}$ *to* Sim_1*.*

Simulation Extractability: *Let* $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ *be the simulator given by the adaptive multi-theorem computational zero-knowledge property. There exists a* QPT *extractor* Ext *such that for every* QPT *algorithm* $\mathcal{A}$ *and sufficiently large* $\lambda \in \mathbb{N}$*, the following probability is, at most, negligible.*

$$\Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, \pi \leftarrow \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, \pi)\end{array} : \mathsf{Verify}(\mathsf{crs}, x, \pi) = 1 \wedge (x, w) \notin R \wedge x \notin Q\right],$$

where Q *is the list of queries from* $\mathcal{A}$ *to* Sim_1*.*

Remark 3.1. [FLS90,Sah99,SCO+01,JK23] It is known that a multi-theorem simulation-extractable NIZK, i.e., Definition 3.9, satisfies Definition 3.8 since adaptive multi-theorem zero-knowledge implies adaptive zero-knowledge, and simulation-soundness implies adaptive computational soundness. Moreover, one can show that the simulation extractability implies simulation soundness.

Theorem 3.11. *[SCO+01,JK23] Given a quantum-secure one-way function and a post-quantum IND-CPA secure encryption scheme, any post-quantum NIZK for NP can be turned into a post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP.*

Corollary 3.1. *Assuming polynomial quantum hardness of learning-with-errors, a post-quantum non-interactive, simulation-extractable, adaptively multi-theorem computationally zero-knowledge argument for NP exists.*

Proof. This follows from Theorems 3.10 and 3.11. □

Finally, we provide the cryptographic definition of the signature of knowledge.

Definition 3.10. ***(Signature of Knowledge)*** *Let R be an* NP *relation with language L and message space $\mathcal{M}$. $\Sigma = \langle \mathsf{Setup}, \mathsf{Sign}, \mathsf{Verify} \rangle$ is a post-quantum* SimExt*-secure signature of knowledge if the below syntax and properties hold.*

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: *on input λ, outputs a* crs *and trapdoor* td*.*
- $\sigma \leftarrow \mathsf{Sign}(\mathsf{crs}, x, w, m)$: *on input* crs*, pair $(x, w) \in R$, and message $m \in \mathcal{M}$, outputs a signature of knowledge σ to the message m.*
- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, m, \sigma)$: *on input* crs*, x, m, and σ, accepts or rejects.*

Correctness: *For every $\lambda \in \mathbb{N}$, pair $(x, w) \in R$ and message $m \in \mathcal{M}$,*

$$\Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ \sigma \leftarrow \mathsf{Sign}(\mathsf{crs}, x, w, m)\end{array} : \mathsf{Verify}(\mathsf{crs}, x, m, \sigma) = 1\right] \geq 1 - \mathsf{negl}(\lambda).$$

Simulation: *There exist a* QPT *simulator algorithm $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ such that for every* QPT *algorithm $\mathcal{A}$ and sufficiently large $\lambda \in \mathbb{N}$,*

$$\left|\Pr\left[\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda) : \mathcal{A}^{\mathsf{Sign}(\mathsf{crs},\cdot,\cdot,\cdot)}(\mathsf{crs}) = 1\right] - \Pr\left[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) : \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot,\cdot)}(\mathsf{crs}) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

Extraction: *Let $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ be the simulators given by the simulation property. There exist a* QPT *extractor algorithm* Ext *such that for every* QPT *algorithm $\mathcal{A}$ and sufficiently large $\lambda \in \mathbb{N}$,*

$$\Pr\left[\begin{array}{c}\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, m, \sigma \leftarrow \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot,\cdot)}(\mathsf{crs}) : \\ w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, m, \sigma)\end{array} \begin{array}{c}\mathsf{Verify}(\mathsf{crs}, x, m, \sigma) = 1 \\ \wedge\ (x, w) \notin R\ \wedge\ (x, m) \notin Q\end{array}\right] \leq \mathsf{negl}(\lambda).$$

where Q is the list of queries from $\mathcal{A}$ to Sim_1.

Theorem 3.12. *[CL06,JK23] Given post-quantum non-interactive simulation extractable multi-theorem zero-knowledge for NP, a post-quantum* SimExt*-secure signature of knowledge exists.*

4 NIZK with Certified Deletion

In this section, we introduce the notion of a non-interactive zero-knowledge with certified deletion (NIZK-CD) and define its security properties. In particular, we provide two different definitions of NIZK-CD. The first one is motivated by simplicity, establishes a single-theorem NIZK-CD, where we guarantee that no computationally-bounded adversary receiving honestly generated proofs for hard NP instances can output both an accepting proof and a valid deletion certificate. In the second definition, we present a multi-theorem NIZK-CD with a guarantee that from any adversary, even having the oracle access to NIZK-CD simulators, returning both an accepting proof and a valid deletion certificate for a hard NP instances, one can extract a satisfying witness corresponding to the instance.

Definition 4.1. (Hard Distribution) *Given an NP relation R, an efficiently samplable distribution $(\mathcal{X}, \mathcal{W})$ over R is hard if for every* QPT *algorithm $\mathcal{A}$ and sufficiently large security parameter λ,*

$$\Pr[(x,w) \leftarrow (\mathcal{X},\mathcal{W}) : (x, \mathcal{A}(x)) \in R] \leq \mathsf{negl}(\lambda).$$

Definition 4.2. (NIZK with Certified Deletion) *Let any* NP *relation R with language L. $\Gamma = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify}, \mathsf{Delete}, \mathsf{Certify} \rangle$ is a NIZK-CD if it satisfies the following syntax and properties.*

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$*: on input λ, outputs a classical* crs *and trapdoor* td*.*
- $\pi, \mathsf{ck} \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)$*: on input* crs *and an instance-witness pair $(x,w) \in \mathcal{R}$, outputs a quantum proof π and a classical certification key* ck*.*
- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, \pi)$*: on input* crs*, x, and π, outputs accept or reject.*
- $\mathsf{cert} \leftarrow \mathsf{Delete}(\pi)$*: on input π, outputs a classical deletion certificate* cert*.*
- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$*: on input* ck *and* cert*, outputs accept or reject.*

Completeness: *For every $\lambda \in \mathbb{N}$ and every $(x,w) \in R$,*

$$\Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ \pi, \mathsf{ck} \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w) \\ \mathsf{cert} \leftarrow \mathsf{Delete}(\pi) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, \pi) = 1 \\ \wedge \\ \mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

Adaptive Computational Soundness: *For every* QPT *algorithm $\mathcal{A}$ and a sufficiently large security parameter λ,*

$$\Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ x, \pi \leftarrow \mathcal{A}(\mathsf{crs}) \end{array} : \mathsf{Verify}(\mathsf{crs}, x, \pi) = 1 \wedge x \notin L\right] \leq \mathsf{negl}(\lambda).$$

Adaptive Computational Zero-Knowledge: *There exists a* QPT *algorithm $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ such that for every* QPT *algorithm $\mathcal{A}$,* QPT *distinguisher $\mathcal{D}$, sufficiently large λ, the following is, at most,* $\mathsf{negl}(\lambda)$*.*

$$\left|\Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ x, w, \xi \leftarrow \mathcal{A}(\mathsf{crs}) \\ \pi, \mathsf{ck} \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w) \end{array} : \mathcal{D}(\mathsf{crs}, x, \pi, \xi) = 1\right] - \Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, w, \xi \leftarrow \mathcal{A}(\mathsf{crs}) \\ \pi, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x) \end{array} : \mathcal{D}(\mathsf{crs}, x, \pi, \xi) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

Deletion Security: *For every* QPT *algorithm $\mathcal{A}$, sufficiently large λ, and hard distribution $(\mathcal{X}, \mathcal{W})$ over $\mathcal{R}$,*

$$\Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) \\ (x,w) \leftarrow (\mathcal{X}, \mathcal{W}) \\ \pi, \mathsf{ck} \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w) \\ \pi^*, \mathsf{cert} \leftarrow \mathcal{A}(\mathsf{crs}, x, \pi) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, \pi^*) = 1 \\ \wedge \\ \mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \end{array}\right] \leq \mathsf{negl}(\lambda).$$

We present the definition of multi-theorem simulation-extractable NIZK-CD.

Definition 4.3. (Simulation-Extractable NIZK with Certified Deletion) *Let R be an* NP *relation with language L. $\Gamma = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify}, \mathsf{Delete}, \mathsf{Certify} \rangle$ is a non-interactive simulation-extractable, adaptive multi-theorem computational zero-knowledge with certified deletion if satisfies completeness as in Definition 4.2 and the following zero-knowledge, extraction, and deletion properties.*

Adaptive Multi-Theorem Computational Zero-Knowledge: *There exists* QPT *simulator algorithm $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ such that for every* QPT *algorithm $\mathcal{A}$ and sufficiently large $\lambda \in \mathbb{N}$,*

$$\Big| \Pr\left[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) : \mathcal{A}^{\mathsf{Prove}(\mathsf{crs},\cdot,\cdot)}(\mathsf{crs}) = 1\right] - \Pr\left[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) : \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) = 1\right] \Big| \leq \mathsf{negl}(\lambda),$$

where $\mathcal{A}$ only receives proofs from the oracles, and certifying keys are discarded.

Simulation Extractability: *Let $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ be the simulator given by the adaptive multi-theorem computational zero-knowledge property. A* QPT *extractor* Ext *exists where for any* QPT *adversary $\mathcal{A}$ and sufficiently large $\lambda \in \mathbb{N}$,*

$$\Pr\left[\begin{array}{c} (\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, \pi) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, \pi) = 1 \\ \wedge (x, w) \notin R \wedge x \notin Q \end{array}\right] \leq \mathsf{negl}(\lambda),$$

where Q is the list of queries from $\mathcal{A}$ to Sim_1.

Simulation Extractability with Deletion: *Let $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ be the simulator given by adaptive multi-theorem computational zero-knowledge. There exists a* QPT *extractor* Ext-Del *such that for every* QPT *algorithm $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ and every sufficiently large $\lambda \in \mathbb{N}$,*

$$\Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, \xi \leftarrow \mathcal{A}_0^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) \\ \pi, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x) \\ \pi^*, \mathsf{cert} \leftarrow \mathcal{A}_1^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}, \xi, x, \pi) \\ w \leftarrow \mathsf{Ext\text{-}Del}(\mathsf{crs}, \mathsf{td}, x, \pi^*, \mathsf{cert}) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, \pi^*) = 1 \\ \wedge \\ [\mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \ \vee \ (x, w) \notin R] \wedge \ x \notin Q \end{array}\right] \leq \mathsf{negl}(\lambda),$$

where Q is the list of queries, and $\mathcal{A}$ only receives the proof from the oracle Sim_1.

Remark 4.1. We remark that in NIZK-CD constructions with publicly verifiable deletion certificates, where the certification key ck is made public, the adversary, distinguisher, and extractor algorithms in the above definitions take ck as input.

Remark 4.2. In our constructions, Setup and Certify are classical while Prove, Verify, and Delete are quantum algorithms. Moreover, the Prove algorithm might be interactive between a classical prover and a quantum verifier, i.e., the prover remotely prepares required quantum states in the verifier's device.

We note that the connection between different notions of zero-knowledge, soundness, and extraction for NIZK arguments that are described in Remark 3.1 also holds for NIZK-CD. The following states that the simulation extractability with deletion implies both simulation extractibility and deletion security.

Theorem 4.1. *Simulation extractability with deletion defined in Definition 4.3 implies simulation extractibility as defined in Definition 4.3 and deletion security as defined in Definition 4.2.*

Proof. First, consider an adversary $\mathcal{B}$ that breaks simulation extractibility with a non-negligible advantage. One can build an adversary $\mathcal{A} = (A_0, A_1)$ that breaks simulation extractibility with deletion with the same advantage. In particular, A_0 runs B to get an instance x and a corresponding proof π^*. The instance x is submitted to the simulator Sim_1, and π^* is included in ξ. The algorithm A_1 receives a proof π from Sim_1, which can be deleted to generate a valid deletion certificate cert. Afterwards, A_1 can return produced cert as a valid certificate and π^* included in ξ as an accepting proof corresponding to the instance x.

Similarly, consider an adversary $\mathcal{B}$ that can break the deletion security with a non-negligible advantage. One can build an adversary $\mathcal{A} = (A_0, A_1)$ where A_0 samples a hard instance x, submitted to Sim_1. A_1 receives a proof π for the instance x from the oracle. Then, A_1 runs B on input x and π to get a valid deletion certificate and an accepting transcript where the success probability is the same as the advantage of the adversary algorithm $\mathcal{B}$. □

5 NIZK with Certified Deletion from OWF

In this section, we propose our NIZK-CD construction from one-way functions.

Theorem 5.1. *Assuming post-quantum one-way function and also post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP, there exists a non-interactive, simulation-extractable, adaptive multi-theorem zero-knowledge with certified deletion.*

Proof. Let $\Lambda = \langle \mathsf{Setup}, \mathsf{Com} \rangle$ be a post-quantum bit-commitment, which can be realized from OWFs as described in Section 3.3. Let $\Pi = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify} \rangle$ be a non-interactive, simulation-extractable, adaptive multi-theorem computational zero-knowledge. Our NIZK-CD construction is as follows.

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: Runs $\mathsf{crs}_\Lambda, \mathsf{td}_\Lambda \leftarrow \Lambda.\mathsf{Setup}(1^\lambda)$, $\mathsf{crs}_\Pi, \mathsf{td}_\Pi \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$ and outputs $\mathsf{crs} := (\mathsf{crs}_\Lambda, \mathsf{crs}_\Pi)$ and $\mathsf{td} := (\mathsf{td}_\Lambda, \mathsf{td}_\Pi)$.

- $\pi, \mathsf{ck} \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)$: Let $n = \nu(\lambda)$ be a polynomial in λ. Prover samples $r_{i,b} \leftarrow \{0,1\}^{\ell(\lambda)}$, $\forall i \in [n], b \in \{0,1\}$, computes $c_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r_{i,b})$. Given $x^* := (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$ with witness $w^* := (w, \{r_{i,b}\}_{i\in[n],b\in\{0,1\}})$, invokes $\Pi.\mathsf{Prove}(\mathsf{crs}_\Pi, x^*, w^*)$ and generates a NIZK proof σ for R^* defined as

$$(x, w) \in R \quad \bigwedge_{i\in[n]} \quad \vee_{b\in\{0,1\}}\, c_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 1-b, r_{i,b}). \tag{1}$$

  Given uniform bits $u_i \leftarrow \{0,1\}$ for all $i \in [n]$, the following state is prepared.

$$|\psi\rangle := \bigotimes_{i\in[n]} \frac{1}{\sqrt{2}} \left(|0\rangle|r_{i,0}\rangle + (-1)^{u_i}|1\rangle|r_{i,1}\rangle\right) \tag{2}$$

  It outputs $\pi := \{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$ and $\mathsf{ck} = \{u_i, r_{i,b}\}_{i\in[n],b\in\{0,1\}}$.

- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, \pi)$: The verifier parses π as $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$ and runs $v_\sigma \leftarrow \Pi.\mathsf{Verify}(\mathsf{crs}_\Pi, x^*, \sigma)$, where $x^* := (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$. Let $U^{\mathsf{com}}_{c_0,c_1}$ be a unitary operation with respect to c_0 and c_1 as follows.

$$|b\rangle|r\rangle|0\rangle \xrightarrow{U^{\mathsf{com}}_{c_0,c_1}} |b\rangle|r\rangle|\mathsf{Cmt} - \mathsf{Cmp}(b, r, c_b)\rangle, \tag{3}$$

  where the commit-and-compare function is defined as below.

$$\mathsf{Cmt} - \mathsf{Cmp}(b, r, c_b) = \begin{cases} 1, & \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r) = c_b \\ 0, & \text{Otherwise} \end{cases} \tag{4}$$

  The verifier adds ancilla qubits, and applies $U = \bigotimes_{i\in[n]} U^{\mathsf{com}}_{c_{i,0},c_{i,1}}$ to $|\psi\rangle$,

$$\begin{aligned} &\xrightarrow{U} \bigotimes_{i\in[n]} \frac{1}{\sqrt{2}}(|0\rangle|r_{i,0}\rangle|\mathsf{Cmt} - \mathsf{Cmp}(0, r_{i,0}, c_{i,0})\rangle \\ &+ (-1)^{u_i}|1\rangle|r_{i,1}\rangle|\mathsf{Cmt} - \mathsf{Cmp}(1, r_{i,1}, c_{i,1})\rangle). \end{aligned} \tag{5}$$

  Then, the verifier for all $i \in [n]$, measures the right-most registers to get bits $v_i = \mathsf{Cmt} - \mathsf{Cmp}(0, r_{i,0}, c_{i,0})$. The verification outcome is $v_{\mathsf{out}} = v_\sigma \wedge \bigwedge_{i\in[n]} v_i$, where $v_{\mathsf{out}} = 1$ indicates accept and $v_{\mathsf{out}} = 0$ indicates reject.

- $\mathsf{cert} \leftarrow \mathsf{Delete}(\pi)$: The verifier parses π as $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle = \bigotimes_{i\in[n]}|\psi_i\rangle$ such that $|\psi_i\rangle = \sum_{b\in\{0,1\}} \frac{1}{\sqrt{2}}(-1)^{b\cdot u_i}|b\rangle|r_{i,b}\rangle$. For all $i \in [n]$, it measures each state $|\psi_i\rangle$ in the Hadamard basis, yielding an outcome string $d_i \in \{0,1\}^{\ell(\lambda)}$ such that it holds $d_i \cdot (r_{i,0} \oplus r_{i,1}) = u_i$. We define $\mathsf{cert} := \{d_i\}_{i\in[n]}$.

- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$: The prover parses ck as $\{u_i, r_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and cert as $\{d_i\}_{i\in[n]}$, validates whether for all $i \in [n]$ it holds $d_i \cdot (r_{i,0} \oplus r_{i,1}) = u_i$.

Next, we prove perfect completeness, adaptive multi-theorem computational zero-knowledge, and simulation extractability with deletion, which are borrowed from completeness, multi-theorem zero-knowledge, simulation extractability of Π, as well as the post-quantum security of the commitment scheme Λ.

Completeness: The completeness property is shown with respect to both proof verification and certificate validation. Consider the former case. Following the perfect completeness of Π, we ensure that the proof σ is accepting and $v_\sigma = 1$ with a probability of 1. Then, it suffices to show that for all $i \in [n]$, $v_i = 1$. As the commitment algorithm $\Lambda.\mathsf{Com}$ is deterministic, and the commitments are honestly generated by the prover, i.e., $c_i = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b_i, r_{i,b})$, we ensure that for all $i \in [n]$ and $b \in \{0,1\}$ we have $\mathsf{Cmt} - \mathsf{Cmp}(b_i, r_{i,b}, c_{i,b}) = 1$. Therefore, the bit v_i is measured to one with a probability of 1. Second, consider completeness with respect to deletion. We parse the sate $|\psi\rangle$ as $\bigotimes_{i\in[n]}|\psi_i\rangle$, such that we have $|\psi_i\rangle = \sum_{b\in\{0,1\}} \frac{1}{\sqrt{2}}(-1)^{b\cdot u_i}|b\rangle|r_{i,b}\rangle$. The verifier measures each $|\psi_i\rangle$ in the Hadamard basis, which yields an outcome string $d_i \in \{0,1\}^{\ell(\lambda)}$, and it holds that $u_i = d_i \cdot (r_{i,0} \oplus r_{i,1})$. Hence, the $\mathsf{Certify}$ algorithm always accepts cert.

Adaptive Multi-Theorem Computational Zero-Knowledge: Consider the $\Pi.Sim = (\Pi.Sim_0, \Pi.\mathsf{Sim}_1)$ as the simulators of Π. We show $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ for adaptive multi-theorem computational zero-knowledge of our construction.

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda)$: Runs $\mathsf{crs}_\Lambda, \mathsf{td}_\Lambda \leftarrow \Lambda.\mathsf{Setup}(1^\lambda)$, $\mathsf{crs}_\Pi, \mathsf{td}_\Pi \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$ and outputs $\mathsf{crs} := (\mathsf{crs}_\Lambda, \mathsf{crs}_\Pi)$ and $\mathsf{td} := (\mathsf{td}_\Lambda, \mathsf{td}_\Pi)$.
- $\pi, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x)$: Samples randomnesses $r'_{i,b} \leftarrow \{0,1\}^{\ell(\lambda)}$ and computes $c'_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r'_{i,b})$ the $\forall i \in [n], b \in \{0,1\}$. $x^* := (x, \{c'_{i,b}\}_{i\in[n],b\in\{0,1\}})$. The the algorithm query $\Pi.\mathsf{Sim}_1$ on input x^* to get the simulated NIZK σ'. Then, it samples bits $u'_i \leftarrow \{0,1\}$ and prepares $|\psi'\rangle$ as defined in Equation 2. It outputs $\pi := \{c'_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma', |\psi'\rangle$ and $\mathsf{ck} = \{u'_i, r'_{i,b}\}_{i\in[n],b\in\{0,1\}}$.

Reduction: Suppose that there exists a QPT adversary algorithm $\mathcal{A}$ such that for some polynomial $p(\lambda)$,

$$\Big|\Pr\Big[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda) : \mathcal{A}^{\mathsf{Prove}(\mathsf{crs},\cdot,\cdot)}(\mathsf{crs}) = 1\Big] - \Pr\Big[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) : \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) = 1\Big]\Big| \geq \frac{1}{p(\lambda)}.$$

We construct a QPT adversary $\mathcal{B}$ for the adaptive multi-theorem computational zero-knowledge property of the underlying NIZK Π as follows.

1. Receives crs from $\Pi.\mathsf{Setup}$ and $\Lambda.\mathsf{Setup}$ or $\Pi.\mathsf{Sim}_0$ and sends it to $\mathcal{A}$.
2. On query (x, w), samples $r_{i,b} \leftarrow \{0,1\}^{\ell(\lambda)}$, gets $c_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r_{i,b})$ for all $i \in [n]$ and $b \in \{0,1\}$. Then, given instance $x^* := (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$ and witness $w^* := (w, \{r_{i,b}\}_{i\in[n],b\in\{0,1\}})$ receive the real or simulated proof σ by query either Prove on input (x^*, w^*) or $\Pi.\mathsf{Sim}_1$ on input x^*. Moreover, samples bits $u_i \leftarrow \{0,1\}$ and prepares $|\psi\rangle$ as defined in Equation 2. The proof $\pi = \{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$ is sent to the adversary algorithm $\mathcal{A}$.
3. Output the outcome of $\mathcal{A}$.

As commitment randomnesses $\{r_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and exponents $\{u_i\}_{i\in[n],b\in\{0,1\}}$ are uniformly sampled, the real and simulated quantum states, i.e., $|\psi\rangle$ and $|\psi'\rangle$, respectively, are statistically indistinguishable. Moreover, computational hiding of the commitment scheme Λ ensures that two sets $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and $\{c'_{i,b}\}_{i\in[n],b\in\{0,1\}}$ are computationally indistinguishable. Thus, $\mathcal{B}$ has a similar non-negligible polynomial advantage to $\mathcal{A}$ at breaking adaptive multi-theorem computational zero-knowledge of Π, i.e., with probability of $\frac{1}{p(\lambda)} - \mathsf{negl}(\lambda)$.

Simulation Extractability with Deletion: One can view the commitment algorithm $\Lambda.\mathsf{Com} : \{0,1\}^{\ell(\lambda)+1}$ as one-way function. Assume that there exists an adversary that, on input $\Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r)$, can invert the function and extract b and r with a non-negligible probability. Then, the adversary can break the hiding property with the same advantage. Lemma 3.6 then implies for any $\mathcal{A}$,

$$\Pr\left[\begin{array}{c} \forall i\in[n] : r_{i,0}, r_{i,1} \leftarrow \{0,1\}^{\ell(\lambda)}, u_i \leftarrow \{0,1\} \\ \forall i\in[n] : c_{i,0} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 0, r_{i,0}) \wedge c_{i,1} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 1, r_{i,1}) \ : \\ \{(b_i, r_i, d_i)\}_{i\in[n]} \leftarrow \mathcal{A}(\otimes_{i\in[n]} \frac{|0,r_{i,0}\rangle + (-1)^{u_i}|1,r_{i,1}\rangle}{\sqrt{2}}, \{c_{i,b}\}_{i,b}) \\ \wedge_{i\in[n]} \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b_i, r_i) \in \{c_{i,0}, c_{i,1}\} \\ \wedge \\ \wedge_{i\in[n]} d_i \cdot (r_{i,0} \oplus r_{i,1}) = u_i \end{array}\right] \leq \mathsf{negl}(\lambda). \tag{6}$$

Let $\Pi.\mathsf{Sim} = (\Pi.\mathsf{Sim}_0, \Pi.\mathsf{Sim}_1)$ be the simulators of Π and $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ be the simulators of our construction given by the corresponding multi-theorem zero-knowledge property. Let $\Pi.\mathsf{Ext}$ be the extractor given by the simulation extractibility of Π. We show the algorithm $\mathsf{Ext\text{-}Del}$, which satisfies simulation extractibility with deletion for our NIZK-CD construction as follows.

1. On input crs, td, x, π, and cert, parses the proof π as $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$.
2. Queries $\Pi.\mathsf{Ext}$ on input x^* and σ and receives the witness w^*.
3. Outputs w^* as w.

Reduction: Consider the case that simulation extractibility with deletion does not hold, i.e., there exists a QPT adversary algorithm $A = (A_0, A_1)$ such that given the extractor $\mathsf{Ext\text{-}Del}$, and some polynomial $p(\lambda)$,

$$\Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, \xi \leftarrow \mathcal{A}_0^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) \\ \pi, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x) \\ \pi^*, \mathsf{cert} \leftarrow \mathcal{A}_1^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}, \xi, x, \pi) \\ w \leftarrow \mathsf{Ext\text{-}Del}(\mathsf{crs}, \mathsf{td}, x, \pi^*, \mathsf{cert}) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, \pi^*) = 1 \\ \wedge \\ [\mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \ \vee \ (x,w) \notin R] \wedge \ x \notin Q \end{array}\right] \geq \frac{1}{p(\lambda)},$$

where Q is the list of queries from $\mathcal{A}$ to Sim_1, and $\mathcal{A}$ only receives the proof from the oracle Sim_1. Since Sim_1 forwards queries to $\Pi.\mathsf{Sim}_1$, we know that x^* is not queried to $\Pi.\mathsf{Sim}_1$. We parse the proof π^* as $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$. Since $\mathsf{Verify}(\mathsf{crs}, x, \pi^*) = 1$, we conclude $v_\sigma = \Pi.\mathsf{Verify}(\mathsf{crs}_\Pi, x^*, \sigma) = 1$. The witness w^* returned by $\Pi.\mathsf{Ext}$ is a valid witness to R^* as defined in Equation 1, except for a negligible probability. Then w^* must include at least one of either w such that $(x, w) \in R$ or $\{r_i\}_{i\in[n]}$ s.t. $c_{i,0} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 1, r_i)$ or $c_{i,1} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 0, r_i)$. Since the simulator Sim_1 generates $c_{i,b}$ as a commitment to message b and the commitment scheme is perfectly binding, the latter case cannot happen. Thus, w^* must include a satisfying witness w. Then, as $(x, w) \in R$, the adversary needs to output a valid deletion certificate $\mathsf{cert} = \{d_i\}_{i\in[n]}$ to pass the above experiment where as $\mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1$, i.e., for all $i \in [n]$, $d_i \cdot (r_{i,0} \oplus r_{i,1}) = u_i$.

On the other hand, since $\mathsf{Verify}(\mathsf{crs}, x, \pi^*) = 1$, we can conclude that the measurement outcomes are accepted, i.e., $v_i = 1$ for all $i \in [n]$. According to the definition of the unitary operation U in Equation 3, a measurement of $|\psi\rangle$ in the standard basis yields $\{r_i\}_{i\in[n]}$ s.t. $c_{i,0} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 0, r_i)$ or $c_{i,1} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 1, r_i)$, which contradicts the adaptive hardcore bit property.

More precisely, we can build an adversary algorithm $\mathcal{B}$ to break the adaptive hardcore bit property of the commitment, such that on input $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}$, $|\psi\rangle = \otimes_{i\in[n]} \frac{|r_{i,0}\rangle + (-1)^{u_i}|r_{i,1}\rangle}{\sqrt{2}}$, for an instance x, queries $x^* = (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$ to $\Pi.\mathsf{Sim}_1$ and get a simulated proof σ, generates $\pi := (x^*, \sigma, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$, and queries (x, π) to $\mathcal{A}_1$ to receive an accepting proof π^* and a valid deletion certificate cert. Then, $\mathcal{B}$ can parse π^* as $\{c'_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma', |\psi'\rangle$ and cert as $\{d_i\}_{i\in[n]}$. A measurement of $|\psi'\rangle$ in the standard basis yields $\{r_i\}_{i\in[n]}$ such that either $c_{i,0} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 0, r_i)$ or $c_{i,1} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 1, r_i)$. Also, $d_i \cdot (r_{i,0} \oplus r_{i,1}) = u_i$. Therefore, $\mathcal{B}$ can return $\{(b_i, r_i, d_i)\}_{i\in[n]}$ to pass the experiment. In summary, having $\mathcal{A}$ one can attack to either the simulation extractability of Π or to the adaptive hardcore bit property of Λ with advantage of $\geq \frac{1}{p(\lambda)} - \mathsf{negl}(\lambda)$. □

6 NIZKs with Certified Deletion from LWE

In this section, we show that our NIZK-CD constructions from LWE.

6.1 NIZK-CD with Public Verifiability

Here, we present NIZK-CD with publicly verifiable deletion certificates.

Theorem 6.1. *Assuming polynomial quantum hardness of LWE and given any post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP, a non-interactive simulation-extractable, adaptive multi-theorem computational zero-knowledge with certified deletion do exists where deletion certificates are publicly verifiable.*

Proof. We know that assuming the quantum hardness of LWE, there exists a compute-and-compare obfuscation scheme [WZ17], which we denote by $\mathsf{CC.Obf}$. Let $\Lambda = \langle \mathsf{Setup}, \mathsf{Com} \rangle$ be a post-quantum commitment with non-interactivity, perfect binding, and computational hiding properties, which can be realized from LWE [PS19]. Moreover, let $\Pi = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify} \rangle$ is a non-interactive simulation-extractable, adaptive multi-theorem computational zero-knowledge. An NIZK-CD scheme with publicly verifiable deletion certificates is as follows.

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: Runs $\mathsf{crs}_\Lambda, \mathsf{td}_\Lambda \leftarrow \Lambda.\mathsf{Setup}(1^\lambda)$, $\mathsf{crs}_\Pi, \mathsf{td}_\Pi \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$ and outputs $\mathsf{crs} := (\mathsf{crs}_\Lambda, \mathsf{crs}_\Pi)$ and $\mathsf{td} := (\mathsf{td}_\Lambda, \mathsf{td}_\Pi)$.
- $\pi, \mathsf{ck} \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)$: Let $n = \nu(\lambda)$ be a polynomial in λ. Proof generation proceeds similar to Section 5. The prover generates $c_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r_{i,b})$ for all $i \in [n]$ and $b \in \{0,1\}$. Given $x^* := (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$ and witness $w^* := (w, \{r_{i,b}\}_{i\in[n],b\in\{0,1\}})$, it runs $\Pi.\mathsf{Prove}(\mathsf{crs}_\Pi, x^*, w^*)$ and generates a NIZK σ for R^*, as defined in Equation 1. Let $\pi := (\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle)$, where $|\psi\rangle$ is defined as Equation 2. The only difference is the certification key. Consider I as a function with hard-coded values $\{r_{i,b}\}_{i\in[n],b\in\{0,1\}}$; on input $\{d_i\}_{i\in[n]}$ it evaluates $w_i = d_i \cdot (r_{i,0} \oplus r_{i,1})$ and outputs $w = w_0 w_1 \dots w_n$. Given a target value $\mathsf{lock} := u_0 u_1 \dots u_n$, the prover computes $\tilde{I} := \mathsf{CC.Obf}[I, \mathsf{lock}, 1]$ and outputs the certification key $\mathsf{ck} := \tilde{I}$. This key ck can be made public.
- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, \pi)$: The verifier first runs $v_\sigma \leftarrow \Pi.\mathsf{Verify}(\mathsf{crs}_\Pi, x^*, \sigma)$. Then, the state defined in Equation 5 is prepared, the verifier for all $i \in [n]$, measures $v_i = \mathsf{Cmt} - \mathsf{Cmp}(0, r_{i,0}, c_{i,0})$, and the outcome is $v = v_\sigma \wedge \bigwedge_{i\in[n]} v_i$.
- $\mathsf{cert} \leftarrow \mathsf{Delete}(\pi)$: The proof π is parsed as $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$ and the state $|\psi\rangle$ as $\bigotimes_{i\in[n]} |\psi_i\rangle$ such that $|\psi_i\rangle = \sum_{b\in\{0,1\}} \frac{1}{2}|b\rangle|r_{i,b}\rangle$. The verifier then measures each $|\psi_i\rangle$ in the Hadamard basis to get strings $d_i \in \{0,1\}^{\ell(\lambda)}$ such that each string satisfies $d_i \cdot (r_{i,0} \oplus r_{i,1}) = 0$. We define $\mathsf{cert} := \{d_i\}_{i\in[n]}$.
- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$: On input $\mathsf{cert} = \{d_i\}_{i\in[n]}$ run the program $\mathsf{ck} = \tilde{I}$, output 1 if it returns 1, and output 0 otherwise if it returns $\perp$.

Reductions for perfect completeness, adaptive multi-theorem computational zero-knowledge, and simulation extractability with deletion work are discussed in short since they are similar to reductions in Section 5.

Completeness: Completeness with respect to proof verification is borrowed from the completeness of Π which ensures that $v_\sigma = 1$ with a probability of 1. The commitment $\Lambda.\mathsf{Com}(.,.)$ is deterministic, i.e., $c_i = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b_i, r_{i,b})$. Thus, we ensure that $\forall i \in [n], b \in \{0,1\}$ we have $\mathsf{Cmt} - \mathsf{Cmp}(b_i, r_{i,b}, c_{i,b}) = 1$, and for all $i \in [n]$, v_i is measured to 1. Then, we consider deletion completeness. We parse the sate $|\psi\rangle$ as $\bigotimes_{i\in[n]} |\psi_i\rangle$, such that we have $|\psi_i\rangle = \sum_{b\in\{0,1\}} \frac{1}{2}|b\rangle|r_{i,b}\rangle$. The verifier measures each $|\psi_i\rangle$ in the Hadamard basis to get $d_i \in \{0,1\}^{\ell(\lambda)}$, where $d_i \cdot (r_{i,0} \oplus r_{i,1}) = 0$. The functionally preserving property of $\mathsf{CC.Obf}$ ensures that the obfuscated function $\tilde{I}$ acts similar to I except for a negligible probability, i.e., on input $\{d_i\}_{i\in[n]}$ outputs 1 with an overwhelming probability of $1 - \mathsf{negl}(\lambda)$.

Adaptive Multi-Theorem Computational Zero-Knowledge: Algorithm $\Pi.\mathsf{Sim} = (\Pi.\mathsf{Sim}_0, \Pi.\mathsf{Sim}_1)$ is the simulators of Π. Moreover, let $\mathsf{CC.Obf.Sim}$ be the simulator of the obfuscator $\mathsf{CC.Obf}$. $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ is as follows:

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda)$: Runs $\mathsf{crs}_\Lambda, \mathsf{td}_\Lambda \leftarrow \Lambda.\mathsf{Setup}(1^\lambda)$, $\mathsf{crs}_\Pi, \mathsf{td}_\Pi \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$ and outputs $\mathsf{crs} := (\mathsf{crs}_\Lambda, \mathsf{crs}_\Pi)$ and $\mathsf{td} := (\mathsf{td}_\Pi, \mathsf{td}_\Lambda)$.
- $\pi, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x)$: Let $r'_{i,b} \leftarrow \{0,1\}^{\ell(\lambda)}$ and $c'_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r'_{i,b})$ for all $i \in [n], b \in \{0,1\}$. Let $x^* := (x, \{c'_{i,b}\}_{i\in[n],b\in\{0,1\}})$. The algorithm queries $\Pi.\mathsf{Sim}_1$ on input x^* to get the simulated NIZK σ'. It samples $u'_i \leftarrow \{0,1\}$ and prepares $|\psi'\rangle$ as defined in Equation 2. It outputs $\pi := \{c'_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma', |\psi'\rangle$ and $\mathsf{ck} := \mathsf{CC.Obf.Sim}(I.\mathsf{params})$, where $I.\mathsf{params}$ are the public parameters, e.g., circuit size, of function the I as defined before in the protocol description.

  Reduction: Let $\mathcal{A}$ be a QPT adversary breaking multi-theorem computational zero-knowledge of NIZK-CD with an advantage of $\frac{1}{p(\lambda)}$ for polynomial $p(\lambda)$. We then construct a QPT adversary $\mathcal{B}$ that breaks the adaptive multi-theorem computational zero-knowledge of the underlying NIZK argument Π as follows.
  1. Receives crs from $\Pi.\mathsf{Setup}$ and $\Lambda.\mathsf{Setup}$ or $\Pi.\mathsf{Sim}_0$ and sends it to $\mathcal{A}$.
  2. On the query (x, w), samples $r_{i,b}$, computes $c_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r_{i,b})$ for all $i \in [n]$ and $b \in \{0,1\}$. Then, given instance $x^* := (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$ and witness $w^* := (w, \{r_{i,b}\}_{i\in[n],b\in\{0,1\}})$ receive the real or simulated proof σ by query either Prove on input (x^*, w^*) or $\Pi.\mathsf{Sim}_1$ on input x^*. Moreover, samples bits $u_i \leftarrow \{0,1\}$ and prepares $|\psi\rangle$ as defined in Equation 2. The certification key is computed as $\tilde{I} := \mathsf{CC.Obf}[I, \mathsf{lock}, 1]$ and $\mathsf{ck} = \tilde{I}$, where I is the function defined is the protocol description, i.e., with hard-coded values $\{r_{i,b}\}_{i\in[n],b\in\{0,1\}}$, which on input $\{d_i\}_{i\in[n]}$ gets $w_i = d_i \cdot (r_{i,0} \oplus r_{i,1})$ and outputs $w = w_0 w_1 \dots w_n$ and we define $\mathsf{lock} := u_0 u_1 \dots u_n$. The proof $\pi := (\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle)$ and the certification key ck are sent to $\mathcal{A}$.
  3. Output the outcome of $\mathcal{A}$.

  We sample $\{r_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and $\{u_i\}_{i\in[n],b\in\{0,1\}}$ uniformly random, thus, $|\psi\rangle$ and $|\psi'\rangle$ are statistically indistinguishable. Hiding property of commitments ensure that the set of commitments $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and $\{c'_{i,b}\}_{i\in[n],b\in\{0,1\}}$ are computationally indistinguishable. Furthermore, u_i is uniformly sampled, and the distribution over $\mathsf{CC}[I, \mathsf{lock}, 1]$ is unpredictable. Therefore, distributional indistinguishability of $\mathsf{CC.Obf}$ ensures functions $\tilde{I}$ and I are computationally indistinguishable. In summary, adversary $\mathcal{B}$ has an advantage of $\frac{1}{p(\lambda)} - \mathsf{negl}(\lambda)$.

Simulation Extractability with Deletion: The extractor $\mathsf{Ext\text{-}Del}$ is built similarly to the extractor described in Section 5, which additionally takes the certification key ck as input. In particular, $\mathsf{Ext\text{-}Del}$ works as follows.

1. On input crs, td, x, π, ck, and cert, parses π as $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$.
2. Queries $\Pi.\mathsf{Ext}$ on input x^* and σ and receives the witness w^*.
3. Outputs w^* as w.

Let $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ be a QPT adversary breaking simulation extractability with deletion with an advantage of $\frac{1}{p(\lambda)}$ for some polynomial $p(\lambda)$. We build a QPT adversary $\mathcal{B}$ breaking adaptive hard-core bit property defined in Equation 6.

On input $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}$, $|\psi\rangle = \otimes_{i\in[n]} \frac{|r_{i,0}\rangle+(-1)^{u_i}|r_{i,1}\rangle}{\sqrt{2}}$, and instance x, the adversary $\mathcal{B}$ queries $x^* = (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$ to $\Pi.\mathsf{Sim}_1$ and get a simulated proof σ. $\mathcal{B}$ queries $I.\mathsf{params}$ to $\mathsf{CC.Obf.Sim}$ to get a simulated obfuscated program $\tilde{I}$ and sets $\mathsf{ck} := \tilde{I}$. Let $\pi := (x^*, \sigma, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$. $\mathcal{B}$ queries (x, π, ck) to $\mathcal{A}_1$ to receive an accepting proof π^* and a valid deletion certificate cert. π^* is parsed as $\{c^*_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma^*, |\psi^*\rangle$ and cert as $\{d_i\}_{i\in[n]}$. As π^* is an accepting proof and commitments are perfectly binding, then the measurement of $|\psi'\rangle$ in the standard basis yields $\{r_i\}_{i\in[n]}$ s.t. $c_{i,0} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 0, r_i)$ or $c_{i,1} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, 1, r_i)$. Since cert is accepting and $\mathsf{CC.Obf}$ is functionality preserving, we can conclude that $d_i \cdot (r_{i,0} \oplus r_{i,1}) = u_i$ with a probability of $1 - \mathsf{negl}(\lambda)$. On the other side, the simulated obfuscated program includes no information about $\{r_i\}_{i\in[n]}$ since $\mathsf{CC.Obf}$ satisfies distributional indistinguishability. Thus, $\mathcal{B}$ breaks the adaptive hard-core bit property with an advantage of $\frac{1}{p(\lambda)} - \mathsf{negl}(\lambda)$.

□

6.2 NIZK-CD with Classical Communication

Here, we present NIZK-CD with solely classical communication.

Theorem 6.2. *Assuming polynomial quantum hardness of LWE and given any post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP, a non-interactive simulation-extractable, adaptive multi-theorem computational zero-knowledge with certified deletion do exists where the communication and prover algorithms are entirely classical.*

Proof. Assuming the hardness of LWE, a perfectly binding and computationally hiding commitment scheme [LS19] do exists, denoted by $\Lambda = \langle \mathsf{Setup}, \mathsf{Com}\rangle$. Let $\mathcal{F} := \{f_{k,b} : \mathcal{X} \to \mathcal{D}_\mathcal{Y}\}_{k\in\mathcal{K},b\in\{0,1\}}$ be that NTCF family from LWE [BCM+18]. Let $\Pi = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify}\rangle$ be the non-interactive simulation-extractable, adaptive multi-theorem computational zero-knowledge. Our construction with classical communication is described as follows.

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: Runs $\mathsf{crs}_\Lambda, \mathsf{td}_\Lambda \leftarrow \Lambda.\mathsf{Setup}(1^\lambda)$, $\mathsf{crs}_\Pi, \mathsf{td}_\Pi \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$ and outputs $\mathsf{crs} := (\mathsf{crs}_\Lambda, \mathsf{crs}_\Pi)$ and $\mathsf{td} := (\mathsf{td}_\Lambda, \mathsf{td}_\Pi)$.
- $\mathsf{ck}, \pi \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)$: Given polynomial n, it proceeds in two phases.

  State Preparation: This is executed interactively between prover and verifier.
  1. The prover runs $\mathsf{NTCF.Gen}_\mathcal{F}$, generates keys $\{k_i\}_{i\in[n]}$, trapdoors $\{\mathsf{td}_{k_i}\}_{i\in[n]}$, and sends the keys to the verifier. The verifier prepares the following state.

$$|\phi\rangle := \bigotimes_{i\in[n]} \frac{1}{\sqrt{|\mathcal{X}|}} \sum_{x\in\mathcal{X}} |0\rangle|x\rangle + |1\rangle|x\rangle \tag{7}$$

From Definition 3.2, $|\phi'\rangle$ can be turned into the following superposition.

$$|\phi'\rangle := \bigotimes_{i\in[n]} \frac{1}{\sqrt{2|\mathcal{X}|}} \sum_{x\in\mathcal{X},y\in\mathcal{Y}} \sqrt{(f'_{k_i,0}(x))(y)}|0\rangle|x\rangle|y\rangle + \frac{1}{\sqrt{2|\mathcal{X}|}} \sum_{x\in\mathcal{X},y\in\mathcal{Y}} \sqrt{(f'_{k_i,1}(x))(y)}|1\rangle|x\rangle|y\rangle \tag{8}$$

The verifier measures images, i.e., $|y\rangle$, in the standard basis, yielding,

$$\bigotimes_{i\in[n]} \frac{1}{\sqrt{2}} (|0\rangle|x_{i,0}\rangle + |1\rangle|x_{i,1}\rangle) \tag{9}$$

Here, for all $i \in [n]$, the tuple $(x_{i,0}, x_{i,1}) \in \mathcal{X}^2$ is a claw with respect to the measured image $y_i \in \mathcal{Y}$. Finally, the verifier sends the images $\{y_i\}_{i\in[n]}$ to the prover and applies the injection $J : \mathcal{X} \to \{0,1\}^{\ell(\lambda)}$, as defined in Definition 3.2, in the above superposition and prepares the state $|\psi\rangle$ as

$$|\psi\rangle := \bigotimes_{i\in[n]} \frac{1}{2} (|0\rangle|r_{i,0}\rangle + |1\rangle|r_{i,1}\rangle)$$
$$\forall i \in [n], b \in \{0,1\} : r_{i,b} = J(x_{i,b}). \tag{10}$$

2. For all $i \in [n]$, $b \in \{0,1\}$, prover runs $\mathsf{Inv}(\mathsf{td}_k, b, y_i)$ and $J(x_{i,b})$ to get $r_{i,b}$.

Proof Generation: The remaining parts are similar to Section 5. The prover generates commitments $c_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r_{i,b})$ for all $i \in [n]$ and $b \in \{0,1\}$. Given $x^* := (x, \{c_{i,b}\}_{i\in[n],b\in\{0,1\}})$, witness $w^* := (w, \{r_{i,b}\}_{i\in[n],b\in\{0,1\}})$, the prover runs $\Pi.\mathsf{Prove}(\mathsf{crs}, x^*, w^*)$ and generates a NIZK σ for R^*, as defined in Equation 1. $\pi = \{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$ and $\mathsf{ck} = \{r_{i,b}\}_{i\in[n],b\in\{0,1\}}$.

- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, \pi)$: The verifier runs $v_\sigma \leftarrow \Pi.\mathsf{Verify}(\mathsf{crs}, x^*, \sigma)$. Let $U^{\mathsf{com}}_{c_0,c_1}$ and U be as before, i.e., Equation 3. Using U the state in Equation 5 is prepared. Then, the verifier for all $i \in [n]$, measures the commit-and-compare registers $v_i = \mathsf{Cmt} - \mathsf{Cmp}(0, r_{i,0}, c_{i,0})$. The outcome is $v = v_\sigma \wedge \bigwedge_{i\in[n]} v_i$.
- $\mathsf{cert} \leftarrow \mathsf{Delete}(\pi)$: The proof π is parsed as $\{c_{i,b}\}_{i\in[n],b\in\{0,1\}}, \sigma, |\psi\rangle$ and also the state $|\psi\rangle$ as $\bigotimes_{i\in[n]}|\psi_i\rangle$ such that $|\psi_i\rangle = \sum_{b\in\{0,1\}} \frac{1}{2}|b\rangle|r_{i,b}\rangle$. The verifier then for all $i \in [n]$ measures the state $|\psi_i\rangle$ in the Hadamard basis to get strings $d_i \in \{0,1\}^{\ell(\lambda)}$ such that $d_i \cdot (r_{i,0} \oplus r_{i,1}) = 0$. We define $\mathsf{cert} := \{d_i\}_{i\in[n]}$.
- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$: The key ck is parsed as $\{r_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and cert as $\{d_i\}_{i\in[n]}$, validates whether for all $i \in [n]$ it holds $d_i \cdot (r_{i,0} \oplus r_{i,1}) = 0$.

Reductions work basically similar to Section 5.

Completeness: Completeness of Π ensures that $v_\sigma = 1$ with a probability of 1. The invert $\mathsf{Inv}(.,.,.)$ and the map $J(.)$ are deterministic, the randomnesses computed by the prover using trapdoor td_k and the map $J(.)$ are the same as

randomnesses encoded in the state $|\psi\rangle$. Moreover, as the commitment algorithm $\Lambda.\mathsf{Com}(.,.,.)$ is deterministic, and the commitments are honestly generated by the prover, i.e., $c_i = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b_i, r_{i,b})$, we ensure that for all $i \in [n]$ and $b \in \{0,1\}$ we have $\mathsf{Cmt} - \mathsf{Cmp}(b_i, r_{i,b}, c_{i,b}) = 1$. For all $i \in [n]$, v_i is measured to 1. Consider completeness with respect to deletion. We parse the sate $|\psi\rangle$ as $\bigotimes_{i\in[n]}|\psi_i\rangle$, such that we have $|\psi_i\rangle = \sum_{b\in\{0,1\}} \frac{1}{2}|b\rangle|r_{i,b}\rangle$. The verifier measures each $|\psi_i\rangle$ in the Hadamard basis to get $d_i \in \{0,1\}^{\ell(\lambda)}$ s.t. $d_i \cdot (r_{i,0} \oplus r_{i,1}) = 0$.

Adaptive Multi-Theorem Computational Zero-Knowledge: Algorithm $\Pi.\mathsf{Sim} = (\Pi.\mathsf{Sim}_0, \Pi.\mathsf{Sim}_1)$ is the simulators of Π. $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ is as:

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda)$: Outputs $\mathsf{crs}_\Pi, \mathsf{td}_\Pi \leftarrow \Pi.\mathsf{Sim}_1(1^\lambda)$.
- $\pi, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x)$: Runs $\mathsf{NTCF.Gen}_\mathcal{F}$, generates $\{k'_i\}_{i\in[n]}$, $\{\mathsf{td}_{k'_i}\}_{i\in[n]}$, and sends the key k' to the adversary. For all $i \in [n]$, on input the image $y'_i \in \mathcal{Y}$, computes $x'_{i,0} \leftarrow \mathsf{Inv}(\mathsf{td}_{k'}, 0, y'_i)$ and $x'_{i,1} \leftarrow \mathsf{Inv}(\mathsf{td}_{k'}, 1, y'_i)$, and the randomnesses $r'_{i,0} = J(x'_{i,0})$ and $r'_{i,1} = J(x'_{i,1})$. Then, the simulator computes commitments $c'_{i,b} = \Lambda.\mathsf{Com}(\mathsf{crs}_\Lambda, b, r'_{i,b})$ the for all $i \in [n]$ and $b \in \{0,1\}$. We define $x^* := (x, \{c'_{i,b}\}_{i\in[n],b\in\{0,1\}})$. The algorithm query $\Pi.\mathsf{Sim}_1$ on input x^* to get the simulated NIZK σ'. Finally, Sim_1 prepares $|\psi\rangle$, as in Equation 10.

  Reduction: It proceeds similarly to the zero-knowledge reduction in Section 5. The additional note is that the key k' are uniformly sampled by $\mathsf{NTCF.Gen}_\mathcal{F}$, hence, it is indistinguishable from the real key k.

Simulation Extractability with Deletion: We can build an extractor Ext-Del that satisfies simulation extractibility with deletion and present the reduction the same as Section 5. The adaptive hardcore bit property of NTCF hash functions, as defined in Definition 3.2 prevents deviating from the proving algorithm. □

7 Applications

We discuss revocable signatures of knowledge and anonymous credentials.

7.1 Revocable Signature of Knowledge

We present the definition and a construction of revocable signature of knowledge.

Definition 7.1. (Revocable Signature of Knowledge) *Let* NP *relation* R *with language* L *and message space* $\mathcal{M}$. $\Sigma_R = \langle \mathsf{Setup}, \mathsf{Sign}, \mathsf{Verify}, \mathsf{Delete}, \mathsf{Certify} \rangle$ *is a revocable signature of knowledge if it satisfies the following.*

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: *on input* λ, *outputs* crs *and trapdoor* td.
- $\sigma, \mathsf{ck} \leftarrow \mathsf{Sign}(\mathsf{crs}, x, w, m)$: *on input* crs, $(x, w) \in R$, $m \in \mathcal{M}$, *outputs a signature ok knowledge* σ *and a certification key* ck.

- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, m, \sigma)$: *on input* crs*,* x*,* m*,* σ*, accepts or rejects.*
- $\mathsf{cert} \leftarrow \mathsf{Delete}(\sigma)$*: on input* σ*, outputs a deletion certificate* cert*.*
- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$*: on input* ck *and* cert*, outputs accept or reject.*

Correctness: *For every* $\lambda \in \mathbb{N}$*, pair* $(x, w) \in R$ *and message* $m \in \mathcal{M}$*,*

$$\Pr\left[\begin{array}{c} \mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda) \\ \sigma, \mathsf{ck} \leftarrow \mathsf{Sign}(\mathsf{crs}, x, w, m) \\ \mathsf{cert} \leftarrow \mathsf{Delete}(\sigma) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, m, \sigma) = 1 \\ \wedge \\ \mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

Simulation: *There exist a* QPT *simulator algorithm* $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ *such that for every* QPT *algorithm* $\mathcal{A}$ *and sufficiently large* $\lambda \in \mathbb{N}$*,*

$$\begin{aligned} &\Big|\Pr\left[\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda) : \mathcal{A}^{\mathsf{Sign}(\mathsf{crs},\cdot,\cdot,\cdot)}(\mathsf{crs}) = 1\right] - \\ &\Pr\left[\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) : \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot,\cdot)}(\mathsf{crs}) = 1\right]\Big| \leq \mathsf{negl}(\lambda). \end{aligned}$$

Extraction with Deletion: *Let* $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ *be the simulators of simulation. A* QPT *extractor algorithm* $\mathsf{Ext\text{-}Del}$ *exists such that for every* QPT *algorithm* $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ *and sufficiently large* $\lambda \in \mathbb{N}$*,*

$$\Pr\left[\begin{array}{c} \mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda) \\ x, m \leftarrow \mathcal{A}_0^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot,\cdot)}(\mathsf{crs}) \\ \sigma, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x, m) \\ \sigma^*, \mathsf{cert} \leftarrow \mathcal{A}_1^{\mathsf{Sim}_1(\mathsf{crs},\mathsf{td},\cdot,\cdot)}(\mathsf{crs}, \sigma) \\ w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, m, \sigma^*, \mathsf{cert}) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, m, \sigma^*) = 1 \\ \wedge \\ [\mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \ \vee \ (x, w) \notin R] \wedge \ (x, m) \notin Q \end{array}\right] \leq \mathsf{negl}(\lambda),$$

where Q *is the list of queries from* $\mathcal{A}$ *to* Sim_1*.*

Theorem 7.1. *Assuming any non-interactive simulation-extractable, adaptive multi-theorem computational zero-knowledge with certified deletion, there exists a revocable signature of knowledge.*

Proof. Let $\Gamma = \langle \mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify}, \mathsf{Delete}, \mathsf{Certify} \rangle$ be NIZK-CD with adaptive multi-theorem computational zero-knowledge and simulation extractability.

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Setup}(1^\lambda)$: The algorithms outputs $\mathsf{crs}, \mathsf{td} \leftarrow \Gamma.\mathsf{Setup}(1^\lambda)$.
- $\sigma, \mathsf{ck} \leftarrow \mathsf{Sign}(\mathsf{crs}, x, w, m)$: Let $x^* = (x, m)$ be instance, $w^* = w$ witness for

$$L^* = \{(x, m) : \exists w \text{ s.t. } (x, w) \in R\}.$$

  Then, we have $\sigma, \mathsf{ck} \leftarrow \Gamma.\mathsf{Prove}(\mathsf{crs}, x^*, w^*)$ with respect to L^*.

- $\{0,1\} \leftarrow \mathsf{Verify}(\mathsf{crs}, x, m, \sigma)$: Given $x^* = (x, m)$, outputs $v = \Gamma.\mathsf{Verify}(\mathsf{crs}, x^*, \sigma)$.
- $\mathsf{cert} \leftarrow \mathsf{Delete}(\sigma)$: The algorithms outputs $\mathsf{cert} = \Gamma.\mathsf{Delete}(\sigma)$.
- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$: The algorithms outputs $v = \Gamma.\mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$.

Next, we prove correctness, simulation, and extraction with deletion.

Correctness: Since the scheme Γ satisfies perfect completeness, for any honestly generated σ and cert, both satisfy Verify and $\mathsf{Certify}$, respectively.

Simulation: Let $\Gamma.Sim = (\Gamma.Sim_0, \Gamma.\mathsf{Sim}_1)$ be the simulators of Γ. We build $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ for our construction as follows.

- $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{Sim}_0(1^\lambda)$: Outputs $\mathsf{crs}_\Gamma, \mathsf{td}_\Gamma \leftarrow \Gamma.\mathsf{Sim}_1(1^\lambda)$.
- $\sigma, \mathsf{ck} \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x, m)$: Outputs $\sigma, \mathsf{ck} \leftarrow \Gamma.\mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, x^* := (x, m))$.

Reduction: Suppose a QPT adversary $\mathcal{A}$ exists such that for polynomial $p(\lambda)$, it breaks simulation property of Σ_R with an advantage more than $\frac{1}{p(\lambda)}$. We construct a QPT adversary $\mathcal{B}$ for the zero-knowledge property of Γ as follows.

1. Receives the real or simulated crs_Γ and sends it to the adversary $\mathcal{A}$.
2. One each query (x, m, w), defines $x^* := (x, m)$ and witness $w^* := w$, receives the signature σ by query either Sign or $\Gamma.\mathsf{Sim}_1$, and sends σ to $\mathcal{A}$.
3. Output the result of $\mathcal{A}$.

$\mathcal{B}$ has the same advantage $\frac{1}{p(\lambda)}$ at breaking the zero-knowledge property of Γ.

Extraction with deletion: Let $\Gamma.\mathsf{Sim} = (\Gamma.\mathsf{Sim}_0, \Gamma.\mathsf{Sim}_1)$ be the simulators of Γ and $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ be the simulators of our construction. Let $\Gamma.\mathsf{Ext}$ be the extractor of Γ. We show an extractor Ext for extractibility with deletion.

1. On input crs, td, x, m, σ, and cert, runs $\Pi.\mathsf{Ext}(\mathsf{crs}, \mathsf{td}, (x, m), \sigma, \mathsf{cert})$, and receive the witness w^*.
2. Outputs w^* as w.

Reduction: Let $A = (A_0, A_1)$ a QPT algorithm such that given the extractor Ext, and some polynomial $p(\lambda)$, it breaks extraction with deletion property of Σ_R with an advantage more than $\frac{1}{p(\lambda)}$. Then, one can build an adversary algorithm $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1)$ to break simulation extractibility with deletion of Γ, where $\mathcal{B}_0$ outputs $x^* = (x, m)$ returned by $\mathcal{A}_0$ and $\mathcal{B}_1$ outputs σ^*, cert returned by $\mathcal{A}_1$. □

Corollary 7.1. *Assuming post-quantum one-way function and post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge, revocable signature of knowledge exists.*

Proof. This follows from Theorem 5.1 and Theorem 7.1. □

Corollary 7.2. *Given polynomial quantum hardness of the LWE problem and post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP, there exists a revocable signatures of knowledge with publicly verifiable deletion.*

Proof. This follows from Theorem 6.1 and Theorem 7.1. □

Corollary 7.3. *Given polynomial quantum hardness of the LWE problem and post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP, there exists a revocable signatures of knowledge with classical communication and signer.*

Proof. This follows from Theorem 6.2 and Theorem 7.1. □

7.2 Revocable Anonymous Credentials

We define and construct revocable anonymous credentials.

Definition 7.2. (Revocable Anonymous Credentials) *[JK23] The scheme* $\Delta_R = \langle \mathsf{Setup}, \mathsf{Sign}, \mathsf{Verify}, \mathsf{Delete}, \mathsf{Certify} \rangle$ *is a revocable anonymous credentials with respect to a set of accesses* $\{S_\lambda\}_{\lambda\in\mathbb{N}}$ *if it satisfies the following.*

- $\mathsf{nym}, \mathsf{sk} \leftarrow \mathsf{IssuerSetup}(1^\lambda)$: *outputs a pseudonym* nym *with a secret key* sk.
- $\mathsf{cred}, \mathsf{ck} \leftarrow \mathsf{Issue}(\mathsf{nym}, \mathsf{sk}, \mathsf{access})$: *on input* nym, sk, *and requested access* access, *outputs an anonymous credentials* cred *and a certification key* ck.
- $\{0,1\} \leftarrow \mathsf{VerifyCred}(\mathsf{nym}, \mathsf{access}, \mathsf{cred})$: *on input* nym, access, *and* cred, *outputs* 1 *as accept or* 0 *as reject for validating the anonymous credentials.*
- $\mathsf{cert} \leftarrow \mathsf{Delete}(\mathsf{cred})$*: on input* cred, *outputs a deletion certificate* cert.
- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$*: on input* ck *and* cert, *accepts or rejects.*

Correctness: *For every* $\lambda \in \mathbb{N}$, *pair* $(x, w) \in R$ *and* $m \in \mathcal{M}$,

$$\Pr\left[\begin{array}{c} \mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda) \\ \sigma, \mathsf{ck} \leftarrow \mathsf{Sign}(\mathsf{crs}, x, w, m) \\ \mathsf{cert} \leftarrow \mathsf{Delete}(\sigma) \end{array} : \begin{array}{c} \mathsf{Verify}(\mathsf{crs}, x, m, \sigma) = 1 \\ \wedge \\ \mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \end{array}\right] \geq 1 - \mathsf{negl}(\lambda)$$

Revocation: *For every* QPT *algorithm* $\mathcal{A}$, *sufficiently large* λ, *and access* access, *the following probability is, at most,* $\mathsf{negl}(\lambda)$.

$$\Pr\left[\begin{array}{c} \mathsf{nym}, \mathsf{sk} \leftarrow \mathsf{IssuerSetup}(1^\lambda) \\ \mathsf{cred}, \mathsf{ck} \leftarrow \mathsf{Issue}(\mathsf{nym}, \mathsf{sk}, \mathsf{access}) \\ \mathsf{cred}^*, \mathsf{cert} \leftarrow \mathcal{A}(\mathsf{nym}, \mathsf{cred}) \end{array} : \begin{array}{c} \mathsf{VerifyCred}(\mathsf{nym}, \mathsf{access}, \mathsf{cred}^*) = 1 \\ \wedge \mathsf{Certify}(\mathsf{ck}, \mathsf{cert}) = 1 \end{array}\right]$$

Theorem 7.2. *Assuming any revocable signature of knowledge, there exists a revocable anonymous credentials.*

Proof. Let $(\mathcal{X}, \mathcal{W})$ be some hard NP distribution. Let Σ_R be revocable signature of knowledge. Our construction of anonymous credentials is presented as follows.

- $\mathsf{nym}, \mathsf{sk} \leftarrow \mathsf{IssuerSetup}(1^\lambda)$: Generates $\mathsf{crs} \leftarrow \Sigma_R.\mathsf{Setup}(1^\lambda)$, samples a pair $(x, w) \leftarrow (\mathcal{X}, \mathcal{W})$, and outputs $\mathsf{nym} = (\mathsf{crs}, x)$ and $\mathsf{sk} = w$.
- $\mathsf{cred}, \mathsf{ck} \leftarrow \mathsf{Issue}(\mathsf{nym}, \mathsf{sk}, \mathsf{access})$: Outputs $\mathsf{cred}, \mathsf{ck} \leftarrow \Sigma_R.\mathsf{Sign}(\mathsf{crs}, x, \mathsf{access}, w)$.
- $\{0,1\} \leftarrow \mathsf{VerifyCred}(\mathsf{nym}, \mathsf{access}, \mathsf{cred})$: Outputs $v = \Gamma.\mathsf{Verify}(\mathsf{crs}, x, \mathsf{access}, \sigma)$.
- $\mathsf{cert} \leftarrow \mathsf{Delete}(\mathsf{cred})$: The algorithms outputs $\mathsf{cert} = \Gamma.\mathsf{Delete}(\sigma)$.
- $\{0,1\} \leftarrow \mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$: The algorithms outputs $v = \Gamma.\mathsf{Certify}(\mathsf{ck}, \mathsf{cert})$.

Next, we prove correctness and revocation for the proposed construction.

Correctness: Since the scheme Σ_R satisfies correctness, for honestly generated credentials cred and certificate cert, both satisfy Verify and $\mathsf{Certify}$, respectively.

Revocation: Suppose that there exists a QPT adversary algorithm $\mathcal{A}$ such that for some polynomial $p(\lambda)$, it breaks revocation property of Δ. We construct a QPT adversary $\mathcal{B}$ to break extraction with deletion of Σ_R as described below.

1. Receives simulated crs_{Σ_R} from $\Sigma_R.\mathsf{Sim}_0$, samples a pair $(x, w) \leftarrow (\mathcal{X}, \mathcal{W})$, samples an access access, queries $\Sigma_R.\mathsf{Sim}_1$ on input (x, access) to get σ, and sends $\mathsf{nym} = (\mathsf{crs}, x), \mathsf{cred} = \sigma$ to $\mathcal{A}$.
3. Outputs the credential cred^* as signature and cert as deletion certificate.

$\mathcal{B}$ has the same advantage $\frac{1}{p(\lambda)}$ at breaking extraction with deletion of Σ_R. □

Corollary 7.4. *Assuming post-quantum one-way function and post-quantum non-interactive simulation-extractable, adaptively multi-theorem computational zero-knowledge, revocable anonymous credentials exists.*

Proof. This follows from Corollary 7.1 and Theorem 7.2. □

Corollary 7.5. *Given polynomial quantum hardness of the LWE problem and post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP, revocable anonymous credentials with publicly verifiable deletion exists.*

Proof. This follows from Corollary 7.2 and Theorem 7.2. □

Corollary 7.6. *Given polynomial quantum hardness of the LWE problem and post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge for NP, revocable anonymous credentials with classical communication and issuer exists.*

Proof. This follows from Corollary 7.3 and Theorem 7.2. □

References

Aar09. Scott Aaronson. Quantum copy-protection and quantum money. In *2009 24th Annual IEEE Conference on Computational Complexity*. IEEE, July 2009.

AC12. Scott Aaronson and Paul Christiano. Quantum money from hidden subspaces. In *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '12, page 41–60, New York, NY, USA, 2012. Association for Computing Machinery.

AKN+23. Shweta Agrawal, Fuyuki Kitagawa, Ryo Nishimaki, Shota Yamada, and Takashi Yamakawa. Public key encryption with secure key leasing. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023*, pages 581–610, Cham, 2023. Springer Nature Switzerland.

AKPW13. Joël Alwen, Stephan Krenn, Krzysztof Pietrzak, and Daniel Wichs. Learning with rounding, revisited. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013*, pages 57–74, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

ALL+21. Scott Aaronson, Jiahui Liu, Qipeng Liu, Mark Zhandry, and Ruizhe Zhang. New approaches for quantum copy-protection. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology – CRYPTO 2021*, pages 526–555, Cham, 2021. Springer International Publishing.

ALP21. Prabhanjan Ananth and Rolando L. La Placa. Secure software leasing. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021*, pages 501–530, Cham, 2021. Springer International Publishing.

AN11. Tolga Acar and Lan Nguyen. Revocation for delegatable anonymous credentials. In Dario Catalano, Nelly Fazio, Rosario Gennaro, and Antonio Nicolosi, editors, *Public Key Cryptography – PKC 2011*, pages 423–440, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

APV23. Prabhanjan Ananth, Alexander Poremba, and Vinod Vaikuntanathan. Revocable cryptography from learning with errors. In Guy Rothblum and Hoeteck Wee, editors, *Theory of Cryptography*, pages 93–122, Cham, 2023. Springer Nature Switzerland.

BB14. Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science*, 560:7–11, December 2014.

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, *Advances in Cryptology - CRYPTO 2009*, pages 108–125, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

BCM+18. Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 320–331, 2018.

BGK+24. James Bartusek, Vipul Goyal, Dakshita Khurana, Giulio Malavolta, Justin Raizes, and Bhaskar Roberts. Software with certified deletion. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology – EUROCRYPT 2024*, pages 85–111, Cham, 2024. Springer Nature Switzerland.

BI20. Anne Broadbent and Rabib Islam. *Quantum Encryption with Certified Deletion*, page 92–122. Springer International Publishing, 2020.

BK23. James Bartusek and Dakshita Khurana. Cryptography with certified deletion. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology – CRYPTO 2023*, pages 192–223, Cham, 2023. Springer Nature Switzerland.

BKM+23. James Bartusek, Dakshita Khurana, Giulio Malavolta, Alexander Poremba, and Michael Walter. Weakening assumptions for publicly-verifiable deletion. In *Theory of Cryptography: 21st International Conference, TCC 2023, Taipei, Taiwan, November 29–December 2, 2023, Proceedings, Part IV*, page 183–197, Berlin, Heidelberg, 2023. Springer-Verlag.

BL20. Anne Broadbent and Sébastien Lord. Uncloneable Quantum Encryption via Oracles. In Steven T. Flammia, editor, *15th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020)*, volume 158 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 4:1–4:22, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

BR24. James Bartusek and Justin Raizes. Secret sharing with certified deletion. In *Advances in Cryptology – CRYPTO 2024: 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2024, Proceedings, Part VII*, page 184–214, Berlin, Heidelberg, 2024. Springer-Verlag.

CGJL23. Orestis Chardouvelis, Vipul Goyal, Aayush Jain, and Jiahui Liu. Quantum key leasing for pke and fhe with a classical lessor. Cryptology ePrint Archive, Paper 2023/1640, 2023. https://eprint.iacr.org/2023/1640.

CKS10. Jan Camenisch, Markulf Kohlweiss, and Claudio Soriente. Solving revocation with efficient update of anonymous credentials. In Juan A. Garay and Roberto De Prisco, editors, *Security and Cryptography for Networks*, pages 454–471, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

CL06. Melissa Chase and Anna Lysyanskaya. On signatures of knowledge. In Cynthia Dwork, editor, *Advances in Cryptology - CRYPTO 2006*, pages 78–96, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

CLLZ21. Andrea Coladangelo, Jiahui Liu, Qipeng Liu, and Mark Zhandry. Hidden cosets and applications to unclonable cryptography. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology – CRYPTO 2021*, pages 556–584, Cham, 2021. Springer International Publishing.

CMP22. Andrea Coladangelo, Christian Majenz, and Alexander Poremba. Quantum copy-protection of compute-and-compare programs in the quantum random oracle model, 2022.

DCIO98. Giovanni Di Crescenzo, Yuval Ishai, and Rafail Ostrovsky. Non-interactive and non-malleable commitment. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, page 141–150, New York, NY, USA, 1998. Association for Computing Machinery.

FLS90. U. Feige, D. Lapidot, and A. Shamir. Multiple non-interactive zero knowledge proofs based on a single random string. In *Proceedings [1990] 31st Annual Symposium on Foundations of Computer Science*, pages 308–317 vol.1, 1990.

GMR85. S Goldwasser, S Micali, and C Rackoff. The knowledge complexity of interactive proof-systems. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, STOC '85, page 291–304, New York, NY, USA, 1985. Association for Computing Machinery.

GMR23. Vipul Goyal, Giulio Malavolta, and Justin Raizes. Unclonable commitments and proofs. Cryptology ePrint Archive, Paper 2023/1538, 2023. https://eprint.iacr.org/2023/1538.

Got03. Daniel Gottesman. Uncloneable encryption. *Quantum Info. Comput.*, 3(6):581–602, nov 2003.

Hel69. Carl W. Helstrom. Quantum detection and estimation theory. *J. Statist. Phys.*, 1:231–252, 1969.

HILL99. Johan HÅstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

HMNY21. Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Quantum encryption with certified deletion, revisited: Public key, attribute-based, and classical communication. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2021*, pages 606–636, Cham, 2021. Springer International Publishing.

Hol73. A.S Holevo. Statistical decision theory for quantum systems. *Journal of Multivariate Analysis*, 3(4):337–394, 1973.

JK23. Ruta Jawale and Dakshita Khurana. Unclonable non-interactive zero-knowledge. Cryptology ePrint Archive, Paper 2023/1532, 2023. https://eprint.iacr.org/2023/1532.

KN22. Fuyuki Kitagawa and Ryo Nishimaki. Functional encryption with secure key leasing. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022*, pages 569–598, Cham, 2022. Springer Nature Switzerland.

KNY21. Fuyuki "Kitagawa, Ryo Nishimaki, and Takashi" Yamakawa. "secure software leasing from standard assumptions". In Kobbi "Nissim and Brent" Waters, editors, *"Theory of Cryptography"*, pages "31–61", "Cham", "2021". "Springer International Publishing".

KNY23. Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Publicly verifiable deletion from minimal assumptions. In Guy Rothblum and Hoeteck Wee, editors, *Theory of Cryptography*, pages 228–245, Cham, 2023. Springer Nature Switzerland.

KS24. Jonathan Katz and Ben Sela. Secret sharing with publicly verifiable deletion. Cryptology ePrint Archive, Paper 2024/1596, 2024.

LLQZ22. Jiahui Liu, Qipeng Liu, Luowen Qian, and Mark Zhandry. Collusion resistant copy-protection for watermarkable functionalities. In Eike Kiltz and Vinod Vaikuntanathan, editors, *Theory of Cryptography*, pages 294–323, Cham, 2022. Springer Nature Switzerland.

LS19. Alex Lombardi and Luke Schaeffer. A note on key agreement and non-interactive commitments. Cryptology ePrint Archive, Paper 2019/279, 2019. https://eprint.iacr.org/2019/279.

MP12. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, pages 700–718, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

MPY24. Tomoyuki Morimae, Alexander Poremba, and Takashi Yamakawa. Revocable Quantum Digital Signatures. In Frédéric Magniez and Alex Bredariol Grilo, editors, *19th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2024)*, volume 310 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 5:1–5:24, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

MY22. Tomoyuki Morimae and Takashi Yamakawa. Quantum commitments and signatures without one-way functions. In *Advances in Cryptology – CRYPTO 2022: 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15–18, 2022, Proceedings, Part I*, page 269–295, Berlin, Heidelberg, 2022. Springer-Verlag.

Nao91. Moni Naor. Bit commitment using pseudorandomness. *J. Cryptol.*, 4(2):151–158, January 1991.

Pas03. Rafael Pass. On deniability in the common reference string and random oracle model. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003*, pages 316–337, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

Por23. Alexander Poremba. Quantum Proofs of Deletion for Learning with Errors. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 90:1–90:14, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

PRSD17. Chris Peikert, Oded Regev, and Noah Stephens-Davidowitz. Pseudorandomness of ring-lwe for any ring and modulus. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, page 461–473, New York, NY, USA, 2017. Association for Computing Machinery.

PS19. Chris Peikert and Sina Shiehian. Noninteractive zero knowledge for np from (plain) learning with errors. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 89–114, Cham, 2019. Springer International Publishing.

Reg09. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), sep 2009.

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.

Sah99. A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039)*, pages 543–553, 1999.

SCO+01. Alfredo De Santis, Giovanni Di Crescenzo, Rafail Ostrovsky, Giuseppe Persiano, and Amit Sahai. Robust non-interactive zero knowledge. In *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '01, page 566–598, Berlin, Heidelberg, 2001. Springer-Verlag.

Unr14. Dominique Unruh. Revocable quantum timed-release encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, pages 129–146, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

Wie83. Stephen Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, jan 1983.

WZ17. Daniel Wichs and Giorgos Zirdelis. Obfuscating compute-and-compare programs under lwe. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 600–611, 2017.