# Certified Everlasting Zero-Knowledge Proof for QMA

Taiga Hiroka[1], Tomoyuki Morimae[1], Ryo Nishimaki[2], Takashi Yamakawa[2]

[1]Yukawa Institute for Theoretical Physics, Kyoto University, Kyoto, Japan
{taiga.hiroka,tomoyuki.morimae}@yukawa.kyoto-u.ac.jp
[2]NTT Corporation, Tokyo, Japan
{ryo.nishimaki.zk,takashi.yamakawa.ga}@hco.ntt.co.jp

September 29, 2021

**Abstract**

In known constructions of classical zero-knowledge protocols for **NP**, either of zero-knowledge or soundness holds only against computationally bounded adversaries. Indeed, achieving both statistical zero-knowledge and statistical soundness at the same time with classical verifier is impossible for **NP** unless the polynomial-time hierarchy collapses, and it is also believed to be impossible even with a quantum verifier. In this work, we introduce a novel compromise, which we call the certified everlasting zero-knowledge proof for **QMA**. It is a computational zero-knowledge proof for **QMA**, but the verifier issues a classical certificate that shows that the verifier has deleted its quantum information. If the certificate is valid, even unbounded malicious verifier can no longer learn anything beyond the validity of the statement.

We construct a certified everlasting zero-knowledge proof for **QMA**. For the construction, we introduce a new quantum cryptographic primitive, which we call commitment with statistical binding and certified everlasting hiding, where the hiding property becomes statistical once the receiver has issued a valid certificate that shows that the receiver has deleted the committed information. We construct commitment with statistical binding and certified everlasting hiding from quantum encryption with certified deletion by Broadbent and Islam [TCC 2020] (in a black box way), and then combine it with the quantum sigma-protocol for **QMA** by Broadbent and Grilo [FOCS 2020] to construct the certified everlasting zero-knowledge proof for **QMA**. Our constructions are secure in the quantum random oracle model. Commitment with statistical binding and certified everlasting hiding itself is of independent interest, and there will be many other useful applications beyond zero-knowledge.

# Contents

# 1 Introduction

## 1.1 Background

Zero-knowledge [GMR89], which roughly states that the verifier cannot learn anything beyond the validity of the statement, is one of the most important concepts in cryptography and computer science. The study of zero-knowledge has a long history in classical cryptography, and recently there have been many results in quantum cryptography. In known constructions of classical zero-knowledge protocols for **NP**, either of zero-knowledge or soundness holds only against computationally bounded adversaries. Indeed, achieving both statistical zero-knowledge and statistical soundness at the same time with classical verifier is impossible for **NP** unless the polynomial-time hierarchy collapses [For87]. It is also believed to be impossible even with a quantum verifier [MW18].

Broadbent and Islam [BI20] recently suggested an idea of the novel compromise: realizing "everlasting zero-knowledge" by using quantum encryption with certified deletion. The everlasting security defined by Unruh [Unr13] states that the protocol remains secure as long as the adversary runs in polynomial-time during the execution of the protocol. Quantum encryption with certified deletion introduced by Broadbent and Islam [BI20] is a new quantum cryptographic primitive where a classical message is encrypted into a quantum ciphertext, and the receiver in possession of a quantum ciphertext can generate a classical certificate that shows that the receiver has deleted the quantum ciphertext. If the certificate is valid, the receiver can no longer decrypt the message even if it receives the secret key. Broadbent and Islam's idea is to use quantum commitment with a similar certified deletion security to encrypt the first message from the prover to the verifier in the standard $\Sigma$-protocol. Once the verifier issues the deletion certificate for all commitments that are not opened by the verifier's challenge, even an unbounded verifier can no longer access the committed values of the unopened commitments. They left the formal definition and the construction as future works.

There are many obstacles to realizing their idea. First, their quantum encryption with certified deletion cannot be directly used in a $\Sigma$-protocol because it does not have any binding property. Their ciphertext consists of a classical and quantum part. The classical part is $m \oplus u \oplus H(r)$, where $m$ is the plaintext, $u$ and $r$ are random bit strings, and $H$ is a hash function. The quantum part is a random BB84 states whose computational basis states encode $r$. The decryption key is $u$ and the place of computational basis states that encode $r$, and therefore it is not binding: by changing $u$, a different message can be obtained. We therefore need to extend quantum encryption with certified deletion in such a way that the statistical binding property is included.

Second, defining a meaningful notion of "everlasting zero-knowledge proof" itself is non-trivial. In fact, everlasting zero-knowledge proofs for **QMA** or even for **NP** in the sense of Unruh's definition [Unr13] are unlikely to exist.[1] To see this, recall that the definition of quantum statistical zero-knowledge [Wat09, MW18] requires a simulator to simulate the view of a *quantum polynomial-time* malicious verifier in a statistically indistinguishable manner. Therefore, everlasting zero-knowledge in the sense of Unruh's definition [Unr13] is actually equivalent to quantum statistical zero-knowledge. On the other hand, as already mentioned, it is believed that quantum statistical zero-knowledge proofs for **NP** do not exist [MW18]. In particular, Menda and Watrous [MW18] constructed an oracle relative to which quantum statistical zero-knowledge proofs for (even a subclass of) **NP** do not exist.

However, we notice that this argument does not go through for *certified everlasting zero-knowledge*, where the verifier can issue a classical certificate that shows that the verifier has deleted its information. Once a valid certificate has been issued, even unbounded malicious verifier can no longer learn anything beyond the validity of the statement. The reason is that certified everlasting zero-knowledge does not imply statistical zero-knowledge since it does not ensure any security against a malicious verifier that refuses to provide a valid certificate of deletion. Therefore, we have the following question.

*Is it possible to define and construct a certified everlasting zero-knowledge proof for **QMA**?*

[1] We mention that everlasting zero-knowledge *arguments*, which only satisfy computational soundness, can exist. Indeed, any statistical zero-knowledge argument is everlasting zero-knowledge argument. One may think that the computational soundness is fine since that ensures everlasting soundness in the sense of Unruh's definition [Unr13]. For practical purposes, this may be true. On the other hand, we believe that it is theoretically interesting to pursue (a kind of) everlasting zero-knowledge without compromising the soundness as is done in this paper.

## 1.2 Our Results

In this work, we define and construct the certified everlasting zero-knowledge proof for **QMA**. This goal is achieved in the following four steps.

1. We define a new quantum cryptographic primitive, which we call *commitment with statistical binding and certified everlasting hiding*. In this new commitment scheme, binding is statistical but hiding is computational. However, the hiding property becomes statistical once the receiver has issued a valid certificate that shows that the receiver has deleted the committed information.

2. We construct commitment with statistical binding and certified everlasting hiding. We use secret-key quantum encryption with certified deletion as the building block in a black box way . This construction is secure in the quantum random oracle model [BDF+11].

3. We define a new notion of zero-knowledge proof, which we call *the certified everlasting zero-knowledge proof for* **QMA**. It is a computational zero-knowledge proof for **QMA** with the following additional property. A verifier can issue a classical certificate that shows that the verifier has deleted its information. If the certificate is valid, even unbounded malicious verifier can no longer learn anything beyond the validity of the statement.

4. We apply commitment with statistical binding and certified everlasting hiding to the quantum $\Sigma$-protocol for **QMA** by Broadbent and Grilo [BG20] to construct the certified everlasting zero-knowledge proof for **QMA**.

We have three remarks on our results. First, although our main results are the definition and the construction of the certified everlasting zero-knowledge proof for **QMA**, our commitment with statistical binding and certified everlasting hiding itself is also of independent interest. There will be many other useful applications beyond zero-knowledge. In fact, it is known that binding and hiding cannot be made statistical at the same time even in the quantum world [LC97, May97], and therefore our new commitment scheme provides a nice compromise.

Second, our new commitment scheme and the new zero-knowledge proof are the first cryptographic applications of symmetric-key quantum encryption with certified deletion. Although certified deletion is conceptually very interesting, there was no concrete construction of cryptographic applications when it was first introduced [BI20]. One reason why the applications are limited is that in cryptography it is not natural to consider the case when the receiver receives the private key later. Hiroka et al. [HMNY21] recently extended the symmetric-key scheme by Broadbent and Islam [BI20] to a public-key encryption scheme, an attribute-based encryption scheme, and a publicly verifiable scheme, which have opened many applications. However, one disadvantage is that their security is the computational one unlike the symmetric-key scheme [BI20]. Therefore it was open whether there is any cryptographic application of the information-theoretically secure certified deletion scheme. Our results provide the first cryptographic applications of it. Interestingly, the setup of the symmetric-key scheme [BI20], where the receiver does not have the private key in advance, nicely fits into the framework of the $\Sigma$-protocol, because the verifier (receiver) in the $\Sigma$-protocol does not have the decryption key of the first encrypted message from the prover (sender).

Finally, note that certified everlasting zero-knowledge and certified everlasting hiding seem to be impossible in the classical world, because a malicious adversary can copy its information. In particular, certified everlasting zero-knowledge against classical verifiers clearly implies honest-verifier statistical zero-knowledge since an honest verifier runs in polynomial-time.[2] Moreover, it is known that **HVSZK** $=$ **SZK** where **HVSZK** and **SZK** are languages that have honest-verifier statistical zero-knowledge proofs and (general) statistical zero-knowledge proofs, respectively [GSV98]. Therefore, if certified everlasting zero-knowledge proofs for **NP** with classical verification exist, we obtain **NP** $\subseteq$ **HVSZK** $=$ **SZK**, which means the collapse of the polynomial-time hierarchy [For87]. Though the above argument only works for protocols in the standard model, no construction of honest-verifier statistical zero-knowledge proofs for **NP** is known in the random oracle model either. Our results therefore add novel items to the list of quantum cryptographic primitives that can be achieved only in the quantum world.

[2] A similar argument does not work for quantum verifiers since the honest-verifier quantum statistical zero-knowledge [Wat02] requires a simulator to simulate honest verifier's internal state *at any point* of the protocol execution. This is not implied by certified everlasting zero-knowledge, which only requires security after generating a valid deletion certificate.

## 1.3 Technical Overview

**Certified everlasting zero-knowledge.** As explained in Section 1.1, everlasting zero-knowledge proofs for **NP** (and for **QMA**) seem impossible even with quantum verifiers. Therefore, we introduce a relaxed notion of zero-knowledge which we call *certified everlasting zero-knowledge* inspired by quantum encryption with certified deletion introduced by Broadbent and Islam [BI20]. Certified everlasting zero-knowledge ensures security against malicious verifiers that run in polynomial-time during the protocol and provide a valid certificate that sensitive information is "deleted". (For the formal definition, see Section 4.1.) The difference from everlasting zero-knowledge is that it does not ensure security against malicious verifiers that do not provide a valid certificate. We believe that this is still a meaningful security notion since if the verifier refuses to provide a valid certificate, the prover may penalize the verifier for cheating.

**Quantum commitment with certified everlasting hiding.** Our construction of certified everlasting zero-knowledge proofs is based on the idea sketched by Broadbent and Islam [BI20]. (For the details of the construction, see Section 4.2.) The idea is to implement a $\Sigma$-protocol using a commitment scheme with certified deletion. However, they did not give a construction or definition of commitment with certified deletion. First, we remark that the encryption with certified deletion in [BI20] cannot be directly used as a commitment. A natural way to use their scheme as a commitment scheme is to consider a ciphertext as a commitment. However, since different secret keys decrypt the same ciphertext into different messages, this does not satisfy the binding property as commitment.

A natural (failed) attempt to fix this problem is to add a classical commitment to the secret key of the encryption scheme with certified deletion making use of the fact that the secret key of the encryption with certified deletion in [BI20] is classical. That is, a commitment to a message $m$ consists of

$$(\mathsf{CT} = \mathsf{Enc}(\mathsf{sk}, m), \mathsf{com} = \mathsf{Commit}(\mathsf{sk}))$$

where $\mathsf{Enc}$ is the encryption algorithm of the scheme in [BI20], $\mathsf{sk}$ is its secret key, and $\mathsf{Commit}$ is a statistically binding and computationally hiding classical commitment scheme. This resolves the issue of binding since the secret key is now committed by the classical commitment scheme. On the other hand, we cannot prove a hiding property that is sufficiently strong for achieving certified everlasting zero-knowledge. It is not difficult to see that what we need here is *certified everlasting hiding*, which ensures that once a receiver generates a valid certificate that it deleted the commitment in a polynomial-time, the hiding property is ensured even if the receiver runs in unbounded-time afterwards. Unfortunately, we observe that the above generic construction seems insufficient for achieving certified everlasting hiding.[3] The reason is as follows: We want to reduce the certified everlasting hiding to the certified deletion security of $\mathsf{Enc}$. However, the security of $\mathsf{Enc}$ can be invoked only if $\mathsf{sk}$ is information theoretically hidden before the deletion. On the other hand, $\mathsf{sk}$ is committed by a statistically binding commitment in the above construction, and thus $\mathsf{sk}$ is information theoretically determined from the commitment. Therefore, we have to somehow delete the information of $\mathsf{sk}$ from the commitment in some hybrid game in a security proof. A similar issue was dealt with by Hiroka et al. [HMNY21] by using receiver non-committing encryption in the context of public key encryption with certified deletion. However, their technique inherently relies on the assumption that an adversary runs in polynomial-time *even after the deletion*. Therefore, their technique is not applicable in the context of certified everlasting hiding.

To overcome the above issue, we rely on random oracles. We modify the above construction as follows:

$$(\mathsf{CT} = \mathsf{Enc}(\mathsf{sk}, m), \mathsf{com} = \mathsf{Commit}(R), H(R) \oplus \mathsf{sk})$$

where $R$ is a sufficiently long random string and $H$ is a hash function modeled as a random oracle whose output length is the same as that of $\mathsf{sk}$. We give an intuition on why the above issue is resolved with this modification. As explained in the previous paragraph, we want to delete the information of $\mathsf{sk}$ from the commitment in some hybrid game. By the computational hiding of commitment, a polynomial-time receiver cannot find $R$ from $\mathsf{Commit}(R)$. Therefore, it cannot get any information on $H(R)$ since otherwise we can "extract" $R$ from one of receiver's queries. This argument can be made rigorous by using the one-way to hiding lemma [Unr15, AHU19]. Importantly, we only have to assume that the receiver runs in polynomial-time *before the deletion* and do not need to assume anything about the running time after

[3] One may think that we can just use statistically hiding commitment. However, such a commitment can only satisfy computational binding, which is not sufficient for achieving certified everlasting zero-knowledge *proofs* rather than arguments.

the deletion because we extract $R$ from one of the queries before the deletion. Since $\mathsf{sk}$ is masked by $H(R)$, the receiver cannot get any information on $\mathsf{sk}$ either. Thus, we can simulate the whole commitment $(\mathsf{CT}, \mathsf{com}, H(R) \oplus \mathsf{sk})$ without using $\mathsf{sk}$, which resolves the issue and enables us to reduce certified everlasting hiding to certified deletion security of $\mathsf{Enc}$.

We remark that quantum commitments in general cannot satisfy the binding property in the classical sense. Indeed, if a malicious sender generates a superposition of valid commitments on different messages $m_0$ and $m_1$, it can later open to $m_0$ or $m_1$ with probability $1/2$ for each. Defining a binding property for quantum commitments is non-trivial, and there have been proposed various flavors of definitions in the literature, e.g., [CDMS04, DFS04, DFR+07, Yan20, BB21]. It might be possible to adopt some of those definitions. However, we choose to introduce a new definition, which we call *classical-extractor-based binding*, tailored to our construction because this is more convenient for our purpose. Classical-extractor-based binding captures the property of our construction that the randomness $R$ is information-theoretically determined by the classical part $\mathsf{com} = \mathsf{Commit}(R)$ of a commitment, and the decommitment can be done by using the rest part of the commitment and $R$.[4] In particular, this roughly means that one can extract the committed message with an unbounded-time extractor before the sender decommits.This enables us to prove soundness for our certified everlasting zero-knowledge proofs in essentially the same manner as in the classical case.

The details of the construction and security proofs are explained in Section 3.2.

**Certified everlasting zero-knowledge proof for QMA.** Once we obtain a commitment scheme with certified everlasting hiding, the construction of certified everlasting zero-knowledge proofs is straightforward based on the idea sketched in [BI20]. Though they only considered a construction for **NP**, we observe that the idea can be naturally extended to a construction for **QMA** since a "quantum version" of $\Sigma$-protocol for **QMA** called $\Xi$-protocol is constructed by Broadbent and Grilo [BG20]. Below, we sketch the construction for clarity. Let $A = (A_{\mathsf{yes}}, A_{\mathsf{no}})$ be a promise problem in **QMA**. [BG20] showed that for any $\mathsf{x} \in A_{\mathsf{yes}}$ and any corresponding witness $\mathsf{w}$, it is possible to generate (in a quantum polynomial-time) so-called the local simulatable history state $\rho_{\mathsf{hist}}$ from $\mathsf{w}$, which satisfies the following two special properties (for details, see Definition 2.4):

(LS1) The verification can be done by measuring randomly chosen five qubits of $\rho_{\mathsf{hist}}$.

(LS2) The classical description of any five-qubit reduced density matrix of $\rho_{\mathsf{hist}}$ can be obtained in classical polynomial-time.

With these properties, the quantum $\Sigma$-protocol of [BG20] is constructed as follows:

1. **Commitment phase**: The prover randomly chooses $x, z \in \{0,1\}^n$, and sends $(X^x Z^z \rho_{\mathsf{hist}} Z^z X^x) \otimes \mathsf{com}(x,z)$ to the verifier, where $X^x Z^z := \prod_{i=1}^n X_i^{x_i} Z_i^{z_i}$, $n$ is the number of qubits of $\rho_{\mathsf{hist}}$, and $\mathsf{com}(x,z)$ is a classical commitment of $(x,z)$.
2. **Challenge phase**: The verifier randomly chooses a subset $S \subset [n]$ of size $|S| = 5$, and sends it to the prover.
3. **Response phase**: The prover opens the commitment for $\{x_i, z_i\}_{i \in S}$.
4. **Verification phase**: The verifier applies $\prod_{i \in S} X_i^{x_i} Z_i^{z_i}$ on the state and measures qubits in $S$.

The correctness and the soundness come from the property (LS1), and the zero-knowledge comes from the property (LS2). If the classical commitment scheme used in the above construction is the one with statistical binding and computational hiding, the quantum $\Sigma$-protocol is a computational zero-knowledge proof for **QMA**, because the unbounded malicious verifier can open the commitment of $\{x_i, z_i\}_{i \in [n] \setminus S}$, and therefore can obtain the entire $\rho_{\mathsf{hist}}$. If more than five qubits of $\rho_{\mathsf{hist}}$ are available to the malicious verifier, the zero-knowledge property no longer holds.

We construct the certified everlasting zero-knowledge proof for **QMA** based on the quantum $\Sigma$-protocol. Our idea is to use commitment with certified everlasting hiding and statistical binding for the commitment of $(x,z)$ in the above construction of the quantum $\Sigma$-protocol. If the verifier issues a valid deletion certificate for the commitment of $\{x_i, z_i\}_{i \in [n] \setminus S}$, even unbounded malicious verifier can no longer learn $\{x_i, z_i\}_{i \in [n] \setminus S}$, and therefore what it can

[4] For this definition to make sense, we need to require that $\mathsf{com} = \mathsf{Commit}(R)$ is classical. This can be ensured if the honest receiver measures it as soon as receiving it even if only quantum communication channel is available.

access is only the five qubits of $\rho_{\mathsf{hist}}$. This gives a proof for certified everlasting zero-knowledge. Using classical-extractor-based binding, the proof of statistical soundness can be done almost in the same way as in [BG20]. Recall that classical-extractor-based binding enables us to extract the committed message with an unbounded-time extractor before the sender decommits. Therefore, we can extract the committed $(x, z)$ from $\mathsf{com}(x, z)$. Since the extraction is done before the challenge phase, the extracted values do not depend on the challenge $S$. Then, it is easy to reduce the soundness of the scheme to that of the the original **QMA** promise problem $A$. The details of the construction is explained in Section 4.2.

### 1.4 Related Works

**Zero-knowledge for QMA.** Zero-knowledge for **QMA** was first constructed by Broadbent, Ji, Song, and Watrous [BJSW16]. Broadbent and Grilo [BG20] gave an elegant and simpler construction what they call the $\Xi$-protocol (which is considered as a quantum version of the standard $\Sigma$-protocol) by using the local simulatability [GSY19]. Our construction is based on the $\Xi$-protocol. Bitansky and Shmueli [BS20] gave the first constant round zero-knowledge argument for **QMA** with negligible soundness error. Brakerski and Yuen [BY20] gave a construction of 3-round *delayed-input* zero-knowledge proof for **QMA** where the prover needs to know the statement and witness only for generating its last message. Chardouvelis and Malavolta [CM21] constructed 4-round statistical zero-knowledge arguments for **QMA** and 2-round zero-knowledge for **QMA** in the timing model.

Regarding non-interactive zero-knowledge proofs or arguments (NIZK), Kobayashi [Kob03] first studied (statistically sound and zero-knowledge) NIZKs in a model where the prover and verifier share Bell pairs, and gave a complete problem in this setting. It is unlikely that the complete problem contains (even a subclass of) **NP** [MW18], and thus even a NIZK for all **NP** languages is unlikely to exist in this model. Chailloux et al. [CCKV08] showed that there exists a (statistically sound and zero-knowledge) NIZK for all languages in **QSZK** in the help model where a trusted party generates a pure state *depending on the statement to be proven* and gives copies of the state to both prover and verifier. Recently, there are many constructions of NIZK proofs or arguments for **QMA** in various kind of setup models and assumptions [ACGH20, CVZ20, BG20, Shm21, BCKM21, MY21, BM21].

**Quantum commitment.** It is well-known that statistically binding and hiding commitments are impossible even with quantum communication [LC97, May97]. On the other hand, there are a large body of literature on constructing quantum commitments assuming some computational assumptions, e.g., see the references in the introduction of [Yan20]. Among them, several works showed the possibility of using quantum commitments in constructions of zero-knowledge proofs and arguments [YWLQ15, FUW+20, Yan20, BB21]. However, they only consider replacing classical commitments with quantum commitments in classical constructions while keeping the same functionality and security level as the classical construction. In particular, none of them considers protocols for **QMA** or properties that are classically impossible to achieve like our notion of the certified everlasting zero-knowledge.

## 2 Preliminaries

### 2.1 Notations

Here we introduce basic notations we will use. In this paper, $x \leftarrow X$ denotes selecting an element from a finite set $X$ uniformly at random, and $y \leftarrow A(x)$ denotes assigning to $y$ the output of a probabilistic or deterministic algorithm $A$ on an input $x$. When we explicitly show that $A$ uses randomness $r$, we write $y \leftarrow A(x; r)$. When $D$ is a distribution, $x \leftarrow D$ denotes sampling an element from $D$. Let $[n]$ be the set $\{1, \dots, n\}$. Let $\lambda$ be a security parameter, and $y := z$ denotes that $y$ is set, defined, or substituted by $z$. For a bit string $s \in \{0,1\}^n$, $s_i$ denotes the $i$-th bit of $s$. QPT stands for quantum polynomial time. PPT stands for (classical) probabilistic polynomial time. For a subset $S \subseteq W$ of a set $W$, $\overline{S}$ is the complement of $S$, i.e., $\overline{S} := W \setminus S$. A function $f : \mathbb{N} \to \mathbb{R}$ is a negligible function if for any constant $c$, there exists $\lambda_0 \in \mathbb{N}$ such that for any $\lambda > \lambda_0$, $f(\lambda) < \lambda^{-c}$. We write $f(\lambda) \leq \mathsf{negl}(\lambda)$ to denote $f(\lambda)$ being a negligible function.

## 2.2 Quantum Computation

We assume the familiarity with basics of quantum computation, and use standard notations. Let us denote $\mathcal{Q}$ be the state space of a single qubit. $I$ is the two-dimensional identity operator. For simplicity, we often write $I^{\otimes n}$ as $I$ for any $n$ when the dimension of the identity operator is clear from the context. For any single-qubit operator $O$, $O_i$ means an operator that applies $O$ on the $i$-th qubit and applies $I$ on all other qubits. $X$ and $Z$ are the Pauli $X$ and $Z$ operators, respectively. For any $n$-bit strings $x := (x_1, x_2, \cdots, x_n) \in \{0,1\}^n$ and $z := (z_1, z_2, \cdots, z_n) \in \{0,1\}^n$, $X^x := \prod_{i\in[n]} X_i^{x_i}$ and $Z^z := \prod_{i\in[n]} Z_i^{z_i}$. For any subset $S$, $\mathrm{Tr}_S$ means the trace over all qubits in $S$. For any quantum state $\rho$ and a bit string $s \in \{0,1\}^n$, $\rho \otimes s$ means $\rho \otimes |s\rangle\langle s|$. The trace distance between two states $\rho$ and $\sigma$ is given by $\frac{1}{2}\|\rho - \sigma\|_{\mathrm{tr}}$, where $\|A\|_{\mathrm{tr}} := \mathrm{Tr}\sqrt{A^\dagger A}$ is the trace norm. If $\frac{1}{2}\|\rho - \sigma\|_{\mathrm{tr}} \leq \epsilon$, we say that $\rho$ and $\sigma$ are $\epsilon$-close. If $\epsilon = \mathsf{negl}(\lambda)$, then we say that $\rho$ and $\sigma$ are statistically indistinguishable.

Let $C_0$ and $C_1$ be quantum channels from $p$ qubits to $q$ qubits, where $p$ and $q$ are polynomials. We say that they are computationally indistinguishable, and denote it by $C_0 \approx_c C_1$ if there exists a negligible function $\mathsf{negl}$ such that $|\Pr[D((C_0 \otimes I)(\sigma)) = 1] - \Pr[D((C_1 \otimes I)(\sigma)) = 1]| \leq \mathsf{negl}(\lambda)$ for any polynomial $k$, any $(p+k)$-qubit state $\sigma$, and any polynomial-size quantum circuit $D$ acting on $q+k$ qubits. We say that $C_0$ and $C_1$ are statistically indistinguishable, and denote it by $C_0 \approx_s C_1$, if $D$ is an unbounded algorithm.

**Lemma 2.1 (Quantum Rewinding Lemma [Wat09]).** *Let $Q$ be a quantum circuit that acts on an $n$-qubit state $|\psi\rangle$ and an $m$-qubit auxiliary state $|0^m\rangle$. Let $p(\psi) := ||(\langle 0| \otimes I)Q(|\psi\rangle \otimes |0^m\rangle)||^2$ and $|\phi(\psi)\rangle := \frac{1}{\sqrt{p(\psi)}}(\langle 0| \otimes I)Q(|\psi\rangle \otimes |0^m\rangle)$. Let $p_0, q \in (0,1)$ and $\epsilon \in (0, \frac{1}{2})$ such that $|p(\psi) - q| < \epsilon$, $p_0(1-p_0) < q(1-q)$, and $p_0 < p(\psi)$. Then there is a quantum circuit $R$ of size at most $O\left(\frac{\log(\frac{1}{\epsilon})\mathrm{size}(Q)}{p_0(1-p_0)}\right)$ such that on input $|\psi\rangle$, $R$ computes a quantum state $\rho(\psi)$ that satisfies $\langle\phi(\psi)|\rho(\psi)|\phi(\psi)\rangle \geq 1 - 16\epsilon\frac{\log^2(\frac{1}{\epsilon})}{p_0^2(1-p_0)^2}$.*

**Lemma 2.2 (One-Way to Hiding Lemma [AHU19]).** *Let $S \subseteq \mathcal{X}$ be a random subset of $\mathcal{X}$. Let $G, H : \mathcal{X} \to \mathcal{Y}$ be random functions satisfying $\forall x \notin S\ [G(x) = H(x)]$. Let $z$ be a random classical bit string. ($S, G, H, z$ may have an arbitrary joint distribution.) Let $\mathcal{A}$ be an oracle-aided quantum algorithm that makes at most $q$ quantum queries. Let $\mathcal{B}$ be an algorithm that on input $z$ chooses $i \leftarrow [q]$, runs $\mathcal{A}^H(z)$, measures $\mathcal{A}$'s $i$-th query, and outputs the measurement outcome. Then we have $\left|\Pr\left[\mathcal{A}^G(z) = 1\right] - \Pr\left[\mathcal{A}^H(z) = 1\right]\right| \leq 2q\sqrt{\Pr[\mathcal{B}^H(z) \in S]}$.*

## 2.3 QMA and $k$-SimQMA

**Definition 2.3 (QMA).** *We say that a promise problem $A = (A_{\mathsf{yes}}, A_{\mathsf{no}})$ is in **QMA** if there exist a polynomial $p$, a QPT algorithm $V$, and $0 \leq \beta < \alpha \leq 1$ with $\alpha - \beta \geq \frac{1}{\mathrm{poly}(|\mathsf{x}|)}$ such that*

**Completeness:** *For any $\mathsf{x} \in A_{\mathsf{yes}}$, there exists a quantum state $\mathsf{w}$ of $p(|\mathsf{x}|)$-qubit (called a witness) such that*

$$\Pr[V(\mathsf{x}, \mathsf{w}) = \top] \geq \alpha.$$

**Soundness:** *For any $\mathsf{x} \in A_{\mathsf{no}}$ and any quantum state $\mathsf{w}$ of $p(|\mathsf{x}|)$-qubit,*

$$\Pr[V(\mathsf{x}, \mathsf{w}) = \top] \leq \beta.$$

*For any $\mathsf{x} \in A_{\mathsf{yes}}$, $R_A(\mathsf{x})$ is the (possibly infinite) set of all quantum states $\mathsf{w}$ such that $\Pr[V(\mathsf{x}, \mathsf{w}) = \top] \geq \frac{2}{3}$.*

A complexity class of $k$-**SimQMA** is introduced, and proven to be equal to **QMA** in [BG20].

**Definition 2.4 ($k$-SimQMA [BG20]).** *A promise problem $A = (A_{\mathsf{yes}}, A_{\mathsf{no}})$ is in $k$-**SimQMA** with soundness $\beta(|\mathsf{x}|) \leq 1 - \frac{1}{\mathrm{poly}(|\mathsf{x}|)}$, if there exist polynomials $m$ and $n$ such that given $\mathsf{x} \in A_{\mathsf{yes}}$, there is an efficient deterministic algorithm that computes $m(|\mathsf{x}|)$ $k$-qubit POVMs $\{\Pi_1, I - \Pi_1\}, \ldots, \{\Pi_{m(|\mathsf{x}|)}, I - \Pi_{m(|\mathsf{x}|)}\}$ such that:*

**Simulatable completeness:** *If $\mathsf{x} \in A_{\mathsf{yes}}$, there exists an $n(|\mathsf{x}|)$-qubit state $\rho_{\mathsf{hist}}$, which we call a simulatable witness, such that for all $c \in [m]$, $\mathrm{Tr}(\Pi_c \rho_{\mathsf{hist}}) \geq 1 - \mathsf{negl}(|\mathsf{x}|)$, and there exists a set of $k$-qubit density matrices $\{\rho_{\mathsf{sim}}^{\mathsf{x},S}\}_{S\subseteq[n(|\mathsf{x}|)],|S|=k}$ that can be computed in polynomial time from $\mathsf{x}$ and $\rho_{\mathsf{hist}}$ such that $||\mathrm{Tr}_{\overline{S}}(\rho_{\mathsf{hist}}) - \rho_{\mathsf{sim}}^{\mathsf{x},S}||_{\mathrm{tr}} \leq \mathsf{negl}(|\mathsf{x}|)$.*

**Soundness:** *If* $\mathsf{x} \in A_{\mathsf{no}}$*, for any* $n(|\mathsf{x}|)$*-qubit state* $\rho$*,* $\frac{1}{m}\sum_{c\in[m]} \mathrm{Tr}(\Pi_c \rho) \leq \beta(|\mathsf{x}|)$.

## 2.4 Cryptographic Tools

In this section, we review cryptographic tools used in this paper.

**Non-interactive commitment.**

**Definition 2.5 (Non-Interactive Commitment (Syntax)).** *Let* $\lambda$ *be the security parameter and let* $p$*,* $q$ *and* $r$ *be some polynomials. A (classical) non-interactive commitment scheme consists of a single PPT algorithm* $\mathsf{Commit}$ *with plaintext space* $\mathcal{M} := \{0,1\}^{p(\lambda)}$*, randomness space* $\{0,1\}^{q(\lambda)}$ *and commitment space* $\mathcal{C} := \{0,1\}^{r(\lambda)}$ *satisfying two properties:*

**Perfect binding:** *For every* $(r_0, r_1) \in \{0,1\}^{q(\lambda)} \times \{0,1\}^{q(\lambda)}$ *and* $(m, m') \in \mathcal{M}^2$ *such that* $m \neq m'$*, we have that* $\mathsf{Commit}(m; r_0) \neq \mathsf{Commit}(m'; r_1)$*, where* $(\mathsf{Commit}(m; r_0), \mathsf{Commit}(m'; r_1)) \in \mathcal{C}^2$.

**Unpredictability:** *Let* $\Sigma := \mathsf{Commit}$*. For any QPT adversary* $\mathcal{A}$*, we define the following security experiment* $\mathsf{Exp}^{\mathsf{unpre}}_{\Sigma,\mathcal{A}}(\lambda)$.

1. *The challenger chooses* $R \leftarrow \mathcal{M}$ *and* $R' \leftarrow \{0,1\}^{q(\lambda)}$*, computes* $\mathsf{com} \leftarrow \mathsf{Commit}(R; R')$*, and sends* $\mathsf{com}$ *to* $\mathcal{A}$.
2. $\mathcal{A}$ *outputs* $R^*$*. The output of the experiment is* $1$ *if* $R^* = R$*. Otherwise, the output of the experiment is* $0$.

*We say that the commitment is unpredictable if for any QPT adversary* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{unpre}}_{\Sigma,\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{unpre}}_{\Sigma,\mathcal{A}}(\lambda) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

*Remark* 2.6. The unpredictability is a weaker version of computational hiding. We define unpredictability instead of computational hiding since this suffices for our purpose.

A non-interactive commitment scheme that satisfies the above definition exists assuming the existence of injective one-way functions or perfectly correct public key encryption [LS19]. Alternatively, we can also instantiate it based on random oracles.

**Quantum encryption with certified deletion.** Broadbent and Islam [BI20] introduced the notion of quantum encryption with certified deletion.

**Definition 2.7 (One-Time SKE with Certified Deletion (Syntax)).** *Let* $\lambda$ *be the security parameter and let* $p$*,* $q$ *and* $r$ *be some polynomials. A one-time secret key encryption scheme with certified deletion consists of a tuple of algorithms* $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Del}, \mathsf{Verify})$ *with plaintext space* $\mathcal{M} := \{0,1\}^n$*, ciphertext space* $\mathcal{C} := \mathcal{Q}^{\otimes p(\lambda)}$*, key space* $\mathcal{K} := \{0,1\}^{q(\lambda)}$ *and deletion certificate space* $\mathcal{D} := \{0,1\}^{r(\lambda)}$.

$\mathsf{KeyGen}(1^\lambda) \to \mathsf{sk}$**:** *The key generation algorithm takes as input the security parameter* $1^\lambda$*, and outputs a secret key* $\mathsf{sk} \in \mathcal{K}$.

$\mathsf{Enc}(\mathsf{sk}, m) \to \mathsf{CT}$**:** *The encryption algorithm takes as input* $\mathsf{sk}$ *and a plaintext* $m \in \mathcal{M}$*, and outputs a ciphertext* $\mathsf{CT} \in \mathcal{C}$.

$\mathsf{Dec}(\mathsf{sk}, \mathsf{CT}) \to m'$ *or* $\bot$**:** *The decryption algorithm takes as input* $\mathsf{sk}$ *and* $\mathsf{CT}$*, and outputs a plaintext* $m' \in \mathcal{M}$ *or* $\bot$.

$\mathsf{Del}(\mathsf{CT}) \to \mathsf{cert}$**:** *The deletion algorithm takes as input* $\mathsf{CT}$*, and outputs a certification* $\mathsf{cert} \in \mathcal{D}$.

$\mathsf{Verify}(\mathsf{sk}, \mathsf{cert}) \to \top$ *or* $\bot$**:** *The verification algorithm takes* $\mathsf{sk}$ *and* $\mathsf{cert}$*, and outputs* $\top$ *or* $\bot$.

**Definition 2.8 (Correctness for One-Time SKE with Certified Deletion).** *There are two types of correctness. One is decryption correctness and the other is verification correctness.*

**Decryption correctness:** *There exists a negligible function* $\mathsf{negl}$ *such that for any* $\lambda \in \mathbb{N}$ *and* $m \in \mathcal{M}$*,*

$$\Pr\left[\mathsf{Dec}(\mathsf{sk},\mathsf{CT}) = m \;\middle|\; \begin{array}{l}\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda)\\ \mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{sk}, m)\end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

**Verification correctness:** *There exists a negligible function* $\mathsf{negl}$ *such that for any* $\lambda \in \mathbb{N}$ *and* $m \in \mathcal{M}$*,*

$$\Pr\left[\mathsf{Verify}(\mathsf{sk},\mathsf{cert}) = \top \;\middle|\; \begin{array}{l}\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda)\\ \mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{sk}, m)\\ \mathsf{cert} \leftarrow \mathsf{Del}(\mathsf{CT})\end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

**Definition 2.9 (Certified Deletion Security for One-Time SKE).** *Let* $\Sigma = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Del}, \mathsf{Verify})$ *be a secret key encryption with certified deletion. We consider the following security experiment* $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma,\mathcal{A}}(\lambda, b)$*.*

1. *The challenger computes* $\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^\lambda)$*.*
2. $\mathcal{A}$ *sends* $(m_0, m_1) \in \mathcal{M}^2$ *to the challenger.*
3. *The challenger computes* $\mathsf{CT}_b \leftarrow \mathsf{Enc}(\mathsf{sk}, m_b)$ *and sends* $\mathsf{CT}_b$ *to* $\mathcal{A}$*.*
4. $\mathcal{A}$ *sends* $\mathsf{cert}$ *to the challenger.*
5. *The challenger computes* $\mathsf{Verify}(\mathsf{sk},\mathsf{cert})$*. If the output is* $\bot$*, the challenger sends* $\bot$ *to* $\mathcal{A}$*. If the output is* $\top$*, the challenger sends* $\mathsf{sk}$ *to* $\mathcal{A}$*.*
6. $\mathcal{A}$ *outputs* $b' \in \{0,1\}$*.*

*We say that the* $\Sigma$ *is OT-CD secure if for any unbounded* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma,\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma,\mathcal{A}}(\lambda,0) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma,\mathcal{A}}(\lambda,1) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

Broadbent and Islam [BI20] showed that one-time SKE scheme with certified deletion that satisfies the above correctness and security exists unconditionally.

# 3 Commitment with Certified Everlasting Hiding and Classical-Extractor-Based Binding

In this section, we define and construct commitment with certified everlasting hiding and statistical binding. We adopt a non-standard syntax for the verification algorithm and a slightly involved definition for the binding, which we call the classical-extractor-based binding, that are tailored to our construction. This is because they are convenient for our construction of certified everlasting zero-knowledge proof for **QMA** given in Section 4. We can also construct one with a more standard syntax of verification and binding property, namely, the sum-binding, by essentially the same construction. The detail is given in Appendix C.

## 3.1 Definition

**Definition 3.1 (Commitment with Certified Everlasting Hiding and Classical-Extractor-Based Binding (Syntax)).** *Let* $\lambda$ *be the security parameter and let* $p$*,* $q$*,* $r$*,* $s$ *and* $t$ *be some polynomials. Commitment with certified everlasting hiding and classical-extractor-based binding consists of a tuple of algorithms* $(\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$ *with message space* $\mathcal{M} := \{0,1\}^n$*, commitment space* $\mathcal{C} := \mathcal{Q}^{\otimes p(\lambda)} \times \{0,1\}^{q(\lambda)}$*, decommitment space* $\mathcal{D} := \{0,1\}^{r(\lambda)}$*, key space* $\mathcal{K} := \{0,1\}^{s(\lambda)}$ *and deletion certificate space* $\mathcal{E} := \{0,1\}^{t(\lambda)}$*.*

$\mathsf{Commit}(1^\lambda, m) \to (\mathsf{com}, \mathsf{d}, \mathsf{ck})$**:** *The commitment algorithm takes as input a security parameter* $1^\lambda$ *and a message* $m \in \mathcal{M}$*, and outputs a commitment* $\mathsf{com} \in \mathcal{C}$*, a decommitment* $\mathsf{d} := (\mathsf{d}_1, \mathsf{d}_2) \in \mathcal{D}$ *and a key* $\mathsf{ck} \in \mathcal{K}$*. Note that* com *consists of a quantum state* $\psi \in \mathcal{Q}^{\otimes p(\lambda)}$ *and a classical bit string* $f \in \{0,1\}^{q(\lambda)}$*.*

$\mathsf{Verify}(\mathsf{com}, \mathsf{d}) \to m'$ *or* $\bot$**:** *The verification algorithm consists of two algorithms,* $\mathsf{Verify}_1$ *and* $\mathsf{Verify}_2$*. It parses* $\mathsf{d} = (\mathsf{d}_1, \mathsf{d}_2)$*.* $\mathsf{Verify}_1$ *takes* com *and* $(\mathsf{d}_1, \mathsf{d}_2)$ *as input, and outputs* $\top$ *or* $\bot$*.* $\mathsf{Verify}_2$ *takes* com *and* $\mathsf{d}_1$ *as input, and outputs* $m'$*. If the output of* $\mathsf{Verify}_1$ *is* $\bot$*, then the output of* Verify *is* $\bot$*. Otherwise the output of* Verify *is* $m'$*.*

$\mathsf{Del}(\mathsf{com}) \to \mathsf{cert}$**:** *The deletion algorithm takes* com *as input, and outputs a certificate* $\mathsf{cert} \in \mathcal{E}$*.*

$\mathsf{Cert}(\mathsf{cert}, \mathsf{ck}) \to \top$ *or* $\bot$**:** *The certification algorithm takes* cert *and* ck *as input, and outputs* $\top$ *or* $\bot$*.*

**Definition 3.2 (Correctness).** *There are two types of correctness, namely, decommitment correctness and deletion correctness.*

**Decommitment correctness:** *There exists a negligible function* negl *such that for any* $\lambda \in \mathbb{N}$ *and* $m \in \mathcal{M}$*,*

$$\Pr\left[m \leftarrow \mathsf{Verify}(\mathsf{com}, \mathsf{d}) \mid (\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, m)\right] \geq 1 - \mathsf{negl}(\lambda).$$

**Deletion correctness:** *There exists a negligible function* negl *such that for any* $\lambda \in \mathbb{N}$ *and* $m \in \mathcal{M}$*,*

$$\Pr\left[\top \leftarrow \mathsf{Cert}(\mathsf{cert}, \mathsf{ck}) \mid (\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, m), \mathsf{cert} \leftarrow \mathsf{Del}(\mathsf{com})\right] \geq 1 - \mathsf{negl}(\lambda).$$

**Definition 3.3 (Classical-Extractor-Based Binding).** *There exists an unbounded-time deterministic algorithm* Ext *that takes* $f \in \{0,1\}^{q(\lambda)}$ *of* com *as input, and outputs* $\mathsf{d}_1^* \leftarrow \mathsf{Ext}(f)$ *such that for any* com*, any* $\mathsf{d}_1 \neq \mathsf{d}_1^*$*, and any* $\mathsf{d}_2$*,* $\Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d} = (\mathsf{d}_1, \mathsf{d}_2)) = \bot] = 1$*.*

**Definition 3.4 (Computational Hiding).** *Let* $\Sigma := (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$*. Let us consider the following security experiment* $\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, b)$ *against any QPT adversary* $\mathcal{A}$*.*

1. $\mathcal{A}$ *generates* $(m_0, m_1) \in \mathcal{M}^2$ *and sends them to the challenger.*
2. *The challenger computes* $(\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, m_b)$*, and sends* com *to* $\mathcal{A}$*.*
3. $\mathcal{A}$ *outputs* $b' \in \{0,1\}$*.*
4. *The output of the experiment is* $b'$*.*

*Computational hiding means that the following is satisfied for any QPT* $\mathcal{A}$*.*

$$\mathsf{Adv}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 0) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 1) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

**Definition 3.5 (Certified Everlasting Hiding).** *Let* $\Sigma := (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$*. Let us consider the following security experiment* $\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, b)$ *against* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *consisting of any QPT adversary* $\mathcal{A}_1$ *and any unbounded adversary* $\mathcal{A}_2$*.*

1. $\mathcal{A}_1$ *generates* $(m_0, m_1) \in \mathcal{M}^2$ *and sends it to the challenger.*
2. *The challenger computes* $(\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, m_b)$*, and sends* com *to* $\mathcal{A}_1$*.*
3. *At some point,* $\mathcal{A}_1$ *sends* cert *to the challenger, and sends its internal state to* $\mathcal{A}_2$*.*
4. *The challenger computes* $\mathsf{Cert}(\mathsf{cert}, \mathsf{ck})$*. If the output is* $\top$*, then the challenger outputs* $\top$*, and sends* $(\mathsf{d}, \mathsf{ck})$ *to* $\mathcal{A}_2$*. Else, the challenger outputs* $\bot$*, and sends* $\bot$ *to* $\mathcal{A}_2$*.*
5. $\mathcal{A}_2$ *outputs* $b' \in \{0,1\}$*.*
6. *If the challenger outputs* $\top$*, then the output of the experiment is* $b'$*. Otherwise, the output of the experiment is* $\bot$*.*

*We say that it is certified everlasting hiding if the following is satisfied for any $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$.*

$$\mathsf{Adv}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda,0)=1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda,1)=1\right]\right| \leq \mathsf{negl}(\lambda).$$

*Remark* 3.6. We remark that certified everlasting hiding does not imply computational hiding since it does not require anything if the adversary does not send a valid certificate.

The following lemma will be used in the construction of the certified everlasting zero-knowledge proof for **QMA** in Section 4. It is shown with the standard hybrid argument (see Appendix D). It is also easy to see that a similar lemma holds for computational hiding.

**Lemma 3.7.** *Let $\Sigma := (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$ and $\mathcal{M} = \{0,1\}$. Let us consider the following security experiment $\mathsf{Exp}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, b)$ against $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ consisting of any QPT adversary $\mathcal{A}_1$ and any unbounded adversary $\mathcal{A}_2$.*

1. *$\mathcal{A}_1$ generates $(m^0, m^1) \in \{0,1\}^n \times \{0,1\}^n$ and sends it to the challenger.*
2. *The challenger computes*

$$(\mathsf{com}_i(m^b_i), \mathsf{d}_i(m^b_i), \mathsf{ck}_i(m^b_i)) \leftarrow \mathsf{Commit}(1^\lambda, m^b_i)$$

   *for each $i \in [n]$, and sends $\{\mathsf{com}_i(m^b_i)\}_{i\in[n]}$ to $\mathcal{A}_1$. Here, $m^b_i$ is the $i$-th bit of $m^b$.*
3. *At some point, $\mathcal{A}_1$ sends $\{\mathsf{cert}_i\}_{i\in[n]}$ to the challenger, and sends its internal state to $\mathcal{A}_2$.*
4. *The challenger computes $\mathsf{Cert}(\mathsf{cert}_i, \mathsf{ck}_i(m^b_i))$ for each $i \in [n]$. If the output is $\top$ for all $i \in [n]$, then the challenger outputs $\top$, and sends $\{\mathsf{d}_i(m^b_i), \mathsf{ck}_i(m^b_i)\}_{i\in[n]}$ to $\mathcal{A}_2$. Else, the challenger outputs $\bot$, and sends $\bot$ to $\mathcal{A}_2$.*
5. *$\mathcal{A}_2$ outputs $b' \in \{0,1\}$.*
6. *If the challenger outputs $\top$, then the output of the experiment is $b'$. Otherwise, the output of the experiment is $\bot$.*

*If $\Sigma$ is certified everlasting hiding,*

$$\mathsf{Adv}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda,0)=1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda,1)=1\right]\right| \leq \mathsf{negl}(\lambda)$$

*for any $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$.*

## 3.2 Construction

Let $\lambda$ be the security parameter, and let $p$, $q$, $r$, $s$, $t$ and $u$ be some polynomials. We construct commitment with certified everlasting hiding and classical-extractor-based binding, $\Sigma_{\mathsf{ccd}} = (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$, with message space $\mathcal{M} = \{0,1\}^n$, commitment space $\mathcal{C} = \mathcal{Q}^{\otimes p(\lambda)} \times \{0,1\}^{q(\lambda)} \times \{0,1\}^{r(\lambda)}$, decommitment space $\mathcal{D} = \{0,1\}^{s(\lambda)} \times \{0,1\}^{t(\lambda)}$, key space $\mathcal{K} = \{0,1\}^{r(\lambda)}$ and deletion certificate space $\mathcal{E} = \{0,1\}^{u(\lambda)}$ from the following primitives:

- Secret-key encryption with certified deletion, $\Sigma_{\mathsf{skcd}} = \mathsf{SKE}.(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Del}, \mathsf{Verify})$, with plaintext space $\mathcal{M} = \{0,1\}^n$, ciphertext space $\mathcal{C} = \mathcal{Q}^{\otimes p(\lambda)}$, key space $\mathcal{K} = \{0,1\}^{r(\lambda)}$, and deletion certificate space $\mathcal{E} = \{0,1\}^{u(\lambda)}$.
- Classical non-interactive commitment, $\Sigma_{\mathsf{com}} = \mathsf{Classical.Commit}$, with plaintext space $\{0,1\}^{s(\lambda)}$, randomness space $\{0,1\}^{t(\lambda)}$, and commitment space $\{0,1\}^{q(\lambda)}$.
- A hash function $H$ from $\{0,1\}^{s(\lambda)}$ to $\{0,1\}^{r(\lambda)}$ modeled as a quantumly-accessible random oracle.

The construction is as follows.

$\mathsf{Commit}(1^\lambda, m)$**:**

- Generate $\mathsf{ske.sk} \leftarrow \mathsf{SKE.KeyGen}(1^\lambda)$, $R \leftarrow \{0,1\}^{s(\lambda)}$, $R' \leftarrow \{0,1\}^{t(\lambda)}$, and a hash function $H$ from $\{0,1\}^{s(\lambda)}$ to $\{0,1\}^{r(\lambda)}$.
- Compute $\mathsf{ske.CT} \leftarrow \mathsf{SKE.Enc}(\mathsf{ske.sk}, m)$, $f \leftarrow \mathsf{Classical.Commit}(R; R')$, and $h := H(R) \oplus \mathsf{ske.sk}$.
- Output $\mathsf{com} := (\mathsf{ske.CT}, f, h)$, $\mathsf{d}_1 := R$, $\mathsf{d}_2 := R'$, and $\mathsf{ck} := \mathsf{ske.sk}$.

$\mathsf{Verify}_1(\mathsf{com}, \mathsf{d}_1, \mathsf{d}_2)$**:**

- Parse $\mathsf{com} = (\mathsf{ske.CT}, f, h)$, $\mathsf{d}_1 = R$, and $\mathsf{d}_2 = R'$.
- Output $\top$ if $f = \mathsf{Classical.Commit}(R; R')$, and output $\bot$ otherwise.

$\mathsf{Verify}_2(\mathsf{com}, \mathsf{d}_1)$**:**

- Parse $\mathsf{com} = (\mathsf{ske.CT}, f, h)$ and $\mathsf{d}_1 = R$.
- Compute $\mathsf{ske.sk}' := H(R) \oplus h$.
- Output $m' \leftarrow \mathsf{SKE.Dec}(\mathsf{ske.sk}', \mathsf{ske.CT})$.

$\mathsf{Del}(\mathsf{com})$**:**

- Parse $\mathsf{com} = (\mathsf{ske.CT}, f, h)$.
- Compute $\mathsf{ske.cert} \leftarrow \mathsf{SKE.Del}(\mathsf{ske.CT})$.
- Output $\mathsf{cert} := \mathsf{ske.cert}$.

$\mathsf{Cert}(\mathsf{cert}, \mathsf{ck})$**:**

- Parse $\mathsf{cert} = \mathsf{ske.cert}$ and $\mathsf{ck} = \mathsf{ske.sk}$.
- Output $\top/\bot \leftarrow \mathsf{SKE.Verify}(\mathsf{ske.sk}, \mathsf{ske.cert})$.

**Correctness.** The decommitment and deletion correctness easily follow from the correctness of $\Sigma_{\mathsf{skcd}}$.

**Security.** We prove the following three theorems.

**Theorem 3.8.** *If $\Sigma_{\mathsf{com}}$ is perfect binding, then $\Sigma_{\mathsf{ccd}}$ is classical-extractor-based binding.*

**Theorem 3.9.** *If $\Sigma_{\mathsf{com}}$ is unpredictable and $\Sigma_{\mathsf{skcd}}$ is OT-CD secure, then $\Sigma_{\mathsf{ccd}}$ is certified everlasting hiding.*

**Theorem 3.10.** *If $\Sigma_{\mathsf{com}}$ is unpredictable and $\Sigma_{\mathsf{skcd}}$ is OT-CD secure, then $\Sigma_{\mathsf{ccd}}$ is computationally hiding.*

*Proof of Theorem 3.8.* Due to the perfect binding of $\Sigma_{\mathsf{com}} = \mathsf{Classical.Commit}$, there exists a unique $\mathsf{d}_1^*$ such that $f = \mathsf{Classical.Commit}(\mathsf{d}_1^*; \mathsf{d}_2)$ for a given $f$. Let $\mathsf{Ext}$ be the algorithm that finds such $\mathsf{d}_1^*$ and outputs it. (If there is no such $\mathsf{d}_1^*$, then $\mathsf{Ext}$ outputs $\bot$.) Then, for any $\mathsf{com} = (\mathsf{ske.CT}, f, h)$, any $\mathsf{d}_1 \neq \mathsf{d}_1^*$, and any $\mathsf{d}_2$,

$$\Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d} = (\mathsf{d}_1, \mathsf{d}_2)) = \bot] \geq \Pr[f \neq \mathsf{Classical.Commit}(\mathsf{d}_1, \mathsf{d}_2)] = 1,$$

which completes the proof. □

*Proof of Theorem 3.9.* For clarity, we describe how the experiment works against an adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$ consisting of any QPT adversary $\mathcal{A}_1$ and any quantum unbounded time adversary $\mathcal{A}_2$.

$\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}}, \mathcal{A}}(\lambda, b)$**:** This is the original experiment.

1. A uniformly random function $H$ from $\{0,1\}^{s(\lambda)}$ to $\{0,1\}^{r(\lambda)}$ is chosen. $\mathcal{A}_1$ and $\mathcal{A}_2$ can make arbitrarily many quantum queries to $H$ at any time in the experiment.

2. $\mathcal{A}_1$ chooses $(m_0, m_1) \leftarrow \mathcal{M}^2$, and sends $(m_0, m_1)$ to the challenger.
3. The challenger generates $\mathsf{ske.sk} \leftarrow \mathsf{SKE.KeyGen}(1^\lambda)$, $R \leftarrow \{0,1\}^{s(\lambda)}$ and $R' \leftarrow \{0,1\}^{t(\lambda)}$. The challenger computes $\mathsf{ske.CT} \leftarrow \mathsf{SKE.Enc}(\mathsf{ske.sk}, m_b)$, $f := \mathsf{Classical.Commit}(R; R')$ and $h := H(R) \oplus \mathsf{ske.sk}$, and sends $(\mathsf{ske.CT}, f, h)$ to $\mathcal{A}_1$.
4. $\mathcal{A}_1$ sends $\mathsf{ske.cert}$ to the challenger and sends its internal state to $\mathcal{A}_2$.
5. If $\top \leftarrow \mathsf{SKE.Verify}(\mathsf{ske.sk}, \mathsf{ske.cert})$, the challenger outputs $\top$ and sends $(R, R', \mathsf{ske.sk})$ to $\mathcal{A}_2$. Otherwise, the challenger outputs $\bot$ and sends $\bot$ to $\mathcal{A}_2$.
6. $\mathcal{A}_2$ outputs $b'$.
7. If the challenger outputs $\top$, then the output of the experiment is $b'$. Otherwise, the output of the experiment is $\bot$.

What we have to prove is that

$$\mathsf{Adv}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, 0) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, 1) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

We define the following sequence of hybrids.

$\mathsf{Hyb}_1(b)$**:** This is identical to $\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, b)$ except that the oracle given to $\mathcal{A}_1$ is replaced with $H_{R \to H'}$ which is $H$ reprogrammed according to $H'$ on an input $R$ where $H'$ is another independent uniformly random function. More formally, $H_{R \to H'}$ is defined by

$$H_{R \to H'}(R^*) := \begin{cases} H(R^*) & (R^* \neq R) \\ H'(R^*) & (R^* = R). \end{cases}$$

We note that the challenger still uses $H$ to generate $h$, and the oracle which $\mathcal{A}_2$ uses is still $H$ similarly to the original experiment.

$\mathsf{Hyb}_2(b)$**:** This is identical to $\mathsf{Hyb}_1(b)$ except for the following three points. First, the challenger generates $h$ uniformly at random. Second, the oracle given to $\mathcal{A}_1$ is replaced with $H'$ which is an independent uniformly random function. Third, the oracle given to $\mathcal{A}_2$ is replaced with $H'_{R \to h \oplus \mathsf{ske.sk}}$ which is $H'$ reprogrammed to $h \oplus \mathsf{ske.sk}$ on an input $R$. More formally, $H'_{R \to h \oplus \mathsf{ske.sk}}$ is defined by

$$H'_{R \to h \oplus \mathsf{ske.sk}}(R^*) := \begin{cases} H'(R^*) & (R^* \neq R) \\ h \oplus \mathsf{ske.sk} & (R^* = R). \end{cases}$$

**Proposition 3.11.** *If $\Sigma_{\mathsf{com}}$ is unpredictable, then*

$$\left|\Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, b) = 1\right] - \Pr[\mathsf{Hyb}_1(b) = 1]\right| \leq \mathsf{negl}(\lambda).$$

*Proof.* The proof is similar to [HMNY21, Propositoin 5.8], but note that this time we have to consider an unbounded adversary after the certificate is issued unlike the case of [HMNY21]. We assume that $\left|\Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, b) = 1\right] - \Pr[\mathsf{Hyb}_1(b) = 1]\right|$ is non-negligible, and construct an adversary $\mathcal{B}$ that breaks the unpredictability of $\Sigma_{\mathsf{com}}$. For notational simplicity, we denote $\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, b)$ by $\mathsf{Hyb}_0(b)$. We consider an algorithm $\widetilde{\mathcal{A}}$ that works as follows. $\widetilde{\mathcal{A}}$ is given an oracle $\mathcal{O}$, which is either $H$ or $H_{R \to H'}$, and an input $z$ that consists of $R$ and the whole truth table of $H$, where $R \leftarrow \{0,1\}^{s(\lambda)}$, and $H$ and $H'$ are uniformly random functions. $\widetilde{\mathcal{A}}$ runs $\mathsf{Hyb}_0(b)$ except that it uses its own oracle $\mathcal{O}$ to simulate $\mathcal{A}_1$'s random oracle queries. On the other hand, $\widetilde{\mathcal{A}}$ uses $H$ to simulate $h$ and $\mathcal{A}_2$'s random oracle queries regardless of $\mathcal{O}$, which is possible because the truth table of $H$ is included in the input $z$. By definition, we have

$$\Pr[\mathsf{Hyb}_0(b) = 1] = \Pr\left[\widetilde{\mathcal{A}}^H(R, H) = 1\right]$$

and

$$\Pr[\mathsf{Hyb}_1(b)=1]=\Pr\left[\widetilde{\mathcal{A}}^{H_{R\to H'}}(R,H)=1\right]$$

where $H$ in the input means the truth table of $H$. We apply the one-way to hiding lemma (Lemma 2.2) to the above $\widetilde{\mathcal{A}}$. Note that $\widetilde{\mathcal{A}}$ is inefficient, but the one-way to hiding lemma is applicable to inefficient algorithms. Then if we let $\widetilde{\mathcal{B}}$ be the algorithm that measures uniformly chosen query of $\widetilde{\mathcal{A}}$, we have

$$\left|\Pr\left[\widetilde{\mathcal{A}}^{H}(R,H)=1\right]-\Pr\left[\widetilde{\mathcal{A}}^{H_{R\to H'}}(R,H)=1\right]\right|\le 2q\sqrt{\Pr\left[\widetilde{\mathcal{B}}^{H_{R\to H'}}(R,H)=R\right]}.$$

By the assumption, the LHS is non-negligible, and thus $\Pr\left[\widetilde{\mathcal{B}}^{H_{R\to H'}}(R,H)=R\right]$ is non-negligible.

Let $\widetilde{\mathcal{B}}'$ be the algorithm that is the same as $\widetilde{\mathcal{B}}$ except that it does not take the truth table of $H$ as input, and sets $h$ to be uniformly random string instead of setting $h:=H(R)\oplus\mathsf{ske.sk}$. Then we have

$$\Pr\left[\widetilde{\mathcal{B}}^{H_{R\to H'}}(R,H)=R\right]=\Pr\left[\widetilde{\mathcal{B}}'^{H_{R\to H'}}(R)=R\right].$$

The reason is as follows: First, $\widetilde{\mathcal{B}}$ uses the truth table of $H$ only for generating $h:=H(R)\oplus\mathsf{ske.sk}$, because it halts before $\widetilde{\mathcal{B}}$ simulates $\mathcal{A}_2$. Second, the oracle $H_{R\to H'}$ reveals no information about $H(R)$, and thus $h$ can be independently and uniformly random.

Moreover, for any fixed $R$, when $H$ and $H'$ are uniformly random, $H_{R\to H'}$ is also a uniformly random function, and therefore we have

$$\Pr\left[\widetilde{\mathcal{B}}'^{H_{R\to H'}}(R)=R\right]=\Pr\left[\widetilde{\mathcal{B}}'^{H}(R)=R\right].$$

Since $\Pr\left[\widetilde{\mathcal{B}}^{H_{R\to H'}}(R,H)=R\right]$ is non-negligible, $\Pr\left[\widetilde{\mathcal{B}}'^{H}(R)=R\right]$ is also non-negligible. Recall that $\widetilde{\mathcal{B}}'^{H}$ is an algorithm that simulates $\mathsf{Hyb}_0(b)$ with the modification that $h$ is set to be uniformly random and measures randomly chosen $\mathcal{A}_1$'s query. Then it is straightforward to construct an adversary $\mathcal{B}$ that breaks the unpredictability of $\Sigma_{\mathsf{com}}$ by using $\widetilde{\mathcal{B}}'$. For clarity, let us give the description of $\mathcal{B}$ as follows.

$\mathcal{B}$ is given $\mathsf{Classical.Commit}(R;R')$ from the challenger of $\mathsf{Exp}^{\mathsf{unpre}}_{\Sigma_{\mathsf{com}},\mathcal{B}}(\lambda)$. $\mathcal{B}$ chooses $i\leftarrow[q]$ and runs $\mathsf{Hyb}_1(b)$ until $\mathcal{A}_1$ makes $i$-th random oracle query or $\mathcal{A}_1$ sends the internal state to $\mathcal{A}_2$, where $\mathcal{B}$ embeds the problem instance $\mathsf{Classical.Commit}(R;R')$ into those sent to $\mathcal{A}_1$ instead of generating it by itself. $\mathcal{B}$ measures the $i$-th random oracle query by $\mathcal{A}_1$, and outputs the measurement outcome. Note that $\mathcal{B}$ can efficiently simulate the random oracle $H$ by Zhandry's compressed oracle technique [Zha19]. It is clear that the probability that $\mathcal{B}$ outputs $R$ is exactly $\Pr\left[\widetilde{\mathcal{B}}'^{H}(R)=R\right]$, which is non-negligible. This contradicts the unpredictability of $\Sigma_{\mathsf{com}}$. Therefore $|\Pr[\mathsf{Hyb}_0(b)=1]-\Pr[\mathsf{Hyb}_1(b)=1]|$ is negligible. □

**Proposition 3.12.** $\Pr[\mathsf{Hyb}_1(b)=1]=\Pr[\mathsf{Hyb}_2(b)=1]$.

*Proof.* First, let us remind the difference between $\mathsf{Hyb}_1(b)$ and $\mathsf{Hyb}_2(b)$. In $\mathsf{Hyb}_1(b)$, $\mathcal{A}_1$ receives $h=\mathsf{ske.sk}\oplus H(R)$. Moreover, $\mathcal{A}_1$ can access to the random oracle $H_{R\to H'}$, and $\mathcal{A}_2$ can access to the random oracle $H$. On the other hand, in $\mathsf{Hyb}_2(b)$, $\mathcal{A}_1$ receives uniformly random $h$. Moreover, $\mathcal{A}_1$ can access to the random oracle $H'$ instead of $H_{R\to H'}$, and $\mathcal{A}_2$ can access to the random oracle $H'_{R\to h\oplus\mathsf{ske.sk}}$ instead of $H$.

Let $\Pr[(h,H_{R\to H'},H)=(r,G,G')\mid\mathsf{Hyb}_1(b)]$ be the probability that the adversary $\mathcal{A}_1$ in $\mathsf{Hyb}_1(b)$ receives a classical bit string $r$ as $h$, random oracle which $\mathcal{A}_1$ can access to is $G$, and random oracle which $\mathcal{A}_2$ can access to is $G'$. Similarly, let us define $\Pr\left[(h,H',H'_{R\to h\oplus\mathsf{ske.sk}})=(r,G,G')\mid\mathsf{Hyb}_2(b)\right]$ for $\mathsf{Hyb}_2(b)$. What we have to show is that the following equation holds for any $(r,G,G')$

$$\Pr[(h,H_{R\to H'},H)=(r,G,G')\mid\mathsf{Hyb}_1(b)]=\Pr\left[(h,H',H'_{R\to h\oplus\mathsf{ske.sk}})=(r,G,G')\mid\mathsf{Hyb}_2(b)\right].$$

Since $h=\mathsf{ske.sk}\oplus H(R)$ in $\mathsf{Hyb}_1(b)$, $H$ is a uniformly random function, and $h$ in $\mathsf{Hyb}_2(b)$ is uniformly generated,

$$\Pr[h=r\mid\mathsf{Hyb}_1(b)]=\Pr[h=r\mid\mathsf{Hyb}_2(b)]$$

holds for any $r$.

For any classical bit string $r$ and any random oracle $G$, we have

$$\Pr[H_{R\to H'} = G \mid h = r, \mathsf{Hyb}_1(b)] = \Pr[H' = G \mid h = r, \mathsf{Hyb}_2(b)].$$

This is shown as follows. First, in $\mathsf{Hyb}_1(b)$, from the construction of $H_{R\to H'}$, $H_{R\to H'}(R)$ is independent from $h$ for any $R \in \{0,1\}^{s(\lambda)}$. Furthermore, since $H$ and $H'$ is random oracle, $H_{R\to H'}(R)$ is uniformly random for any $R \in \{0,1\}^{s(\lambda)}$. Second, in $\mathsf{Hyb}_2(b)$, from the construction of $H'$, $H'(R)$ is independent from $h$ for any $R \in \{0,1\}^{s(\lambda)}$. Furthermore, since $H'$ is random oracle, $H'(R)$ is uniformly random for any $R \in \{0,1\}^{s(\lambda)}$. Therefore, we have the above equation.

For any classical bit string $r$ and any random oracles $G$ and $G'$, we have

$$\Pr[H = G' \mid (h, H_{R\to H'}) = (r, G), \mathsf{Hyb}_1(b)] = \Pr\big[H'_{R\to h\oplus \mathsf{ske.sk}} = G' \mid (h, H') = (r, G), \mathsf{Hyb}_2(b)\big].$$

This can be shown as follows. First, in $\mathsf{Hyb}_1(b)$, we obtain $H(R) = r \oplus \mathsf{ske.sk}$, because $h := \mathsf{ske.sk} \oplus H(R)$ and $h = r$. Furthermore, from the definition of $H_{R\to H'}$, we obtain $H(R^*) = G(R^*)$ for $R^* \neq R$. Second, in $\mathsf{Hyb}_2(b)$, from the definition of $H'_{R\to h\oplus \mathsf{ske.sk}}$, we have $H'_{R\to h\oplus \mathsf{ske.sk}}(R) = r \oplus \mathsf{ske.sk}$ and $H'_{R\to h\oplus \mathsf{ske.sk}}(R^*) = G(R^*)$ for $R^* \neq R$.

From all above discussions, we have

$$\Pr[(h, H_{R\to H'}, H) = (r, G, G') \mid \mathsf{Hyb}_1(b)] = \Pr\big[(h, H', H'_{R\to h\oplus \mathsf{ske.sk}}) = (r, G, G') \mid \mathsf{Hyb}_2(b)\big].$$

□

**Proposition 3.13.** *If $\Sigma_{\mathsf{skcd}}$ is OT-CD secure, then*

$$|\Pr[\mathsf{Hyb}_2(1) = 1] - \Pr[\mathsf{Hyb}_2(0) = 1]| \leq \mathsf{negl}(\lambda).$$

*Proof.* To show this, we assume that $|\Pr[\mathsf{Hyb}_2(1) = 1] - \Pr[\mathsf{Hyb}_2(0) = 1]|$ is non-negligible, and construct an adversary $\mathcal{B}$ that breaks the OT-CD security of $\Sigma_{\mathsf{skcd}}$.

$\mathcal{B}$ plays the experiment $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b')$ for some $b' \in \{0,1\}$. First, $\mathcal{B}$ sends $(m_0, m_1) \in \mathcal{M}^2$ to the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b')$. $\mathcal{B}$ receives $\mathsf{ske.CT}$ from the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b')$. $\mathcal{B}$ generates $R \leftarrow \{0,1\}^{s(\lambda)}$, $R' \leftarrow \{0,1\}^{t(\lambda)}$ and $h \leftarrow \{0,1\}^{r(\lambda)}$, and computes $f := \mathsf{Classical.Commit}(R; R')$. $\mathcal{B}$ sends $(\mathsf{ske.CT}, f, h)$ to $\mathcal{A}_1$. $\mathcal{B}$ simulates the random oracle $H'$ given to $\mathcal{A}_1$ by itself. At some point, $\mathcal{A}_1$ sends $\mathsf{ske.cert}$ to $\mathcal{B}$, and sends the internal state to $\mathcal{A}_2$. $\mathcal{B}$ passes $\mathsf{ske.cert}$ to the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b')$.

The challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b')$ runs $\mathsf{SKE.Verify}(\mathsf{ske.sk}, \mathsf{ske.cert}) \to \top/\bot$. If it is $\top$, the challenger sends $\mathsf{ske.sk}$ to $\mathcal{B}$. In that case, $\mathcal{B}$ outputs $\top$, and sends $(R, R', \mathsf{ske.sk})$ to $\mathcal{A}_2$. We denote this event by $\mathsf{Reveal}_{\mathsf{sk}}(b')$. $\mathcal{B}$ simulates $\mathcal{A}_2$, and outputs the output of $\mathcal{A}_2$. On the other hand, if $\mathsf{SKE.Verify}(\mathsf{ske.sk}, \mathsf{ske.cert}) \to \bot$, then the challenger sends $\bot$ to $\mathcal{B}$. In that case, $\mathcal{B}$ outputs $\bot$ and aborts. Note that $\mathcal{B}$ can simulate the random oracle $H'_{R\to h\oplus \mathsf{ske.sk}}$ given to $\mathcal{A}_2$ when $\mathcal{B}$ does not abort, because $\mathcal{B}$ receives $\mathsf{ske.sk}$ from the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b')$ when $\mathcal{B}$ does not abort.

Now we have

$$\begin{aligned}
&\mathsf{Adv}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda)\\
&:= \left|\Pr\left[\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b') = 1 \mid b' = 0\right] - \Pr\left[\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b') = 1 \mid b' = 1\right]\right|\\
&= |\Pr[\mathcal{B} = 1 \wedge \mathsf{Reveal}_{\mathsf{sk}}(b') \mid b' = 0] - \Pr[\mathcal{B} = 1 \wedge \mathsf{Reveal}_{\mathsf{sk}}(b') \mid b' = 1]|\\
&= |\Pr[\mathcal{A}_2 = 1 \wedge \mathsf{Reveal}_{\mathsf{sk}}(b') \mid b' = 0] - \Pr[\mathcal{A}_2 = 1 \wedge \mathsf{Reveal}_{\mathsf{sk}}(b') \mid b' = 1]|\\
&= |\Pr[\mathsf{Hyb}_2(b') = 1 \wedge \mathsf{Reveal}_{\mathsf{sk}}(b') \mid b' = 0] - \Pr[\mathsf{Hyb}_2(b') = 1 \wedge \mathsf{Reveal}_{\mathsf{sk}}(b') \mid b' = 1]|\\
&= |\Pr[\mathsf{Hyb}_2(b') = 1 \mid b' = 0] - \Pr[\mathsf{Hyb}_2(b') = 1 \mid b' = 1]|\\
&= |\Pr[\mathsf{Hyb}_2(0) = 1] - \Pr[\mathsf{Hyb}_2(1) = 1]|.
\end{aligned}$$

In the second equation, we have used the fact that $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b) = 1$ if and only if $\mathcal{B} = 1$ and the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda, b')$ outputs $\top$. In the third equation, we have used the fact that the output of $\mathcal{B}$ is equal to the output of

$\mathcal{A}_2$ conditioned that $\mathsf{Reveal}_{\mathsf{sk}}(b')$ occurs. In the fourth equation, we have used the fact that $\mathcal{B}$ simulates the challenger of $\mathsf{Hyb}_2(b)$ when $\mathsf{Reveal}_{\mathsf{sk}}(b)$ occurs. In the fifth equation, we have used the fact that $\mathsf{Hyb}_2(b) = 1$ only when $\mathsf{Reveal}_{\mathsf{sk}}(b)$ occurs. Since $|\Pr[\mathsf{Hyb}_2(0) = 1] - \Pr[\mathsf{Hyb}_2(1) = 1]|$ is non-negligible, $\mathsf{Adv}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda)$ is non-negligible. This contradicts the OT-CD security of $\Sigma_{\mathsf{skcd}}$. □

By Propositions 3.11 to 3.13, we immediately obtain Theorem 3.9.

□

*Proof of Theorem 3.10.* For clarity, we describe how the experiment works against a QPT adversary $\mathcal{A}$.

$\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, b)$**:** This is the original experiment.

1. A uniformly random function $H$ from $\{0,1\}^{s(\lambda)}$ to $\{0,1\}^{r(\lambda)}$ is chosen, and $\mathcal{A}$ can make arbitrarily quantum queries to $H$ at any time in the experiment.
2. $\mathcal{A}$ chooses $(m_0, m_1) \leftarrow \mathcal{M}^2$, and sends $(m_0, m_1)$ to the challenger.
3. The challenger generates $\mathsf{ske.sk} \leftarrow \mathsf{SKE.KeyGen}(1^\lambda)$, $R \leftarrow \{0,1\}^{s(\lambda)}$ and $R' \leftarrow \{0,1\}^{t(\lambda)}$. The challenger computes $\mathsf{ske.CT} \leftarrow \mathsf{SKE.Enc}(\mathsf{ske.sk}, m_b)$, $f := \mathsf{Classical.Commit}(R; R')$ and $h := H(R) \oplus \mathsf{ske.sk}$, and sends $(\mathsf{ske.CT}, f, h)$ to $\mathcal{A}$.
4. $\mathcal{A}$ outputs $b'$. The output of the experiment is $b'$.

Note that what we have to prove is

$$\mathsf{Adv}^{\mathsf{c\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}} := \left|\Pr\left[\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, 0) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, 1) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

We define the following sequence of hybrids.

$\mathsf{Hyb}_1(b)$**:** This is identical to $\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, b)$ except that the oracle given to $\mathcal{A}$ is replaced with $H_{R\to H'}$ which is $H$ reprogrammed according to $H'$ on an input $R$ where $H'$ is another independent random function. More formally, $H_{R\to H'}$ is defined by

$$H_{R\to H'}(R^*) := \begin{cases} H(R^*) & (R^* \neq R) \\ H'(R^*) & (R^* = R). \end{cases}$$

We note that the challenger still uses $H$ to generate $h$.

$\mathsf{Hyb}_2(b)$**:** This is identical to $\mathsf{Hyb}_1(b)$ except that the challenger generates $h$ uniformly random.

**Proposition 3.14.** *If $\Sigma_{\mathsf{com}}$ is unpredictable, then*

$$\left|\Pr\left[\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{A}}(\lambda, b) = 1\right] - \Pr[\mathsf{Hyb}_1(b) = 1]\right| \leq \mathsf{negl}(\lambda).$$

*Proof.* It is the same as that of Proposition 3.11. □

**Proposition 3.15.** $\Pr[\mathsf{Hyb}_1(b) = 1] = \Pr[\mathsf{Hyb}_2(b) = 1]$.

*Proof.* This is similar to the proof of Proposition 3.12. For clarity, we describe the proof. The difference between $\mathsf{Hyb}_1(b)$ and $\mathsf{Hyb}_2(b)$ is as follows. In $\mathsf{Hyb}_1(b)$, $\mathcal{A}$ receives $h := H(R) \oplus \mathsf{ske.sk}$. In $\mathsf{Hyb}_2(b)$, $\mathcal{A}$ receives a uniformly random $h$. In $\mathsf{Hyb}_1(b)$, $h$ is uniformly random since $H(R)$ is uniformly distributed. Therefore, the probability distribution that $\mathcal{A}$ in $\mathsf{Hyb}_1(b)$ receives $h$ is equal to the probability distribution that $\mathcal{A}$ in $\mathsf{Hyb}_2(b)$ receives $h$. This completes the proof. □

**Proposition 3.16.** *If $\Sigma_{\mathsf{skcd}}$ is OT-CD secure, then*

$$|\Pr[\mathsf{Hyb}_2(0) = 1] - \Pr[\mathsf{Hyb}_2(1) = 1]| \leq \mathsf{negl}(\lambda).$$

*Proof.* To show this, we assume that $|\Pr[\mathsf{Hyb}_2(1)=1]-\Pr[\mathsf{Hyb}_2(0)=1]|$ is non-negligible, and construct an adversary $\mathcal{B}$ that breaks the OT-CD security of $\Sigma_{\mathsf{skcd}}$.

First, $\mathcal{B}$ sends $(m_0,m_1)\in\mathcal{M}^2$ to the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda,b')$. $\mathcal{B}$ receives $\mathsf{ske.CT}$ from the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda,b')$ and generates $R\leftarrow\{0,1\}^{s(\lambda)}$, $R'\leftarrow\{0,1\}^{t(\lambda)}$, $f:=\mathsf{Classical.Commit}(R;R')$ and $h\leftarrow\{0,1\}^{r(\lambda)}$. $\mathcal{B}$ sends $(\mathsf{ske.CT},f,h)$ to $\mathcal{A}$. $\mathcal{B}$ simulates the random oracle given to $\mathcal{A}$.

- If $b'=0$, $\mathcal{B}$ simulates the challenger of $\mathsf{Hyb}_2(0)$.
- If $b'=1$, $\mathcal{B}$ simulates the challenger of $\mathsf{Hyb}_2(1)$.

Thus, if $\mathcal{A}$ distinguishes the two experiments, $\mathcal{B}$ breaks the OT-CD security of $\Sigma_{\mathsf{skcd}}$ by generating $\mathsf{ske.cert}$ and sends it to the challenger of $\mathsf{Exp}^{\mathsf{otsk\text{-}cert\text{-}del}}_{\Sigma_{\mathsf{skcd}},\mathcal{B}}(\lambda,b')$. This completes the proof. □

By Propositions 3.14 to 3.16, we immediately obtain Theorem 3.10. □

# 4 Certified Everlasting Zero-Knowledge Proof for QMA

In this section, we define and construct the certified everlasting zero-knowledge proof for **QMA**. In Section 4.1, we define the certified everlasting zero-knowledge proof for **QMA**. We then construct a three round protocol with completeness-soundness gap $\frac{1}{\mathrm{poly}(\lambda)}$ in Section 4.2, and finally amplify the gap to $1-\mathsf{negl}(\lambda)$ with the sequential repetition in Section 4.3.

## 4.1 Definition

We first define a quantum interactive protocol. Usually, in zero-knowledge proofs or arguments, we do not consider prover's output. However, in this paper, we also consider prover's output, because we are interested in the certified everlasting zero-knowledge. Furthermore, in this paper, we consider only an interactive proof, which means that a malicious prover is unbounded.

**Definition 4.1 (Quantum Interactive Protocol).** *A quantum interactive protocol is modeled as an interaction between QPT machines $\mathcal{P}$ referred as a prover and $\mathcal{V}$ referred as a verifier. We denote by $\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x)$ an execution of the protocol where $x$ is a common input, $x_P$ is $\mathcal{P}$'s private input, and $x_V$ is $\mathcal{V}$'s private input. We denote by $\mathrm{OUT}_{\mathcal{V}}\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x)$ the final output of $\mathcal{V}$ in the execution. An honest verifier's output is $\top$ indicating acceptance or $\bot$ indicating rejection, and a malicious verifier's output is an arbitrary quantum state. We denote by $\mathrm{OUT}_{\mathcal{P}}\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x)$ the final output of $\mathcal{P}$ in the execution. An honest prover's output is $\top$ indicating acceptance or $\bot$ indicating rejection. We also define $\mathrm{OUT}'_{\mathcal{P},\mathcal{V}}\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x)$ by*

$$\mathrm{OUT}'_{\mathcal{P},\mathcal{V}}\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x):=\begin{cases}(\top,\mathrm{OUT}_{\mathcal{V}}\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x)) & (\mathrm{OUT}_{\mathcal{P}}\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x)=\top)\\ (\bot,\bot) & (\mathrm{OUT}_{\mathcal{P}}\langle\mathcal{P}(x_P),\mathcal{V}(x_V)\rangle(x)\neq\top).\end{cases}$$

We next define a computational zero-knowledge proof for **QMA**, which is the standard definition.

**Definition 4.2 (Computational Zero-Knowledge Proof for QMA).** *A $c$-complete $s$-sound computational zero-knowledge proof for a **QMA** promise problem $A=(A_{\mathsf{yes}},A_{\mathsf{no}})$ is a quantum interactive protocol between a QPT prover $\mathcal{P}$ and a QPT verifier $\mathcal{V}$ that satisfies the followings:*

***$c$*-completeness:** *For any $\mathsf{x}\in A_{\mathsf{yes}}$ and any $\mathsf{w}\in R_A(\mathsf{x})$,*

$$\Pr\left[\mathsf{Out}_{\mathcal{V}}\langle\mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}),\mathcal{V}\rangle(\mathsf{x})=\top\right]\geq c$$

*for some polynomial $k$.*

**$s$-soundness:** *For any* $\mathsf{x} \in A_{\mathsf{no}}$ *and any unbounded-time prover* $\mathcal{P}^*$*,*

$$\Pr[\mathsf{Out}_{\mathcal{V}}\langle \mathcal{P}^*, \mathcal{V}\rangle(\mathsf{x}) = \top] \leq s.$$

**Computational zero-knowledge:** *There exists a QPT algorithm* $\mathcal{S}$ *such that*

$$\mathsf{OUT}_{\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x}) \approx_c \mathcal{S}(\mathsf{x}, \mathcal{V}^*, \cdot\,)$$

*for any QPT malicious verifier* $\mathcal{V}^*$*, any* $\mathsf{x} \in A_{\mathsf{yes}} \cap \{0,1\}^\lambda$*, any* $\mathsf{w} \in R_A(\mathsf{x})$*, and some polynomial* $k$*. Note that* $\mathsf{OUT}_{\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$ *and* $\mathcal{S}(\mathsf{x}, \mathcal{V}^*, \cdot\,)$ *are quantum channels that map any quantum state* $\xi$ *to quantum states* $\mathsf{OUT}_{\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\xi)\rangle(\mathsf{x})$ *and* $\mathcal{S}(\mathsf{x}, \mathcal{V}^*, \xi)$*, respectively.*

*We just call it a computational zero-knowledge proof if it satisfies* $(1 - \mathsf{negl}(|\mathsf{x}|))$*-completeness,* $\mathsf{negl}(|\mathsf{x}|)$*-soundness, and computational zero-knowledge.*

We finally define a certified everlasting zero-knowledge proof for **QMA**, which is the main target of this paper.

**Definition 4.3 (Certified Everlasting Zero-Knowledge Proof for QMA).** *A certified everlasting zero-knowledge proof for a* ***QMA*** *promise problem* $A = (A_{\mathsf{yes}}, A_{\mathsf{no}})$ *is a computational zero-knowledge proof for* $A$ *(Definition 4.2) that additionally satisfies the followings:*

**Prover's completeness:** $\Pr\left[\mathrm{OUT}_{\mathcal{P}}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}\rangle(\mathsf{x}) = \top\right] \geq 1 - \mathsf{negl}(\lambda)$ *for any* $\mathsf{x} \in A_{\mathsf{yes}} \cap \{0,1\}^\lambda$ *and any* $\mathsf{w} \in R_A(\mathsf{x})$.

**Certified everlasting zero-knowledge:** *There exists a QPT algorithm* $\mathcal{S}$ *such that*

$$\mathrm{OUT}'_{\mathcal{P},\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x}) \approx_s \mathcal{S}(\mathsf{x}, \mathcal{V}^*, \cdot\,)$$

*for any QPT malicious verifier* $\mathcal{V}^*$*, any* $\mathsf{x} \in A_{\mathsf{yes}} \cap \{0,1\}^\lambda$*, any* $\mathsf{w} \in R_A(\mathsf{x})$*, and some polynomial* $k$*. Note that* $\mathsf{OUT}'_{\mathcal{P},\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$ *and* $\mathcal{S}(\mathsf{x}, \mathcal{V}^*, \cdot\,)$ *are quantum channels that map any quantum state* $\xi$ *to quantum states* $\mathsf{OUT}'_{\mathcal{P},\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\xi)\rangle(\mathsf{x})$ *and* $\mathcal{S}(\mathsf{x}, \mathcal{V}^*, \xi)$*, respectively.*

*Remark* 4.4. We remark that certified everlasting zero-knowledge does not imply computational zero-knowledge since it does not require anything if the prover does not output $\top$.

## 4.2 Construction of Three Round Protocol

In this section, we construct a three round protocol with completeness-soundness gap $\frac{1}{\mathrm{poly}(\lambda)}$. In the next section, we will amplify its completeness-soundness gap by the sequential repetition.

In the following, $n$, $m$, $\Pi_c$, $\rho_{\mathsf{hist}}$, and $\rho_{\mathsf{Sim}}^{\mathsf{x},S}$ are given in Definition 2.4. Let $S_c \subseteq [n]$ be the set of qubits on which $\Pi_c$ acts non-trivially. The three round protocol $\Sigma_{\Xi\mathsf{cd}}$ is constructed from commitment with certified everlasting hiding and classical-extractor-based binding, $\Sigma_{\mathsf{ccd}} = (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$.

**The first action by the prover (commitment phase):**

- Generate $x, z \leftarrow \{0,1\}^n$.
- Compute

$$(\mathsf{com}_i(x_i), \mathsf{d}_i(x_i), \mathsf{ck}_i(x_i)) \leftarrow \mathsf{Commit}(1^\lambda, x_i)$$
$$(\mathsf{com}_i(z_i), \mathsf{d}_i(z_i), \mathsf{ck}_i(z_i)) \leftarrow \mathsf{Commit}(1^\lambda, z_i)$$

  for all $i \in [n]$.
- Generate a simulatable witness $\rho_{\mathsf{hist}}$ for the instance $\mathsf{x}$ and generate $X^x Z^z \rho_{\mathsf{hist}} Z^z X^x$.
- Send the first message (commitment), $\mathsf{msg}_1 := (X^x Z^z \rho_{\mathsf{hist}} Z^z X^x) \otimes \mathsf{com}(x) \otimes \mathsf{com}(z)$, to the verifier, where $\mathsf{com}(x) := \bigotimes_{i=1}^n \mathsf{com}_i(x_i)$ and $\mathsf{com}(z) := \bigotimes_{i=1}^n \mathsf{com}_i(z_i)$.

**The second action by the verifier (challenge phase):**

- Generate $c \leftarrow [m]$.
- Compute $\mathsf{cert}_i(x_i) \leftarrow \mathsf{Del}(\mathsf{com}_i(x_i))$ and $\mathsf{cert}_i(z_i) \leftarrow \mathsf{Del}(\mathsf{com}_i(z_i))$ for all $i \in \overline{S}_c$.
- Send the second message (challenge), $\mathsf{msg}_2 := (c, \{\mathsf{cert}_i(x_i), \mathsf{cert}_i(z_i)\}_{i\in\overline{S}_c})$, to the prover.

**The third action by the prover (reply phase):**

- Send the third message (reply), $\mathsf{msg}_3 := \{\mathsf{d}_i(x_i), \mathsf{d}_i(z_i)\}_{i\in S_c}$, to the verifier.
- Output $\top$ if $\top \leftarrow \mathsf{Cert}(\mathsf{cert}_i(x_i), \mathsf{ck}_i(x_i))$ and $\top \leftarrow \mathsf{Cert}(\mathsf{cert}_i(z_i), \mathsf{ck}_i(z_i))$ for all $i \in \overline{S}_c$, and output $\bot$ otherwise.

**The fourth action by the verifier (verification phase):**

- Compute $x'_i \leftarrow \mathsf{Verify}(\mathsf{com}_i(x_i), \mathsf{d}_i(x_i))$ and $z'_i \leftarrow \mathsf{Verify}(\mathsf{com}_i(z_i), \mathsf{d}_i(z_i))$ for all $i \in S_c$. If $x'_i = \bot$ or $z'_i = \bot$ for at least one $i \in S_c$, output $\bot$ and abort.
- Apply $X_i^{x'_i} Z_i^{z'_i}$ on the $i$-th qubit of $X^x Z^z \rho_{\mathsf{hist}} Z^z X^x$ for each $i \in S_c$, and perform the POVM measurement $\{\Pi_c, I - \Pi_c\}$ on the state.
- Output $\top$ if the result $\Pi_c$ is obtained, and output $\bot$ otherwise.

**Theorem 4.5.** *$\Sigma_{\Xi\mathsf{cd}}$ is a certified everlasting zero-knowledge proof for **QMA** with $(1 - \mathsf{negl}(\lambda))$-completeness and $\left(1 - \frac{1}{\mathrm{poly}(\lambda)}\right)$-soundness.*

This is shown from the following Lemmata 4.6 to 4.9.

**Lemma 4.6.** *$\Sigma_{\Xi\mathsf{cd}}$ satisfies the $(1 - \mathsf{negl}(\lambda))$-completeness and prover's completeness.*

**Lemma 4.7.** *If $\Sigma_{\mathsf{ccd}}$ is classical-extractor-based binding, then $\Sigma_{\Xi\mathsf{cd}}$ satisfies $\left(1 - \frac{1}{\mathrm{poly}(\lambda)}\right)$-soundness.*

**Lemma 4.8.** *If $\Sigma_{\mathsf{ccd}}$ is certified everlasting hiding and computational hiding, then $\Sigma_{\Xi\mathsf{cd}}$ satisfies certified everlasting zero-knowledge.*

**Lemma 4.9.** *If $\Sigma_{\mathsf{ccd}}$ is computational hiding, then $\Sigma_{\Xi\mathsf{cd}}$ satisfies computational zero-knowledge.*

*Proof of Lemma 4.6.* It is clear from the definition of $k$-**SimQMA** (Definition 2.4) and the correctness of $\Sigma_{\mathsf{ccd}}$. □

*Proof of Lemma 4.7.* Let us show the soundness by analyzing the case for $\mathsf{x} \in A_{\mathsf{no}}$. The prover sends the first message to the verifier. The first message consists of three registers, $RS$, $RCX$, and $RCZ$. The register $RCX$ further consists of $n$ registers $\{RCX_i\}_{i\in[n]}$. The register $RCZ$ also consists of $n$ registers $\{RCZ_i\}_{i\in[n]}$. If the prover is honest, $RS$ contains $X^x Z^z \rho_{\mathsf{hist}} Z^z X^x$, $RCX_i$ contains $\mathsf{com}_i(x_i)$, and $RCZ_i$ contains $\mathsf{com}_i(z_i)$. Let $\mathsf{com}'_{i,x}$ and $\mathsf{com}'_{i,z}$ be the (reduced) states of the registers $RCX_i$ and $RCZ_i$, respectively. Let $f'_{i,x}$ and $f'_{i,z}$ be classical parts of $\mathsf{com}'_{i,x}$ and $\mathsf{com}'_{i,z}$, respectively.

The verifier generates $c \leftarrow [m]$, and issues the deletion certificate. The verifier sends $c$ and the deletion certificate to the prover. The verifier then receives $\{\mathsf{d}_1^{x,i}, \mathsf{d}_2^{x,i}, \mathsf{d}_1^{z,i}, \mathsf{d}_2^{z,i}\}_{i\in S_c}$ from the prover. For each $i \in [n]$, let us define $\mathsf{d}_1^{*,x,i}$ and $\mathsf{d}_1^{*,z,i}$ by $\mathsf{d}_1^{*,x,i} \leftarrow \mathsf{Ext}(f'_{i,x})$ and $\mathsf{d}_1^{*,z,i} \leftarrow \mathsf{Ext}(f'_{i,z})$, respectively. Note that each $\mathsf{d}_1^{*,x,i}$ and $\mathsf{d}_1^{*,z,i}$ is independent of $c$, because $\mathsf{com}'_{i,x}$ and $\mathsf{com}'_{i,z}$ are sent to the verifier before the verifier chooses $c$.

We have only to consider the case when $\mathsf{d}_1^{x,i} = \mathsf{d}_1^{*,x,i}$ and $\mathsf{d}_1^{z,i} = \mathsf{d}_1^{*,z,i}$ for all $i \in S_c$, because of the following reason: Due to the classical-extractor-based binding of $\Sigma_{\mathsf{ccd}}$, $\mathsf{Verify}(\mathsf{com}'_{i,x}, (\mathsf{d}_1^{x,i}, \mathsf{d}_2^{x,i})) = \bot$ for any $\mathsf{d}_1^{x,i} \neq \mathsf{d}_1^{*,x,i}$ and any $\mathsf{d}_2^{x,i}$. Similarly, $\mathsf{Verify}(\mathsf{com}'_{i,z}, (\mathsf{d}_1^{z,i}, \mathsf{d}_2^{z,i})) = \bot$ for any $\mathsf{d}_1^{z,i} \neq \mathsf{d}_1^{*,z,i}$ and any $\mathsf{d}_2^{z,i}$. Therefore, the prover who wants to make the verifier accept has to send $\mathsf{d}_1^{x,i} = \mathsf{d}_1^{*,x,i}$ and $\mathsf{d}_1^{z,i} = \mathsf{d}_1^{*,z,i}$ for all $i \in S_c$.

Let us define

$$p(x,z) \coloneqq \Pr\left[\bigwedge_{i\in[n]} \left(\mathsf{Verify}_2(\mathsf{com}'_{i,x}, \mathsf{d}_1^{*,x,i}) \to x_i \wedge \mathsf{Verify}_2(\mathsf{com}'_{i,z}, \mathsf{d}_1^{*,z,i}) \to z_i\right)\right].$$

Note that $p(x,z)$ is independent of $c$, because $\{\mathsf{com}'_{i,x}, \mathsf{com}'_{i,z}\}_{i\in[n]}$ and $\{\mathsf{d}_1^{*,x,i}, \mathsf{d}_1^{*,z,i}\}_{i\in[n]}$ are independent of $c$. Let $\psi$ be the (reduced) state of the register $RS$. The verifier's acceptance probability is

$$\begin{aligned}
&\frac{1}{m}\sum_{c\in[m]}\sum_{x,z\in\{0,1\}^n} p(x,z)\,\mathrm{Tr}\left[\Pi_c\left(\prod_{i\in S_c} Z_i^{z_i}X_i^{x_i}\right)\psi\left(\prod_{i\in S_c} X_i^{x_i}Z_i^{z_i}\right)\right]\\
&= \frac{1}{m}\sum_{c\in[m]}\sum_{x,z\in\{0,1\}^n} p(x,z)\,\mathrm{Tr}\left[\Pi_c\left(\prod_{i\in[n]} Z_i^{z_i}X_i^{x_i}\right)\psi\left(\prod_{i\in[n]} X_i^{x_i}Z_i^{z_i}\right)\right]\\
&= \frac{1}{m}\sum_{c\in[m]}\mathrm{Tr}\left[\Pi_c\sum_{x,z\in\{0,1\}^n} p(x,z)\left(\prod_{i\in[n]} Z_i^{z_i}X_i^{x_i}\right)\psi\left(\prod_{i\in[n]} X_i^{x_i}Z_i^{z_i}\right)\right]\\
&\le 1-\frac{1}{\mathrm{poly}(\lambda)},
\end{aligned}$$

where the last inequality comes from Definition 2.4. This completes the proof. □

*Proof of Lemma 4.8.* Let us show certified everlasting zero-knowledge. For a subset $S_c \subseteq [n]$ and $x,z\in\{0,1\}^n$, let us define $x^{S_c} \coloneqq (x_1^{S_c}, x_2^{S_c}, \cdots, x_n^{S_c})$ and $z^{S_c} \coloneqq (z_1^{S_c}, z_2^{S_c}, \cdots, z_n^{S_c})$, where $x_i^{S_c} = x_i$ and $z_i^{S_c} = z_i$ for $i\in S_c$, and $x_i^{S_c} = z_i^{S_c} = 0$ for $i\notin S_c$.

For clarity, we describe how the interactive algorithm $\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\xi)\rangle(\mathsf{x})$ runs against a QPT verifier $\mathcal{V}^*$ with an input $\xi$, where $\mathsf{w}$ is the witness and $\mathsf{x}$ is the instance.

$\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\xi)\rangle(\mathsf{x})$**:**

1. $\mathcal{P}$ generates $x,z\leftarrow\{0,1\}^n$, and computes
$$(\mathsf{com}_i(x_i), \mathsf{d}_i(x_i), \mathsf{ck}_i(x_i)) \leftarrow \mathsf{Commit}(1^\lambda, x_i)$$
$$(\mathsf{com}_i(z_i), \mathsf{d}_i(z_i), \mathsf{ck}_i(z_i)) \leftarrow \mathsf{Commit}(1^\lambda, z_i)$$
for all $i\in[n]$. $\mathcal{P}$ sends $\mathsf{msg}_1 \coloneqq (X^xZ^z\rho_{\mathsf{hist}}Z^zX^x)\otimes \mathsf{com}(x)\otimes\mathsf{com}(z)$ to $\mathcal{V}^*$.
2. $\mathcal{V}^*$ appends $\xi$ to the received state, and runs a QPT circuit $V_1^*$ on it to obtain $(c, \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in\overline{S}_c})$. $\mathcal{V}^*$ sends $\mathsf{msg}_2 \coloneqq (c, \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in\overline{S}_c})$ to $\mathcal{P}$.
3. $\mathcal{P}$ sends $\mathsf{msg}_3 \coloneqq \{\mathsf{d}_i(x_i), \mathsf{d}_i(z_i)\}_{i\in S_c}$ to $\mathcal{V}^*$.
4. $\mathcal{V}^*$ appends $\mathsf{msg}_3$ to its state, and runs a QPT circuit $V_2^*$ on it. $\mathcal{V}^*$ outputs its state $\xi'$.
5. $\mathcal{P}$ computes $\mathsf{Cert}(\mathsf{cert}'_{i,x}, \mathsf{ck}_i(x_i))$ and $\mathsf{Cert}(\mathsf{cert}'_{i,z}, \mathsf{ck}_i(z_i))$ for all $i\in\overline{S}_c$. If all outputs are $\top$, then $\mathcal{P}$ outputs $\top$. Otherwise, $\mathcal{P}$ outputs $\bot$.

Next let us define a simulator $\mathcal{S}^{(1)}$ as follows.

**The simulator** $\mathcal{S}^{(1)}(\mathsf{x}, \mathcal{V}^*, \xi)$**:**

1. Pick $c\leftarrow[m]$ and $x,z\leftarrow\{0,1\}^n$. Compute
$$(\mathsf{com}_i(x_i^{S_c}), \mathsf{d}_i(x_i^{S_c}), \mathsf{ck}_i(x_i^{S_c})) \leftarrow \mathsf{Commit}(1^\lambda, x_i^{S_c})$$
$$(\mathsf{com}_i(z_i^{S_c}), \mathsf{d}_i(z_i^{S_c}), \mathsf{ck}_i(z_i^{S_c})) \leftarrow \mathsf{Commit}(1^\lambda, z_i^{S_c})$$
for all $i\in[n]$.

2. Generate $(X^x Z^z \sigma(c) Z^z X^x) \otimes \mathsf{com}(x^{S_c}) \otimes \mathsf{com}(z^{S_c}) \otimes \xi$, where $\sigma(c) := \rho_{\mathsf{sim}}^{\mathsf{x},S_c} \otimes \left(\prod_{i\in\overline{S}_c} |0\rangle\langle 0|_i\right)$. Run $V_1^*$ on the state to obtain $(c', \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in\overline{S}_{c'}})$.
3. If $c' \neq c$, abort and output a fixed state $\eta$ and the flag state fail.
4. Append $\{\mathsf{d}_i(x_i^{S_c}), \mathsf{d}_i(z_i^{S_c})\}_{i\in S_c}$ to its quantum state, and run $V_2^*$ on the state to obtain $\xi'$.
5. Compute $\mathsf{Cert}(\mathsf{cert}'_{i,x}, \mathsf{ck}_i(x_i^{S_c}))$ and $\mathsf{Cert}(\mathsf{cert}'_{i,z}, \mathsf{ck}_i(z_i^{S_c}))$ for all $i \in \overline{S}_c$. If all outputs are $\top$, then output the state $(\top, \xi')$. Otherwise, output $(\bot, \bot)$. Also output the flag state success.

Let us also define other two simulators, $\mathcal{S}^{(2)}$ and $\mathcal{S}^{(3)}$, as follows.

**The simulator $\mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \xi)$:** It is the same as $\mathcal{S}^{(1)}$ except that $\sigma(c)$ is replaced with $\rho_{\mathsf{hist}}$.

**The simulator $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \xi)$:** $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,)$ is the channel that postselects the output of $\mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,)$ on the non-aborting state. More precisely, if we write $\mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \rho_{in}) = p\rho_{out} \otimes \mathsf{success} + (1-p)\eta \otimes \mathsf{fail}$, where $p$ is the non-aborting probability, $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \rho_{in}) = \rho_{out}$.

Lemma 4.8 is shown from the following Propositions 4.10 to 4.12 (whose proofs will be given later) and quantum rewinding lemma (Lemma 2.1), which is used to reduce the probability that $\mathcal{S}^{(1)}$ aborts to $\mathsf{negl}(\lambda)$. In fact, from Proposition 4.10 and Lemma 2.1, there exists a quantum circuit $\mathcal{S}^{(0)}$ of size at most $O(m\,\mathrm{poly}(n)\mathrm{size}(\mathcal{S}^{(1)}))$ such that the probability that $\mathcal{S}^{(0)}$ aborts is $\mathsf{negl}(\lambda)$, and the output quantum states of $\mathcal{S}^{(0)}$ and $\mathcal{S}^{(1)}$ are $\mathsf{negl}(\lambda)$-close when they do not abort. From Propositions 4.11 and 4.12, $\mathcal{S}^{(0)}$ is $\mathsf{negl}(\lambda)$-close to the real protocol, which completes the proof. □

**Proposition 4.10.** *If $\Sigma_{\mathsf{ccd}}$ is computationally hiding, then the probability that $\mathcal{S}^{(1)}$ does not abort is $\frac{1}{m} \pm \mathsf{negl}(\lambda)$.*

**Proposition 4.11.** $\mathcal{S}^{(1)}(\mathsf{x}, \mathcal{V}^*, \cdot\,) \approx_s \mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,)$ *for any* $\mathsf{x} \in A_{\mathsf{yes}} \cap \{0,1\}^\lambda$ *and any* $\mathsf{w} \in R_A(\mathsf{x})$.

**Proposition 4.12.** *If $\Sigma_{\mathsf{ccd}}$ is certified everlasting hiding,* $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,) \approx_s \mathsf{OUT}'_{\mathcal{P},\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$.

*Proof of Proposition 4.10.* This can be shown similarly to [BG20, Lemma 5.6]. For the convenience of readers, we provide a proof in Appendix A. □

*Proof of Proposition 4.11.* It is clear from the local simulatability (Definition 2.4) and the definition of $x^{S_c}$ and $z^{S_c}$ (all $x_i^{S_c}$ and $z_i^{S_c}$ are 0 except for those in $i \in S_c$). □

*Proof of Proposition 4.12.* We prove the proposition by contradiction. We construct an adversary $\mathcal{B}$ that breaks the security of the certified everlasting hiding of $\Sigma_{\mathsf{ccd}}$ by assuming the existence of a distinguisher $\mathcal{D}$ that distinguishes two states $\delta_0$ and $\delta_1$,

$$\begin{aligned}\delta_0 &:= (\mathsf{OUT}'_{\mathcal{P},\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x}) \otimes I)\sigma \\ \delta_1 &:= (\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,) \otimes I)\sigma,\end{aligned}$$

with a certain state $\sigma$. Let us describe how $\mathcal{B}$ works.

1. $\mathcal{B}$ generates $c \leftarrow [m]$ and $x, z \leftarrow \{0,1\}^n$.
2. $\mathcal{B}$ sends $m_0 := \{x_i, z_i\}_{i\in\overline{S}_c}$ and $m_1 := 0^{2n-10}$ to the challenger of $\mathsf{Exp}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{B}}(\lambda, b)$. $\mathcal{B}$ receives commitments from the challenger which is either $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$ or $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$.
3. $\mathcal{B}$ computes

$$\begin{aligned}(\mathsf{com}_i(x_i), \mathsf{d}_i(x_i), \mathsf{ck}_i(x_i)) &\leftarrow \mathsf{Commit}(1^\lambda, x_i) \\ (\mathsf{com}_i(z_i), \mathsf{d}_i(z_i), \mathsf{ck}_i(z_i)) &\leftarrow \mathsf{Commit}(1^\lambda, z_i)\end{aligned}$$

for $i \in S_c$ by itself.

4. $\mathcal{B}$ generates $X^xZ^z\rho_{\mathsf{hist}}Z^zX^x$. $\mathcal{B}$ appends commitments and $\sigma$ to the quantum state. If the commitments for $i \in \overline{S}_c$ are $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$, $\mathcal{B}$ obtains $(X^xZ^z\rho_{\mathsf{hist}}Z^zX^x) \otimes \mathsf{com}(x) \otimes \mathsf{com}(z) \otimes \sigma$. If the commitments for $i \in \overline{S}_c$ are $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$, $\mathcal{B}$ obtains $(X^xZ^z\rho_{\mathsf{hist}}Z^zX^x) \otimes \mathsf{com}(x^{S_c}) \otimes \mathsf{com}(z^{S_c}) \otimes \sigma$.
5. $\mathcal{B}$ runs $V_1^*$ on it to obtain $(c', \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in\overline{S}_{c'}})$. $\mathcal{B}$ aborts when $c \neq c'$.
6. $\mathcal{B}$ appends $\{\mathsf{d}_i(x_i), \mathsf{d}_i(z_i)\}_{i\in S_c}$ to the post-measurement state and runs $V_2^*$ on it to obtain $\sigma'$.
7. $\mathcal{B}$ sends $\{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in\overline{S}_c}$ to the challenger of $\mathsf{Exp}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{B}}(\lambda, b)$, and receives $\bot$ or $\{\mathsf{d}_i(x_i), \mathsf{d}_i(z_i)\}_{i\in\overline{S}_c}$ and $\{\mathsf{ck}_i(x_i), \mathsf{ck}_i(z_i)\}_{i\in\overline{S}_c}$ from the challenger.
8. $\mathcal{B}$ passes $(\bot, \bot)$ to $\mathcal{D}$ if $\mathcal{B}$ receives $\bot$ from the challenger, and passes $(\top, \sigma')$ to $\mathcal{D}$ otherwise.
9. When $\mathcal{D}$ outputs $b$, $\mathcal{B}$ outputs $b$.

When $\mathcal{B}$ receives $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$ from the challenger and it does not abort, it simulates $\mathsf{OUT}'_{\mathcal{P},\mathcal{V}^*}\langle\mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$. Because $(X^xZ^z\rho_{\mathsf{hist}}Z^zX^x) \otimes \mathsf{com}(x) \otimes \mathsf{com}(z) \otimes \sigma$ is independent of $c$, the probability that $\mathcal{B}$ does not abort is $\frac{1}{m}$. Therefore, $\mathcal{B}$ can simulate $\mathsf{OUT}'_{\mathcal{P},\mathcal{V}^*}\langle\mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$ with probability $\frac{1}{m}$.

When $\mathcal{B}$ receives $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$ from the challenger and it does not abort, it simulates $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot)$. The probability that $\mathcal{B}$ does not abort is $\frac{1}{m} \pm \mathsf{negl}(\lambda)$ from Propositions 4.10 and 4.11. Therefore, $\mathcal{B}$ can simulate $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot)$ with probability $\frac{1}{m} \pm \mathsf{negl}(\lambda)$.

Therefore, if there exists a distinguisher $\mathcal{D}$ that distinguishes $\delta_0$ and $\delta_1$, $\mathcal{B}$ can distinguish $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$ from $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$. From Lemma 3.7, this contradicts the certified everlasting hiding of $\Sigma_{\mathsf{ccd}}$. □

*Proof of Lemma 4.9.* Computational zero-knowledge can be proven similarly to [BG20, Lemma 5.3] because our protocol is identical to theirs if we ignore the deletion certificates, which are irrelevant to the computational zero-knowledge property. For the convenience of readers, we provide a proof in Appendix B. □

## 4.3 Sequential Repetition for Certified Everlasting Zero-Knowledge Proof for QMA

In this section, we amplify the completeness-soundness gap of the three-round protocol constructed in the previous section by sequential repetition.

**Theorem 4.13.** *Let $\Sigma_{\Xi\mathsf{cd}}$ be a certified everlasting zero-knowledge proof for a **QMA** promise problem $A$ with $(1-\mathsf{negl}(\lambda))$-completeness and $\left(1-\frac{1}{\mathrm{poly}(\lambda)}\right)$-soundness. For any polynomial $N = \mathrm{poly}(\lambda)$, let $\Sigma^N_{\Xi\mathsf{cd}}$ be the $N$-sequential repetition of $\Sigma_{\Xi\mathsf{cd}}$. That is, $\mathcal{P}$ and $\mathcal{V}$ in $\Sigma^N_{\Xi\mathsf{cd}}$ run $\Sigma_{\Xi\mathsf{cd}}$ sequentially $N$ times. Let $\mathcal{P}_j$ and $\mathcal{V}_j$ be the prover and the verifier in the $j$-th run of $\Sigma_{\Xi\mathsf{cd}}$, respectively. $\mathcal{P}$ in $\Sigma^N_{\Xi\mathsf{cd}}$ outputs $\top$ if $\mathcal{P}_j$ outputs $\top$ for all $j \in [N]$, and outputs $\bot$ otherwise. $\mathcal{V}$ in $\Sigma^N_{\Xi\mathsf{cd}}$ outputs $\top$ if $\mathcal{V}_j$ outputs $\top$ for all $j \in [N]$, and outputs $\bot$ otherwise. $\Sigma^N_{\Xi\mathsf{cd}}$ is a certified everlasting zero-knowledge proof for $A$ with $(1-\mathsf{negl}(\lambda))$-completeness and $\mathsf{negl}(\lambda)$-soundness.*

*Proof of Theorem 4.13.* It is easy to show that $\Sigma^N_{\Xi\mathsf{cd}}$ satisfies $(1-\mathsf{negl}(\lambda))$-completeness and $\mathsf{negl}(\lambda)$-soundness. Moreover, as proven in [GO94], the sequential repetition of a computational zero-knowledge proof preserves the computational zero-knowledge property. Let us show that $\Sigma^N_{\Xi\mathsf{cd}}$ satisfies certified everlasting zero-knowledge. For clarity, we describe how $\langle\mathcal{P}(\mathsf{w}^{\otimes Nk(|\mathsf{x}|)}), \mathcal{V}^*(\xi_1)\rangle(\mathsf{x})$ runs against any QPT verifier $\mathcal{V}^*$ with an input $\xi_1$, where $\mathsf{w}$ is a witness and $\mathsf{x}$ is the instance.

$\langle\mathcal{P}(\mathsf{w}^{\otimes Nk(|\mathsf{x}|)}), \mathcal{V}^*(\xi_1)\rangle(\mathsf{x})$**:**

1. For $1 \leq j \leq N$, $\mathcal{V}^*$ and $\mathcal{P}$ run $\langle\mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*_j(\xi_j)\rangle(\mathsf{x})$ sequentially to get the outputs

$$\xi_{j+1} \coloneqq \mathsf{OUT}_{\mathcal{V}^*_j}\langle\mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*_j(\xi_j)\rangle(\mathsf{x})$$

and

$$\mathsf{OUT}_{\mathcal{P}_j}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x}) = \top/\bot,$$

respectively.

2. $\mathcal{V}^*$ outputs $\xi_{N+1}$.
3. $\mathcal{P}$ outputs $\top$ if $\mathsf{OUT}_{\mathcal{P}_j}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle = \top$ for all $j \in [N]$, and outputs $\bot$ otherwise.

Since $\Sigma_{\exists\mathsf{cd}}$ satisfies the certified everlasting zero-knowledge property, for each $j \in [N]$ and any $\mathcal{V}_j^*$ there exists a QPT algorithm (a simulator) $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \cdot\ )$ such that the following holds for any $\mathsf{x}$ and $\mathsf{w}$.

$$\mathsf{OUT}'_{\mathcal{P}_j, \mathcal{V}_j^*}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\cdot)\rangle(\mathsf{x}) \approx_s \mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \cdot\ ).$$

We show that for any $\mathcal{V}^*$ there exists a QPT algorithm (a simulator) $\mathcal{S}(\mathsf{x}, \mathcal{V}^*, \cdot\ )$ such that the following holds for any $\mathsf{x}$ and $\mathsf{w}$.

$$\mathsf{OUT}'_{\mathcal{P}, \mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes Nk(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x}) \approx_s \mathcal{S}(\mathsf{x}, \mathcal{V}^*, \cdot\ ).$$

Let us define the simulator $\mathcal{S}$ as follows.

**The simulator $\mathcal{S}(\mathsf{x}, \mathcal{V}^*, \xi_1)$:**

1. For $1 \leq j \leq N$, $\mathcal{S}$ runs $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \cdot\ )$ on $\xi_j$ to get $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)/(\top, \xi_{j+1})$ sequentially. If $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)$, then $\xi_{j+1} \coloneqq \bot$ for each $j \in [N]$.
2. $\mathcal{S}$ outputs $(\bot, \bot)$ if $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)$ for some $j \in [N]$, and outputs $(\top, \xi_{N+1})$ otherwise.

We define the sequence of hybrids $\mathsf{Hyb}_i(\xi_1)$ as follows.

**$\mathsf{Hyb}_i(\xi_1)$:**

1. For $1 \leq j \leq i$, $\mathcal{V}^*$ and $\mathcal{P}$ run $\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x})$ sequentially to get the outputs

$$\xi_{j+1} \coloneqq \mathsf{OUT}_{\mathcal{V}_j^*}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x})$$

and

$$\mathsf{OUT}_{\mathcal{P}_j}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x}) = \top/\bot,$$

respectively.

2. For $i+1 \leq j \leq N$, $\mathcal{S}$ runs $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \cdot\ )$ on $\xi_j$ to get $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)/(\top, \xi_{i+1})$ sequentially. If $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)$, then $\xi_{j+1} \coloneqq \bot$ for each $j \in [N]$.
3. The output of $\mathsf{Hyb}_i(\xi_1)$ is $(\bot, \bot)$ if $\mathsf{OUT}_{\mathcal{P}_j}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x}) = \bot$ for some $j \in [i]$ or $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)$ for some $j \in \{i+1, \cdots N\}$. Otherwise, the output of $\mathsf{Hyb}_i(\xi_1)$ is $(\top, \xi_{N+1})$.

$\mathsf{Hyb}_0(\cdot)$ and $\mathsf{Hyb}_N(\cdot)$ correspond to $\mathcal{S}(\mathsf{x}, \mathcal{V}^*, \cdot\ )$ and $\mathsf{OUT}'_{\mathcal{P}, \mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes Nk(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$, respectively. Therefore, it suffices to prove that no distinguisher can distinguish $\mathsf{Hyb}_i(\cdot)$ from $\mathsf{Hyb}_{i+1}(\cdot)$ for any $i \in [N-1]$. We assume that there exists a distinguisher $\mathcal{D}'$ that distinguishes $(\mathsf{Hyb}_i(\cdot) \otimes I)\,\sigma$ from $\left(\mathsf{Hyb}_{i+1}(\cdot) \otimes I\right)\sigma$ for a certain state $\sigma$, and construct a distinguisher $\mathcal{D}$ that breaks the certified everlasting zero-knowledge property of $\Sigma_{\exists\mathsf{cd}}$. $\mathcal{D}$ can access to the channel $\mathsf{O}(\cdot)$, which is either $\mathcal{S}_{i+1}(\mathsf{x}, \mathcal{V}_{i+1}^*, \cdot\ )$ or $\mathsf{OUT}'_{\mathcal{P}_{i+1}, \mathcal{V}_{i+1}^*}\langle \mathcal{P}_{i+1}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_{i+1}^*(\cdot)\rangle(\mathsf{x})$, and guesses whether $\mathsf{O}(\cdot)$ is $\mathcal{S}_{i+1}(\mathsf{x}, \mathcal{V}_{i+1}^*, \cdot\ )$ or $\mathsf{OUT}'_{\mathcal{P}_{i+1}, \mathcal{V}_{i+1}^*}\langle \mathcal{P}_{i+1}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_{i+1}^*(\cdot)\rangle(\mathsf{x})$. Let us define $\mathcal{D}$ as follows.

**The distinguisher $\mathcal{D}(\xi_1)$:**

1. For $1 \leq j \leq i$, $\mathcal{D}$ runs $\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x})$ sequentially to get $\xi_{j+1} \coloneqq \mathsf{OUT}_{\mathcal{V}_j^*}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x})$ and $\mathsf{OUT}_{\mathcal{P}_j}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x}) = \top/\bot$.

2. $\mathcal{D}$ runs $\mathsf{O}(\xi_{i+1})$ to get $\mathsf{O}(\xi_{i+1}) = (\bot, \bot)/(\top, \xi_{i+2})$. If $\mathsf{O}(\xi_{i+1}) = (\bot, \bot)$, $\mathcal{D}$ sets $\xi_{i+2} \coloneqq \bot$.
3. For $i+2 \leq j \leq N$, $\mathcal{D}$ runs $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \cdot\ )$ on $\xi_j$ to get $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)/(\top, \xi_{j+1})$ sequentially. If $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)$, $\mathcal{D}$ sets $\xi_{j+1} \coloneqq \bot$.
4. $\mathcal{D}$ outputs $(\bot, \bot)$ if $\mathsf{OUT}_{\mathcal{P}_j}\langle \mathcal{P}_j(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_j^*(\xi_j)\rangle(\mathsf{x}) = \bot$ for some $j \in [i]$, $\mathsf{O}(\xi_{i+1}) = (\bot, \bot)$ or $\mathcal{S}_j(\mathsf{x}, \mathcal{V}_j^*, \xi_j) = (\bot, \bot)$ for some $j \in \{i+2, \cdots, N\}$, and outputs $(\top, \xi_{N+1})$ otherwise.
5. $\mathcal{D}$ sends the output of $\mathcal{D}$ to $\mathcal{D}'$.
6. If $\mathcal{D}'$ outputs $b$, $\mathcal{D}$ outputs $b$.

We can see that $\mathcal{D}$ generates $(\mathsf{Hyb}_i(\cdot) \otimes I)\,\sigma$ when $\mathsf{O}(\cdot)$ is $\mathcal{S}_{i+1}(\mathsf{x}, \mathcal{V}_{i+1}^*, \cdot\ )$ and $\mathcal{D}$ takes $\sigma$ as input. Similarly, we can see that $\mathcal{D}$ generates $\left(\mathsf{Hyb}_{i+1}(\cdot) \otimes I\right)\sigma$ when $\mathsf{O}(\cdot)$ is $\mathsf{OUT}'_{\mathcal{P}_{i+1}, \mathcal{V}_{i+1}^*}\langle \mathcal{P}_{i+1}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_{i+1}^*(\cdot)\rangle(\mathsf{x})$ and $\mathcal{D}$ takes $\sigma$ as input. Therefore, if $\mathcal{D}'$ distinguishes $(\mathsf{Hyb}_i(\cdot) \otimes I)\,\sigma$ from $\left(\mathsf{Hyb}_{i+1}(\cdot) \otimes I\right)\sigma$, then $\mathcal{D}$ can distinguish $\mathcal{S}_{i+1}(\mathsf{x}, \mathcal{V}_{i+1}^*, \cdot\ )$ from $\mathsf{OUT}'_{\mathcal{P}_{i+1}, \mathcal{V}_{i+1}^*}\langle \mathcal{P}_{i+1}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}_{i+1}^*(\cdot)\rangle(\mathsf{x})$. This contradicts the certified everlasting zero-knowledge property of $\Sigma_{\Xi\mathsf{cd}}$, which completes the proof. □

## Acknowledgement

TM is supported by the JST Moonshot R&D JPMJMS2061-5-1-1, JST FOREST, MEXT Q-LEAP, and the Grant-in-Aid for Scientific Research (B) No.JP19H04066 of JSPS.

## References

[ACGH20] Gorjan Alagic, Andrew M. Childs, Alex B. Grilo, and Shih-Han Hung. Non-interactive classical verification of quantum computation. In Rafael Pass and Krzysztof Pietrzak, editors, *TCC 2020, Part III*, volume 12552 of *LNCS*, pages 153–180. Springer, Heidelberg, November 2020. (Cited on page 5.)

[AHU19] Andris Ambainis, Mike Hamburg, and Dominique Unruh. Quantum security proofs using semi-classical oracles. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 269–295. Springer, Heidelberg, August 2019. (Cited on page 3, 6.)

[BB21] Nir Bitansky and Zvika Brakerski. Classical binding for quantum commitments. *IACR Cryptol. ePrint Arch.*, 2021:1001, 2021. (Cited on page 4, 5.)

[BCKM21] James Bartusek, Andrea Coladangelo, Dakshita Khurana, and Fermi Ma. On the round complexity of secure quantum computation. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 406–435, Virtual Event, August 2021. Springer, Heidelberg. (Cited on page 5.)

[BDF+11] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Heidelberg, December 2011. (Cited on page 2.)

[BG20] Anne Broadbent and Alex B. Grilo. QMA-hardness of consistency of local density matrices with applications to quantum zero-knowledge. In *61st FOCS*, pages 196–205. IEEE Computer Society Press, November 2020. (Cited on page 2, 4, 5, 6, 20, 21.)

[BI20] Anne Broadbent and Rabib Islam. Quantum encryption with certified deletion. In Rafael Pass and Krzysztof Pietrzak, editors, *TCC 2020, Part III*, volume 12552 of *LNCS*, pages 92–122. Springer, Heidelberg, November 2020. (Cited on page 1, 2, 3, 4, 7, 8.)

[BJSW16] Anne Broadbent, Zhengfeng Ji, Fang Song, and John Watrous. Zero-knowledge proof systems for QMA. In Irit Dinur, editor, *57th FOCS*, pages 31–40. IEEE Computer Society Press, October 2016. (Cited on page 5.)

[BM21] James Bartusek and Giulio Malavolta. Candidate obfuscation of null quantum circuits and witness encryption for QMA. *IACR Cryptology ePrint Archive*, 2021:421, 2021. (Cited on page 5.)

[BS20] Nir Bitansky and Omri Shmueli. Post-quantum zero knowledge in constant rounds. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *52nd ACM STOC*, pages 269–279. ACM Press, June 2020. (Cited on page 5.)

[BY20] Zvika Brakerski and Henry Yuen. Quantum garbled circuits. *arXiv:2006.01085*, 2020. (Cited on page 5.)

[CCKV08] André Chailloux, Dragos Florin Ciocan, Iordanis Kerenidis, and Salil P. Vadhan. Interactive and noninteractive zero knowledge are equivalent in the help model. In Ran Canetti, editor, *TCC 2008*, volume 4948 of *LNCS*, pages 501–534. Springer, Heidelberg, March 2008. (Cited on page 5.)

[CDMS04] Claude Crépeau, Paul Dumais, Dominic Mayers, and Louis Salvail. Computational collapse of quantum state with application to oblivious transfer. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 374–393. Springer, Heidelberg, February 2004. (Cited on page 4.)

[CM21] Orestis Chardouvelis and Giulio Malavolta. The round complexity of quantum zero-knowledge. *IACR Cryptol. ePrint Arch.*, 2021. (Cited on page 5.)

[CVZ20] Andrea Coladangelo, Thomas Vidick, and Tina Zhang. Non-interactive zero-knowledge arguments for QMA, with preprocessing. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020, Part III*, volume 12172 of *LNCS*, pages 799–828. Springer, Heidelberg, August 2020. (Cited on page 5.)

[DFR+07] Ivan Damgård, Serge Fehr, Renato Renner, Louis Salvail, and Christian Schaffner. A tight high-order entropic quantum uncertainty relation with applications. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 360–378. Springer, Heidelberg, August 2007. (Cited on page 4.)

[DFS04] Ivan Damgård, Serge Fehr, and Louis Salvail. Zero-knowledge proofs and string commitments withstanding quantum attacks. In Matthew Franklin, editor, *CRYPTO 2004*, volume 3152 of *LNCS*, pages 254–272. Springer, Heidelberg, August 2004. (Cited on page 4.)

[For87] Lance Fortnow. The complexity of perfect zero-knowledge (extended abstract). In Alfred Aho, editor, *19th ACM STOC*, pages 204–209. ACM Press, May 1987. (Cited on page 1, 2.)

[FUW+20] Junbin Fang, Dominique Unruh, Jian Weng, Jun Yan, and Dehua Zhou. How to base security on the perfect/statistical binding property of quantum bit commitment? *IACR Cryptol. ePrint Arch.*, 2020:621, 2020. (Cited on page 5.)

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

[GO94] Oded Goldreich and Yair Oren. Definitions and properties of zero-knowledge proof systems. *J. Cryptology*, 7:1–32, 1994. (Cited on page 21.)

[GSV98] Oded Goldreich, Amit Sahai, and Salil P. Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *30th ACM STOC*, pages 399–408. ACM Press, May 1998. (Cited on page 2.)

[GSY19] Alex Bredariol Grilo, William Slofstra, and Henry Yuen. Perfect zero knowledge for quantum multiprover interactive proofs. In David Zuckerman, editor, *60th FOCS*, pages 611–635. IEEE Computer Society Press, November 2019. (Cited on page 5.)

[HMNY21] Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Quantum encryption with certified deletion, revisited: Public key, attribute-based, and classical communication. *IACR Cryptol. ePrint Arch.*, 2021:617, 2021. (Cited on page 2, 3, 12.)

[Kob03] Hirotada Kobayashi. Non-interactive quantum perfect and statistical zero-knowledge. In Toshihide Ibaraki, Naoki Katoh, and Hirotaka Ono, editors, *Algorithms and Computation, 14th International Symposium, ISAAC 2003, Kyoto, Japan, December 15-17, 2003, Proceedings*, volume 2906 of *Lecture Notes in Computer Science*, pages 178–188. Springer, 2003. (Cited on page 5.)

[LC97] Hoi-Kwong Lo and H. F. Chau. Is quantum bit commitment really possible? *Phys. Rev. Lett.*, 78:3410–3413, 1997. (Cited on page 2, 5.)

[LS19] Alex Lombardi and Luke Schaeffer. A note on key agreement and non-interactive commitments. Cryptology ePrint Archive, Report 2019/279, 2019. https://eprint.iacr.org/2019/279. (Cited on page 7.)

[May97] Dominic Mayers. Unconditionally secure quantum bit commitment is impossible. *Phys. Rev. Lett.*, 78:3414–3417, 1997. (Cited on page 2, 5.)

[MW18] Sanketh Menda and John Watrous. Oracle separations for quantum statistical zero-knowledge. *arXiv:1801.08967*, 2018. (Cited on page 1, 5.)

[MY21] Tomoyuki Morimae and Takashi Yamakawa. Classically verifiable (dual-mode) NIZK for QMA with preprocessing. *arXiv:2102.09149*, 2021. (Cited on page 5.)

[Shm21] Omri Shmueli. Multi-theorem designated-verifier NIZK for QMA. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 375–405, Virtual Event, August 2021. Springer, Heidelberg. (Cited on page 5.)

[Unr13] Dominique Unruh. Everlasting multi-party computation. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 380–397. Springer, Heidelberg, August 2013. (Cited on page 1.)

[Unr15] Dominique Unruh. Revocable quantum timed-release encryption. *J. ACM*, 62(6):49:1–49:76, 2015. (Cited on page 3.)

[Wat02] John Watrous. Limits on the power of quantum statistical zero-knowledge. In *43rd FOCS*, pages 459–470. IEEE Computer Society Press, November 2002. (Cited on page 2.)

[Wat09] John Watrous. Zero-knowledge against quantum attacks. *SIAM J. Comput.*, 39(1):25–58, 2009. (Cited on page 1, 6.)

[Yan20] Jun Yan. Quantum computationally predicate-binding commitment with application in quantum zero-knowledge argument for NP. *IACR Cryptol. ePrint Arch.*, 2020:1510, 2020. (Cited on page 4, 5.)

[YWLQ15] Jun Yan, Jian Weng, Dongdai Lin, and Yujuan Quan. Quantum bit commitment with application in quantum zero-knowledge proof (extended abstract). In Khaled M. Elbassioni and Kazuhisa Makino, editors, *Algorithms and Computation - 26th International Symposium, ISAAC 2015, Nagoya, Japan, December 9-11, 2015, Proceedings*, volume 9472 of *Lecture Notes in Computer Science*, pages 555–565. Springer, 2015. (Cited on page 5.)

[Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 239–268. Springer, Heidelberg, August 2019. (Cited on page 13.)

# A Proof of Proposition 4.10

*Proof of Proposition 4.10.* We prove the proposition by contradiction. Let $p$ be the probability that $\mathcal{S}^{(1)}$ does not abort. Assume that the probability $p$ satisfies $|p - \frac{1}{m}| \geq \frac{1}{q(\lambda)}$ for a polynomial $q$. Then, we can construct an adversary $\mathcal{B}$ that breaks the computational hiding of $\Sigma_{\mathsf{ccd}}$. Let us describe how $\mathcal{B}$ works below.

1. $\mathcal{B}$ generates $c \leftarrow [m]$ and $x, z \leftarrow \{0,1\}^n$.
2. $\mathcal{B}$ sends $m_0 := \{x_i, z_i\}_{i\in S_c}$ and $m_1 := 0^{10}$ to the challenger. $\mathcal{B}$ receives commitments from the challenger which is either $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in S_c}$ or $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in S_c}$.
3. $\mathcal{B}$ generates $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in \overline{S}_c}$.
4. $\mathcal{B}$ generates $X^x Z^z \sigma(c) Z^z X^x$. $\mathcal{B}$ appends commitments and $\xi$ to the quantum state in the ascending order. If the commitments for $i \in S_c$ are $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in S_c}$, $\mathcal{B}$ obtains $X^x Z^z \sigma(c) Z^z X^x \otimes \mathsf{com}(x^{S_c}) \otimes \mathsf{com}(z^{S_c}) \otimes \xi$. If the commitments for $i \in S_c$ are $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in S_c}$, $\mathcal{B}$ obtains $X^x Z^z \sigma(c) Z^z X^x \otimes \mathsf{com}(0^n) \otimes \mathsf{com}(0^n) \otimes \xi$.
5. $\mathcal{B}$ runs $V_1^*$ on it to obtain $(c', \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in \overline{S}_{c'}})$. $\mathcal{B}$ outputs 0 when $c \neq c'$. $\mathcal{B}$ outputs 1 when $c = c'$.

When $\mathcal{B}$ receives $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in S_c}$ from the challenger, it outputs 1 with probability $p$ since it simulates $\mathcal{S}^{(1)}$. When $\mathcal{B}$ receives $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in S_c}$ from the challenger, on the other hand, it outputs 1 with probability $\frac{1}{m}$, because $(X^x Z^z \sigma(c) Z^z X^x) \otimes \mathsf{com}(0^n) \otimes \mathsf{com}(0^n) \otimes \xi$ is independent of $c$. (Note that $\sigma(c)$ is one-time padded by $x, z$.) Therefore if there exists some polynomial $q$ such that $|p - \frac{1}{m}| \geq \frac{1}{q(\lambda)}$, $\mathcal{B}$ can break the computational hiding of $\Sigma_{\mathsf{ccd}}$ from (the computational hiding version of) Lemma 3.7. □

# B Proof of Lemma 4.9

*Proof of Lemma 4.9.* This proof is similar to the proof of Lemma 4.8. For a subset $S_c \subseteq [n]$ and $x, z \in \{0,1\}^n$, let us define $x^{S_c} := (x_1^{S_c}, x_2^{S_c}, \cdots, x_n^{S_c})$ and $z^{S_c} := (z_1^{S_c}, z_2^{S_c}, \cdots, z_n^{S_c})$, where $x_i^{S_c} = x_i$ and $z_i^{S_c} = z_i$ for $i \in S_c$, and $x_i^{S_c} = z_i^{S_c} = 0$ for $i \notin S_c$.

For clarity, we describe how the interactive algorithm $\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\xi)\rangle(\mathsf{x})$ runs against a QPT verifier $\mathcal{V}^*$ with an input $\xi$, where $\mathsf{w}$ is the witness and $\mathsf{x}$ is the instance.

**$\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\xi)\rangle(\mathsf{x})$:**

1. $\mathcal{P}$ generates $x, z \leftarrow \{0,1\}^n$, and computes
$$(\mathsf{com}_i(x_i), \mathsf{d}_i(x_i), \mathsf{ck}_i(x_i)) \leftarrow \mathsf{Commit}(1^\lambda, x_i)$$
$$(\mathsf{com}_i(z_i), \mathsf{d}_i(z_i), \mathsf{ck}_i(z_i)) \leftarrow \mathsf{Commit}(1^\lambda, z_i)$$
for all $i \in [n]$. $\mathcal{P}$ sends $\mathsf{msg}_1 := (X^x Z^z \rho_{\mathsf{hist}} Z^z X^x) \otimes \mathsf{com}(x) \otimes \mathsf{com}(z)$ to $\mathcal{V}^*$.
2. $\mathcal{V}^*$ appends $\xi$ to the received state, and runs a QPT circuit $V_1^*$ on it to obtain $(c, \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in \overline{S}_c})$. $\mathcal{V}^*$ sends $\mathsf{msg}_2 := (c, \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in \overline{S}_c})$ to $\mathcal{P}$.
3. $\mathcal{P}$ sends $\mathsf{msg}_3 := \{\mathsf{d}_i(x_i), \mathsf{d}_i(z_i)\}_{i\in S_c}$ to $\mathcal{V}^*$.
4. $\mathcal{V}^*$ appends $\mathsf{msg}_3$ to its state, and runs a QPT circuit $V_2^*$ on it. $\mathcal{V}^*$ outputs its state $\xi'$.

Next let us define a simulator $\mathcal{S}^{(1)}$ as follows.

**The simulator $\mathcal{S}^{(1)}(\mathsf{x}, \mathcal{V}^*, \xi)$:**

1. Pick $c \leftarrow [m]$ and $x, z \leftarrow \{0,1\}^n$. Compute
$$(\mathsf{com}_i(x_i^{S_c}), \mathsf{d}_i(x_i^{S_c}), \mathsf{ck}_i(x_i^{S_c})) \leftarrow \mathsf{Commit}(1^\lambda, x_i^{S_c})$$
$$(\mathsf{com}_i(z_i^{S_c}), \mathsf{d}_i(z_i^{S_c}), \mathsf{ck}_i(z_i^{S_c})) \leftarrow \mathsf{Commit}(1^\lambda, z_i^{S_c})$$
for all $i \in [n]$.
2. Generate $(X^x Z^z \sigma(c) Z^z X^x) \otimes \mathsf{com}(x^{S_c}) \otimes \mathsf{com}(z^{S_c}) \otimes \xi$, where $\sigma(c) := \rho_{\mathsf{sim}}^{\mathsf{x}, S_c} \otimes \left(\prod_{i\in \overline{S}_c} |0\rangle\langle 0|_i\right)$. Run $V_1^*$ on the state to obtain $(c', \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in \overline{S}_{c'}})$.

3. If $c' \neq c$, abort and output a fixed state $\eta$ and the flag state fail.
4. Append $\{\mathsf{d}_i(x_i^{S_c}), \mathsf{d}_i(z_i^{S_c})\}_{i\in S_c}$ to its quantum state, and run $V_2^*$ on the state. $\mathcal{S}$ outputs the output state and the flag state success.

Let us also define other two simulators $\mathcal{S}^{(2)}$ and $\mathcal{S}^{(3)}$ as follows.

**The modified simulator $\mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \xi)$:** It is the same as $\mathcal{S}^{(1)}$ except that $\sigma(c)$ is replaced with $\rho_{\mathsf{hist}}$.

**The simulator $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \xi)$:** $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,)$ is the channel that postselects the output of $\mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,)$ on the non-aborting state. More precisely, if we write $\mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \rho_{in}) = p\rho_{out} \otimes \mathsf{success} + (1-p)\eta \otimes \mathsf{fail}$, where $p$ is the non-aborting probability, $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \rho_{in}) = \rho_{out}$.

Lemma 4.9 is shown from the following Propositions B.1 to B.3 (whose proofs will be given later) and quantum rewinding lemma(Lemma 2.1), which is used to reduce the probability that $\mathcal{S}^{(1)}$ aborts to $\mathsf{negl}(\lambda)$. In fact, from Proposition B.1 and Lemma 2.1, there exists a quantum circuit $\mathcal{S}^{(0)}$ of size at most $O(m\,\mathrm{poly}(n)\mathrm{size}(\mathcal{S}^{(1)}))$ such that the probability $\mathcal{S}^{(0)}$ aborts is $\mathsf{negl}(\lambda)$, and the output quantum states of $\mathcal{S}^{(0)}$ and $\mathcal{S}^{(1)}$ are $\mathsf{negl}(\lambda)$-close when they do not abort. From Propositions B.2 and B.3, $\mathcal{S}^{(0)}$ is $\mathsf{negl}(\lambda)$-close to the run of the real protocol, which completes the proof. □

**Proposition B.1.** *If $\Sigma_{\mathsf{ccd}}$ is computational hiding, then the probability that $\mathcal{S}^{(1)}$ does not abort is $\frac{1}{m} \pm \mathsf{negl}(\lambda)$.*

**Proposition B.2.** *$\mathcal{S}^{(1)}(\mathsf{x}, \mathcal{V}^*, \cdot\,) \approx_s \mathcal{S}^{(2)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,)$ for any $\mathsf{x} \in A_{\mathsf{yes}} \cap \{0,1\}^\lambda$ and any $\mathsf{w} \in R_A(\mathsf{x})$.*

**Proposition B.3.** *If $\Sigma_{\mathsf{ccd}}$ is computational hiding, $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,) \approx_c \mathsf{OUT}_{\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$.*

*Proof of Proposition B.1.* This proof is the same as the proof of Proposition 4.10. □

*Proof of Proposition B.2.* This proof is the same as the proof of Proposition 4.11. □

*Proof of Proposition B.3.* We prove the proposition by contradiction. We construct an adversary $\mathcal{B}$ that breaks the security of the computationally hiding of $\Sigma_{\mathsf{ccd}}$ by assuming the existence of a distinguisher $\mathcal{D}$ that distinguishes two states $\delta_0$ and $\delta_1$,

$$\begin{aligned}\delta_0 &:= (\mathsf{OUT}_{\mathcal{V}^*}\langle \mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x}) \otimes I)\sigma\\ \delta_1 &:= (\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot\,) \otimes I)\sigma,\end{aligned}$$

with a certain state $\sigma$. Let us describe how $\mathcal{B}$ works.

1. $\mathcal{B}$ generates $c \leftarrow [m]$ and $x, z \leftarrow \{0,1\}^n$.
2. $\mathcal{B}$ sends $m_0 := \{x_i, z_i\}_{i\in\overline{S}_c}$ and $m_1 := 0^{2n-10}$ to the challenger. $\mathcal{B}$ receives commitments from the challenger which is either $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$ or $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$.
3. $\mathcal{B}$ computes

$$\begin{aligned}(\mathsf{com}_i(x_i), \mathsf{d}_i(x_i), \mathsf{ck}_i(x_i)) &\leftarrow \mathsf{Commit}(1^\lambda, x_i)\\ (\mathsf{com}_i(z_i), \mathsf{d}_i(z_i), \mathsf{ck}_i(z_i)) &\leftarrow \mathsf{Commit}(1^\lambda, z_i)\end{aligned}$$

   for $i \in S_c$ by itself.
4. $\mathcal{B}$ generates $X^x Z^z \rho_{\mathsf{hist}} Z^z X^x$. $\mathcal{B}$ appends commitments and $\sigma$ to the quantum state. If the commitments for $i \in \overline{S}_c$ are $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$, $\mathcal{B}$ obtains $X^x Z^z \rho_{\mathsf{hist}} Z^z X^x \otimes \mathsf{com}(x) \otimes \mathsf{com}(z) \otimes \sigma$. If the commitments for $i \in \overline{S}_c$ are $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$, $\mathcal{B}$ obtains $X^x Z^z \rho_{\mathsf{hist}} Z^z X^x \otimes \mathsf{com}(x^{S_c}) \otimes \mathsf{com}(z^{S_c}) \otimes \sigma$.
5. $\mathcal{B}$ runs $V_1^*$ on it to obtain $(c', \{\mathsf{cert}'_{i,x}, \mathsf{cert}'_{i,z}\}_{i\in\overline{S}_{c'}})$. $\mathcal{B}$ aborts when $c \neq c'$.

6. $\mathcal{B}$ appends $\{\mathsf{d}_i(x_i), \mathsf{d}_i(z_i)\}_{i\in S_c}$ to the post-measurement state and runs $V_2^*$ on it.
7. $\mathcal{B}$ passes the output state to $\mathcal{D}$.
8. When $\mathcal{D}$ outputs $b$, $\mathcal{B}$ outputs $b$.

When $\mathcal{B}$ receives $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$ from the challenger and it does not abort, it simulates $\mathsf{OUT}_{\mathcal{V}^*}\langle\mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$. Because $(X^xZ^z\rho_{\mathsf{hist}}Z^zX^x)\otimes\mathsf{com}(x)\otimes\mathsf{com}(z)\otimes\sigma$ is independent of $c$, the probability that $\mathcal{B}$ does not abort is $\frac{1}{m}$. Therefore, $\mathcal{B}$ can simulate $\mathsf{OUT}_{\mathcal{V}^*}\langle\mathcal{P}(\mathsf{w}^{\otimes k(|\mathsf{x}|)}), \mathcal{V}^*(\cdot)\rangle(\mathsf{x})$ with probability $\frac{1}{m}$.

When $\mathcal{B}$ receives $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$ from the challenger and it does not abort, it simulates $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot)$. The probability that $\mathcal{B}$ does not abort is $\frac{1}{m} \pm \mathsf{negl}(\lambda)$ from Propositions B.1 and B.2. Therefore, $\mathcal{B}$ can simulate $\mathcal{S}^{(3)}(\mathsf{x}, \mathsf{w}^{\otimes k(|\mathsf{x}|)}, \mathcal{V}^*, \cdot)$ with probability $\frac{1}{m} \pm \mathsf{negl}(\lambda)$.

Therefore, if there exists the distinguisher $\mathcal{D}$ that distinguishes $\delta_0$ and $\delta_1$, $\mathcal{B}$ can distinguish $\{\mathsf{com}_i(x_i), \mathsf{com}_i(z_i)\}_{i\in\overline{S}_c}$ from $\{\mathsf{com}_i(0), \mathsf{com}_i(0)\}_{i\in\overline{S}_c}$. From (the computational hiding version of) Lemma 3.7, this contradicts the computational hiding of $\Sigma_{\mathsf{ccd}}$.

□

# C Commitment with Certified Everlasting Hiding and Sum-Binding

In this appendix, we define and construct commitment with certified everlasting hiding and statistical sum-binding.

## C.1 Definition

**Definition C.1 (Commitment with Certified Everlasting Hiding and Sum-Binding (Syntax)).** *Let $\lambda$ be the security parameter, and let $p$, $q$, $r$ and $s$ be some polynomials. Commitment with certified everlasting hiding and sum-binding consists of a tuple of algorithms* $(\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$ *with message space $\mathcal{M} := \{0,1\}$, commitment space $\mathcal{C} := \mathcal{Q}^{\otimes p(\lambda)}$, decommitment space $\mathcal{D} := \{0,1\}^{q(\lambda)}$, key space $\mathcal{K} := \{0,1\}^{r(\lambda)}$ and deletion certificate space $\mathcal{E} := \{0,1\}^{s(\lambda)}$.*

$\mathsf{Commit}(1^\lambda, b) \to (\mathsf{com}, \mathsf{d}, \mathsf{ck})$**:** *The commitment algorithm takes as input a security parameter $1^\lambda$ and a message $b \in \{0,1\}$, and outputs a commitment* $\mathsf{com} \in \mathcal{C}$*, a decommitment* $\mathsf{d} \in \mathcal{D}$*, and a key* $\mathsf{ck} \in \mathcal{K}$*.*

$\mathsf{Verify}(\mathsf{com}, \mathsf{d}, b) \to \top$ *or* $\bot$**:** *The verification algorithm takes as input* com, d *and $b$, and outputs $\top$ or $\bot$.*

$\mathsf{Del}(\mathsf{com}) \to \mathsf{cert}$**:** *The deletion algorithm takes* com *as input, and outputs a certificate* $\mathsf{cert} \in \mathcal{E}$*.*

$\mathsf{Cert}(\mathsf{cert}, \mathsf{ck}) \to \top$ *or* $\bot$**:** *The certification algorithm takes* cert *and* ck *as input, and outputs $\top$ or $\bot$.*

**Definition C.2 (Correctness).** *There are two types of correctness, namely, decommitment correctness and deletion correctness.*

**Decommitment correctness:** *There exists a negligible function* negl *such that for any $\lambda \in \mathbb{N}$ and $b \in \{0,1\}$,*

$$\Pr\left[\mathsf{Verify}(\mathsf{com}, \mathsf{d}, b) = \top \mid (\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, b)\right] \geq 1 - \mathsf{negl}(\lambda).$$

**Deletion correctness:** *There exists a negligible function* negl *such that for any $\lambda \in \mathbb{N}$ and $b \in \{0,1\}$,*

$$\Pr\left[\mathsf{Cert}(\mathsf{cert}, \mathsf{ck}) = \top \mid (\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, b), \mathsf{cert} \leftarrow \mathsf{Del}(\mathsf{com})\right] \geq 1 - \mathsf{negl}(\lambda).$$

**Definition C.3 ($\epsilon$-Sum-Binding).** *For any* com, d, *and* d′*, it holds that*

$$\Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d}, 0) = \top] + \Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d}', 1) = \top] \leq 1 + \epsilon.$$

*We call $\epsilon$-sum-binding just sum-binding if $\epsilon$ is negligible.*

**Definition C.4 (Computational Hiding).** *Let $\Sigma := (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$. Let us consider the following security experiment $\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, b)$ against any QPT adversary $\mathcal{A}$.*

1. *The challenger computes $(\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, b)$, and sends $\mathsf{com}$ to $\mathcal{A}$.*
2. *$\mathcal{A}$ outputs $b' \in \{0,1\}$.*
3. *The output of the experiment is $b'$.*

*Computational hiding means that the following is satisfied for any QPT $\mathcal{A}$.*

$$\mathsf{Adv}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 0) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{c\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 1) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

**Definition C.5 (Certified Everlasting Hiding).** *Let $\Sigma := (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$. Let us consider the following security experiment $\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, b)$ against $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ consisting of any QPT adversary $\mathcal{A}_1$ and any unbounded adversary $\mathcal{A}_2$.*

1. *The challenger computes $(\mathsf{com}, \mathsf{d}, \mathsf{ck}) \leftarrow \mathsf{Commit}(1^\lambda, b)$, and sends $\mathsf{com}$ to $\mathcal{A}_1$.*
2. *At some point, $\mathcal{A}_1$ sends $\mathsf{cert}$ to the challenger, and sends its internal state to $\mathcal{A}_2$.*
3. *The challenger computes $\mathsf{Cert}(\mathsf{cert}, \mathsf{ck})$. If the output is $\top$, then the challenger outputs $\top$, and sends $(\mathsf{d}, \mathsf{ck})$ to $\mathcal{A}_2$. Else, the challenger outputs $\bot$, and sends $\bot$ to $\mathcal{A}_2$.*
4. *$\mathcal{A}_2$ outputs $b' \in \{0,1\}$.*
5. *If the challenger outputs $\top$, then the output of the experiment is $b'$. Otherwise, the output of the experiment is $\bot$.*

*We say that it is certified everlasting hiding if the following is satisfied for any $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$.*

$$\mathsf{Adv}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 0) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 1) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

## C.2 Construction

Though the construction is essentially the same as that in Section 3.2, we give the full description for clarity. Let $\lambda$ be the security parameter, and let $p$, $q$, $r$, $s$, $t$ and $u$ be some polynomials. We construct a bit commitment with certified everlasting hiding and sum-binding, $\Sigma_{\mathsf{ccd}} = (\mathsf{Commit}, \mathsf{Verify}, \mathsf{Del}, \mathsf{Cert})$, with message space $\mathcal{M} = \{0,1\}$, commitment space $\mathcal{C} = \mathcal{Q}^{\otimes p(\lambda)} \times \{0,1\}^{q(\lambda)} \times \{0,1\}^{r(\lambda)}$, decommitment space $\mathcal{D} = \{0,1\}^{s(\lambda)} \times \{0,1\}^{t(\lambda)}$, key space $\mathcal{K} = \{0,1\}^{r(\lambda)}$ and deletion certificate space $\mathcal{E} = \{0,1\}^{u(\lambda)}$ from the following primitives:

- Secret-key encryption with certified deletion $\Sigma_{\mathsf{skcd}} = \mathsf{SKE}.(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Del}, \mathsf{Verify})$, with plaintext space $\mathcal{M} = \{0,1\}$, ciphertext space $\mathcal{C} = \mathcal{Q}^{\otimes p(\lambda)}$, key space $\mathcal{K} = \{0,1\}^{r(\lambda)}$, and deletion certificate space $\mathcal{E} = \{0,1\}^{u(\lambda)}$.
- Classical non-interactive commitment, $\Sigma_{\mathsf{com}} = \mathsf{Classical.Commit}$, with plaintext space $\{0,1\}^{s(\lambda)}$, randomness space $\{0,1\}^{t(\lambda)}$, and commitment space $\{0,1\}^{q(\lambda)}$.
- A hash function $H$ from $\{0,1\}^{s(\lambda)}$ to $\{0,1\}^{r(\lambda)}$ modeled as a quantumly-accessible random oracle.

The construction is as follows.

$\mathsf{Commit}(1^\lambda, b)$**:**

- Generate $\mathsf{ske.sk} \leftarrow \mathsf{SKE.KeyGen}(1^\lambda)$, $R \leftarrow \{0,1\}^{s(\lambda)}$, $R' \leftarrow \{0,1\}^{t(\lambda)}$, and a hash function $H$ from $\{0,1\}^{s(\lambda)}$ to $\{0,1\}^{r(\lambda)}$.
- Compute $\mathsf{ske.CT} \leftarrow \mathsf{SKE.Enc}(\mathsf{ske.sk}, b)$, $f \leftarrow \mathsf{Classical.Commit}(R; R')$, and $h := H(R) \oplus \mathsf{ske.sk}$.

- Output $\mathsf{com} := (\mathsf{ske.CT}, f, h)$, $\mathsf{d} := (R, R')$, and $\mathsf{ck} := \mathsf{ske.sk}$.

$\mathsf{Verify}(\mathsf{com}, \mathsf{d}, b)$**:**

- Parse $\mathsf{com} = (\mathsf{ske.CT}, f, h)$ and $\mathsf{d} = (R, R')$.
- Compute $\mathsf{ske.sk}' := H(R) \oplus h$.
- Compute $b' \leftarrow \mathsf{SKE.Dec}(\mathsf{ske.sk}', \mathsf{ske.CT})$.
- Output $\top$ if $f = \mathsf{Classical.Commit}(R; R')$ and $b' = b$, and output $\bot$ otherwise.

$\mathsf{Del}(\mathsf{com})$**:**

- Parse $\mathsf{com} = (\mathsf{ske.CT}, f, h)$.
- Compute $\mathsf{ske.cert} \leftarrow \mathsf{SKE.Del}(\mathsf{ske.CT})$.
- Output $\mathsf{cert} := \mathsf{ske.cert}$.

$\mathsf{Cert}(\mathsf{cert}, \mathsf{ck})$**:**

- Parse $\mathsf{cert} = \mathsf{ske.cert}$ and $\mathsf{ck} = \mathsf{ske.sk}$.
- Output $\top/\bot \leftarrow \mathsf{SKE.Verify}(\mathsf{ske.sk}, \mathsf{ske.cert})$.

**Correctness.** The decommitment and deletion correctness easily follow from the correctness of $\Sigma_{\mathsf{skcd}}$.

**Security.** We prove the following three theorems.

**Theorem C.6.** *If $\Sigma_{\mathsf{com}}$ is perfect binding, then $\Sigma_{\mathsf{ccd}}$ is sum-binding.*

**Theorem C.7.** *If $\Sigma_{\mathsf{com}}$ is unpredictable and $\Sigma_{\mathsf{skcd}}$ is OT-CD secure, then $\Sigma_{\mathsf{ccd}}$ is certified everlasting hiding.*

**Theorem C.8.** *If $\Sigma_{\mathsf{com}}$ is unpredictable and $\Sigma_{\mathsf{skcd}}$ is OT-CD secure, then $\Sigma_{\mathsf{ccd}}$ is computationally hiding.*

*Proof of Theorem C.6.* What we have to prove is that for any $\mathsf{com}$, $\mathsf{d}$, and $\mathsf{d}'$, it holds that

$$\Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d}, 0) = \top] + \Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d}', 1) = \top] \leq 1 + \mathsf{negl}(\lambda).$$

Let $\mathsf{d} = (R_0, R_0')$, $\mathsf{d}' = (R_1, R_1')$, and $\mathsf{com} = (\mathsf{ske.CT}, f, h)$. Then,

$$\begin{aligned}
&\Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d}, 0) = \top] + \Pr[\mathsf{Verify}(\mathsf{com}, \mathsf{d}', 1) = \top] \\
&= \Pr[0 \leftarrow \mathsf{SKE.Dec}(h \oplus H(R_0), \mathsf{ske.CT}) \wedge f = \mathsf{Classical.Commit}(R_0; R_0')] \\
&+ \Pr[1 \leftarrow \mathsf{SKE.Dec}(h \oplus H(R_1), \mathsf{ske.CT}) \wedge f = \mathsf{Classical.Commit}(R_1; R_1')] \\
&\leq \Pr\left[0 \leftarrow \mathsf{SKE.Dec}(h \oplus H(\tilde{R}), \mathsf{ske.CT}) \wedge f = \mathsf{Classical.Commit}(\tilde{R}; R_0')\right] \\
&+ \Pr\left[1 \leftarrow \mathsf{SKE.Dec}(h \oplus H(\tilde{R}), \mathsf{ske.CT}) \wedge f = \mathsf{Classical.Commit}(\tilde{R}; R_1')\right] \\
&\leq \Pr\left[0 \leftarrow \mathsf{SKE.Dec}(h \oplus H(\tilde{R}), \mathsf{ske.CT})\right] + \Pr\left[1 \leftarrow \mathsf{SKE.Dec}(h \oplus H(\tilde{R}), \mathsf{ske.CT})\right] \\
&= \Pr\left[0 \leftarrow \mathsf{SKE.Dec}(h \oplus H(\tilde{R}), \mathsf{ske.CT}) \vee 1 \leftarrow \mathsf{SKE.Dec}(h \oplus H(\tilde{R}), \mathsf{ske.CT})\right] \\
&\leq 1,
\end{aligned}$$

where we have used perfect binding of $\Sigma_{\mathsf{com}}$ in the second inequality. □

*Proof of Theorem C.7.* It is the same as that of Theorem 3.9. □

*Proof of Theorem C.8.* It is the same as that of Theorem 3.10. □

# D Proof of Lemma 3.7

Let us consider the following hybrids for $j \in \{0, 1, ..., n\}$.

$\mathsf{Hyb}_j$:

1. $\mathcal{A}_1$ generates $(m^0, m^1) \in \{0,1\}^n \times \{0,1\}^n$ and sends it to the challenger.
2. The challenger computes
$$(\mathsf{com}_i(m_i^1), \mathsf{d}_i(m_i^1), \mathsf{ck}_i(m_i^1)) \leftarrow \mathsf{Commit}(1^\lambda, m_i^1)$$
for $i \in [j]$ and
$$(\mathsf{com}_i(m_i^0), \mathsf{d}_i(m_i^0), \mathsf{ck}_i(m_i^0)) \leftarrow \mathsf{Commit}(1^\lambda, m_i^0)$$
for each $i \in \{j+1, ..., n\}$, and sends $\{\mathsf{com}_i(m_i^1)\}_{i\in[j]}$ and $\{\mathsf{com}_i(m_i^0)\}_{i\in\{j+1,...,n\}}$ to $\mathcal{A}_1$. Here, $m_i^b$ is the $i$-th bit of $m^b$.
3. At some point, $\mathcal{A}_1$ sends $\{\mathsf{cert}_i\}_{i\in[n]}$ to the challenger, and sends its internal state to $\mathcal{A}_2$.
4. The challenger computes $\mathsf{Cert}(\mathsf{cert}_i, \mathsf{ck}_i(m_i^1))$ for each $i \in [j]$ and $\mathsf{Cert}(\mathsf{cert}_i, \mathsf{ck}_i(m_i^0))$ for each $i \in \{j+1, ..., n\}$. If the outputs are $\top$ for all $i \in [n]$, then the challenger outputs $\top$, and sends $\{\mathsf{d}_i(m_i^1), \mathsf{ck}_i(m_i^1)\}_{i\in[j]}$ and $\{\mathsf{d}_i(m_i^0), \mathsf{ck}_i(m_i^0)\}_{i\in\{j+1,...,n\}}$ to $\mathcal{A}_2$. Else, the challenger outputs $\bot$, and sends $\bot$ to $\mathcal{A}_2$.
5. $\mathcal{A}_2$ outputs $b' \in \{0,1\}$.
6. If the challenger outputs $\top$, then the output of the experiment is $b'$. Otherwise, the output of the experiment is $\bot$.

It is clear that $\mathsf{Hyb}_0 = \mathsf{Exp}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 0)$ and $\mathsf{Hyb}_n = \mathsf{Exp}^{\mathsf{bit\text{-}ever\text{-}hide}}_{\Sigma,\mathcal{A}}(\lambda, 1)$. Furthermore, we can show
$$\left|\Pr\left[\mathsf{Hyb}_j = 1\right] - \Pr\left[\mathsf{Hyb}_{j+1} = 1\right]\right| \leq \mathsf{negl}(\lambda)$$
for each $j \in \{0, 1, ..., n-1\}$. (Its proof is given below.) From these facts, we obtain Lemma 3.7.

Let us show the remaining one. To show it, let us assume that $\left|\Pr\left[\mathsf{Hyb}_j = 1\right] - \Pr\left[\mathsf{Hyb}_{j+1} = 1\right]\right|$ is non-negligible. Then, we can construct an adversary $\mathcal{B}$ that can break the certified everlasting hiding of $\Sigma_{\mathsf{ccd}}$ as follows.

1. $\mathcal{B}$ receives $(m^0, m^1)$ from $\mathcal{A}_1$, and computes
$$(\mathsf{com}_i(m_i^1), \mathsf{d}_i(m_i^1), \mathsf{ck}_i(m_i^1)) \leftarrow \mathsf{Commit}(1^\lambda, m_i^1)$$
for $i \in [j]$ and
$$(\mathsf{com}_i(m_i^0), \mathsf{d}_i(m_i^0), \mathsf{ck}_i(m_i^0)) \leftarrow \mathsf{Commit}(1^\lambda, m_i^0)$$
for $i \in \{j+2, ..., n\}$.
2. $\mathcal{B}$ sends $(m_{j+1}^0, m_{j+1}^1)$ to the challenger of $\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{B}}(\lambda, b')$, and receives $\mathsf{com}_{j+1}(m_{j+1}^{b'})$ from the challenger.
3. $\mathcal{B}$ sends $\{\mathsf{com}_i(m_i^1)\}_{i\in[j]}$, $\mathsf{com}_{j+1}(m_{j+1}^{b'})$, and $\{\mathsf{com}_i(m_i^0)\}_{i\in\{j+2,...,n\}}$, to $\mathcal{A}_1$.
4. $\mathcal{A}_1$ sends $\{\mathsf{cert}_i\}_{i\in[n]}$ to $\mathcal{B}$, and sends its internal state to $\mathcal{A}_2$.
5. $\mathcal{B}$ sends $\mathsf{cert}_{j+1}$ to the challenger of $\mathsf{Exp}^{\mathsf{ever\text{-}hide}}_{\Sigma_{\mathsf{ccd}},\mathcal{B}}(\lambda, b')$, and receives $(\mathsf{d}_{j+1}(m_{j+1}^{b'}), \mathsf{ck}_{j+1}(m_{j+1}^{b'}))$ or $\bot$ from the challenger. If $\mathcal{B}$ receives $\bot$ from the challenger, it outputs $\bot$ and aborts.
6. $\mathcal{B}$ sends all $\mathsf{d}_i$ and $\mathsf{ck}_i$ to $\mathcal{A}_2$.
7. $\mathcal{A}_2$ outputs $b''$.
8. $\mathcal{B}$ computes $\mathsf{Cert}$ for all $\mathsf{cert}_i$, and outputs $b''$ if all results are $\top$. Otherwise, $\mathcal{B}$ outputs $\bot$.

It is clear that $\Pr[\mathcal{B} \to 1 \mid b' = 0] = \Pr\left[\mathsf{Hyb}_j = 1\right]$ and $\Pr[\mathcal{B} \to 1 \mid b' = 1] = \Pr\left[\mathsf{Hyb}_{j+1} = 1\right]$. By assumption, $|\Pr\left[\mathsf{Hyb}_j = 1\right] - \Pr\left[\mathsf{Hyb}_{j+1} = 1\right]|$ is non-negligible, and therefore $|\Pr[\mathcal{B} \to 1 \mid b' = 0] - \Pr[\mathcal{B} \to 1 \mid b' = 1]|$ is non-negligible, which contradict the certified everlasting hiding of $\Sigma_{\mathsf{ccd}}$.