Zero-Knowledge Proofs of Quantumness

Duong Hieu Phan[1], Weiqiang Wen[1], Xingyu Yan[2,3], and Jinwei Zheng[1]

[1] LTCI, Telecom Paris, Institut Polytechnique de Paris, Paris, France
{hieu.phan,weiqiang.wen,jinwei.zheng}@telecom-paris.fr

[2] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876

[3] School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing, 100081
yanxy2020@bupt.edu.cn

Abstract. With the rapid development of quantum computers, proofs of quantumness have recently become an interesting and intriguing research direction. However, in all current schemes for proofs of quantumness, quantum provers almost invariably face the risk of being maliciously exploited by classical verifiers. In fact, through malicious strategies in interaction with quantum provers, classical verifiers could solve some instances of hard problems that arise from the specific scheme in use. In other words, malicious verifiers can break some schemes (that quantum provers are not aware of) through interaction with quantum provers. All this is due to the lack of formalization that prevents malicious verifiers from extracting useful information in proofs of quantumness.

To address this issue, we formalize zero-knowledge proofs of quantumness. Intuitively, the zero-knowledge property necessitates that the information gained by the classical verifier from interactions with the quantum prover should not surpass what can be simulated using a simulated classical prover interacting with the same verifier. As a result, the new zero-knowledge notion can prevent any malicious verifier from exploiting quantum advantage. Interestingly, we find that the classical zero-knowledge proof is sufficient to compile some existing proofs of quantumness schemes into zero-knowledge proofs of quantumness schemes. Due to some technical reasons, it appears to be more general to require zero-knowledge proof on the verifier side instead of the prover side. Intuitively, this helps to regulate the verifier's behavior from malicious to be honest-but-curious. As a result, both parties will play not only one role in the proofs of quantumness but also the dual role in the classical zero-knowledge proof.

Specifically, the two principle proofs of quantumness schemes: Shor's factoring-based scheme and learning with errors-based scheme in [Brakerski et al, FOCS, 2018], can be transformed into zero-knowledge proofs of quantumness by requiring an extractable non-interactive zero-knowledge argument on the verifier side. Notably, the zero-knowledge proofs of quantumness can be viewed as an enhanced security notion for proofs of quantumness. To prevent malicious verifiers from exploiting the quantum device's capabilities or knowledge, it is advisable to transition existing proofs of quantumness schemes to this framework whenever feasible.

Keywords: Quantum cryptography, Zero-knowledge, Proofs of quantumness.

1 Introduction

A cryptographic proofs of quantumness (PoQ) is a scheme that validates a quantum device's ability to perform tasks that classical devices cannot handle efficiently. These schemes enable a quantum prover to convince a classical verifier of its quantum computational power through specific challenging tasks, such as factoring [Sho94,Reg24], BosonSampling [AA11,BJS11], learning with errors (LWE)-based trapdoor claw-free (TCF) hash functions [BCM+18,BKVV20], computational Bell test [KMCVY22,KLVY23,BGKM+23], etc.

Recent PoQ schemes primarily address scenarios where the prover may act maliciously, such as checking whether a classical prover is masquerading as a quantum device and verifying whether the results produced by the quantum device are indeed correct. In addition to this, this work considers an important supplementary perspective, focusing on situations where **the classical verifier might act maliciously**. To see how a malicious verifier could behave, we provide a scenario about how the quantum prover could be maliciously exploited in the PoQ as shown in Figure 1.

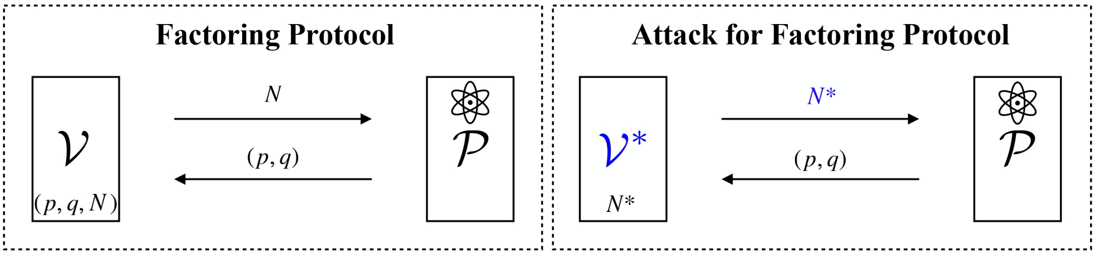

Fig. 1. Schematic diagram of a malicious verifier $\mathcal{V}^*$ spoofing to utilize the quantum capabilities of the quantum prover $\mathcal{P}$ in factoring-based quantumness proof scheme.

In the factoring-based PoQ scheme, a malicious verifier $\mathcal{V}^*$ can simply replace the large integer number N with their carefully chosen number N^* (e.g., that can be taken from an RSA public key). With such substitution, the verifier could easily exploit the prover's quantum capabilities to factor N^* for gaining benefit. Then in the LWE-based PoQ scheme in [BCM+18], how the verifier can make use of the quantum prover is much less clear. On one side, assuming the post-quantum hardness of LWE problem, the classical verifier should not be able to get much more advantage from the quantum prover for solving the LWE instance. Whereas the transcripts from the quantum prover should contain more information than that which any classical prover could possess. This actually helps the verifier to distinguish quantum prover from any macilious classical prover. To conclude, we can not properly prevent the verifier from obtaining additional information through the quantum prover beyond being merely convinced of the prover's quantumness.

Regarding the explicit attack and potential unexpected advantage that the verifier could gain from the quantum prover, preventing malicious verifiers from extracting useful information from PoQ is necessary and raises the following question:

Is there a "zero-knowledge" proofs of quantumness that allows a prover to demonstrate quantum capability without revealing any other useful information?

The "zero-knowledge" property is to ensure that, at any point, the quantum prover's capabilities are not maliciously exploited by the verifier. In other words, the interactive proof process does not reveal any information beyond the fact that "the prover possesses quantum capabilities". We believe that considering the zero-knowledge PoQ (ZKPoQ) scheme is crucial and has far-reaching implications. In the post-quantum era, the test of quantumness may serve as the first step for classical users to subscribe to quantum services. The ZKPoQ can fundamentally ensure that the quantum server's computational power is not swindled by classical users during this verification process, thereby safeguarding the server's interests.

1.1 Our Contributions

The traditional security model for the proofs of quantumness only concerns the completeness and soundness, where the latter one requires that no classical prover can successfully run the protocol and pass the verification. In this work, we introduce the zero-knowledge property for the proofs of quantumness protocol for the first time. In particular, the zero-knowledge property is also defined in a classical manner but with respect to quantum capability instead of particular knowledge. Informally, a proofs of quantumness protocol is said to be zero-knowledge when the communication between the quantum prover and the classical verifier can be perfectly simulated with a polynomial-time probabilistic classical prover communicated with the same verifier (as shown in Figure 2). Under this zero-knowledge definition, the verifier can not exploit the quantum advantage of the quantum prover via their communication anymore, as their communication does not leak more information than the communication between the same verifier and a (simulated) classical prover.

To demonstrate the rationality of this new notion, we migrate two mainstream PoQ schemes, factoring-based scheme and LWE-based scheme [BCM+18], to the ones satisfying zero-knowledge. Informally, we define the PoQ with zero knowledge property as follows.

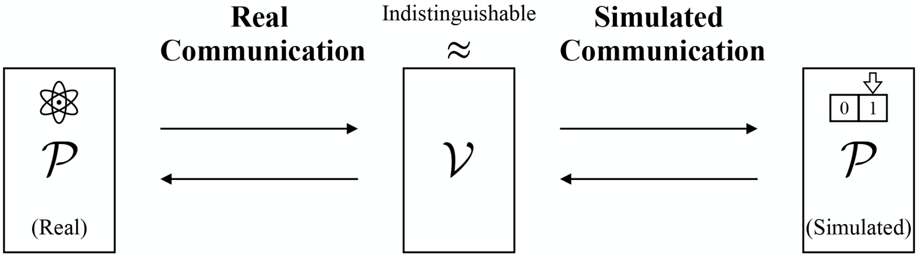

Fig. 2. Illustration of our new zero-knowledge property for proofs of quantumness.

Definition 1 (Zero-knowledge proofs of quantumness, ZKPoQ). *A ZKPoQ scheme satisfies quantum completeness, classical soundness, and (computational) zero-knowledge properties.*

Our first technical contribution (the formal statement is in Section 3) concerns the zero-knowledge property. We show that certain existing PoQ schemes can be efficiently transformed into the ones with zero-knowledge property.

Infeasibility of requiring zero-knowledge proofs from prover. Intuitively, one might think that a quantum prover could directly make use of a zero-knowledge proof to hide the information from the prover while still making the verification feasible. Indeed, this approach works well with the factoring-based scheme, where the prover could directly respond with zero-knowledge proof of factorization (p, q) for N. However, it is not suitable for other schemes such as the LWE-based PoQ scheme [BCM$^+$18]. In the latter one, the verifier will challenge the quantum prover to provide the witness to some statement that is only known by the verifier itself. In this case, it is infeasible to require the prover to provide a zero-knowledge proof without knowing the explicit statement to prove. In more details, in the equation test of the PoQ scheme in [BCM$^+$18] (refer to Figure 6 for the description of the scheme), the prover is asked to provide a witness $\sigma_1 = (c, \mathbf{d}) \in \{0,1\} \times \{0,1\}^n$ to the statement $c = \mathbf{d}^\top \cdot (\mathbf{x}_0 \oplus \mathbf{x}_1) \bmod 2$, where $\mathbf{x}_0$ and $\mathbf{x}_1$ are two preimages for a same output y under a post-quantum secure claw-free function. Due to the claw-free property, even the quantum prover should not be able to obtain both preimages $\mathbf{x}_0$ and $\mathbf{x}_1$. As a result, the adversary does not have access to the statement. Therefore, instead of requiring the prover to provide a zero-knowledge proof, our approach considers ensuring ZKPoQ by requiring the verifier to provide a zero-knowledge proof to claim that it is not behaving maliciously.

This leads to our first technical contribution, which supplements zero-knowledge on top of two existing PoQ schemes by requiring an extractable proof from the verifier's side. To sketch the idea, any classical prover communicated with the verfier, can now extract necessary secret knowledge from the additional extractable proof from the verifier, which is sufficient to be used to simulate the communication without any quantum resource. This properly ensures the zero-knowledge on the prover side for the PoQ scheme. However, the newly added extractable proof from the verifier causes an issue on the soundness of the PoQ protocol, as any classical prover can simulate the communication using the knowledge under the extractable proof without exploiting quantum advantage. To fix this issue, we further require the extractable proof to be zero-knowledge. The latter one can be constructed from one-way function and public key encryption [JK25]. As an interesting result, both zero-knowledge and soundness properties of the aforementioned ZKPoQ primitive can be satisfied simultaneously. In particular, we provide transformation from two PoQ schemes to ZKPoQ schemes (refer to Theorems 1 and 2 for the informal statements of our results).

As shown in Figure 3, the main modification on top of the original scheme is that we further require the verifier to provide an extractable-NIZK proof of the factorization (p, q) of the integer N. One might notice that both the prover and the verifier in the proofs of quantumness scheme will also play the dual role in the additional NIZK proof. In particular, to argue the soundness of this resulting ZKPoQ scheme, we can rely on the zero-knowledge property of the extractable-NIZK proof such that any malicious classical prover can not get any information from this additional proof and should factor the integer N by itself. Notably, a classically secure NIZK will be sufficient as the ZK property is only required to hold facing a malicious classical prover for the soundness of proofs of quantumness. Regarding the zero-knowledge property of this ZKPoQ scheme, the simulator

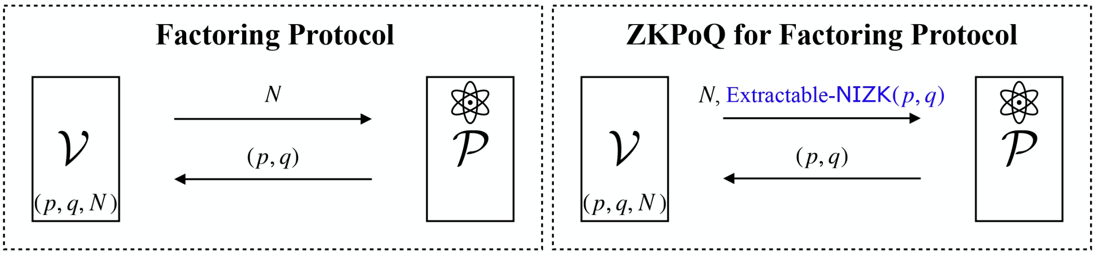

Fig. 3. Diagram of ZKPoQ for factoring-based scheme.

can then make use of the extractability property of the extractable-NIZK proof to extract the witness (p, q) from the proof. Therefore, the transcript from the quantum prover can be simulated perfectly.

Theorem 1 (ZKPoQ for factoring-based scheme). *Given an extractable-NIZK proof, the factoring-based PoQ scheme shown in Figure 3 can be transformed into ZKPoQ one.*

Now we can see how this ZKPoQ scheme can be used to avoid the attack scenarios depicted in Figure 1. As illustrated in Figure 3, according to the extractability of the extractable-NIZK proof, it would be infeasible for a malicious verifier $\mathcal{V}^*$ to provide a valid proof without knowing the factorization (p, q), which efficiently regulate the malicious behavior of the verifier in this scenario.

Next, we move the transformation from the LWE-based scheme in Figure 4 to ZKPoQ scheme. Again, the principle adjustment is to ask the verifier to submit an extractable-NIZK proof of the secret of the LWE instance. As before, the soundness of the ZKPoQ scheme is based on the zero-knowledge property of the extractable-NIZK proof. Notably, the zero-knowledge property of our LWE-based ZKPoQ scheme can also be achieved by the extratability of the extractable-NIZK proof. The main observation is that any classical prover knowing the secret of the LWE instance (sent from the verifier) can also pass the verification. Therefore, the simulator can extract the LWE secret and then perfectly simulate the transcripts from the prover to the verifier. This does not invalidate the soundness of the scheme, because the verifier is the only one knowing the secret, and therefore anyone else should follow the scheme to prove quantumness.

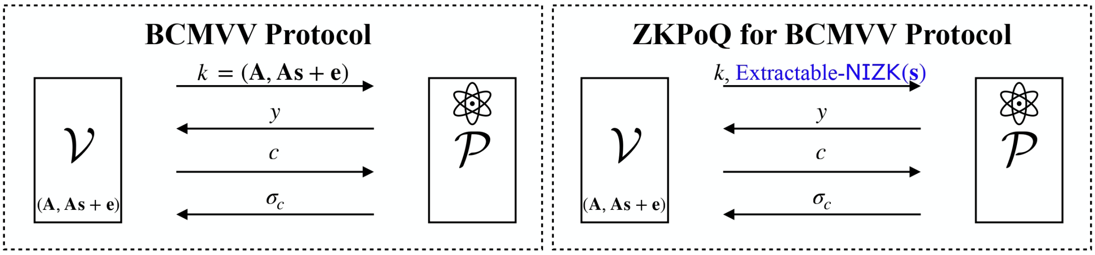
BCMVV Protocol
$\mathcal{V}$
$(\mathbf{A}, \mathbf{As}+\mathbf{e})$
$k = (\mathbf{A}, \mathbf{As}+\mathbf{e})$
y
c
σ_c
$\mathcal{P}$
ZKPoQ for BCMVV Protocol
$\mathcal{V}$
$(\mathbf{A}, \mathbf{As}+\mathbf{e})$
k, Extractable-NIZK$(\mathbf{s})$
y
c
σ_c
$\mathcal{P}$

Fig. 4. Diagram of ZKPoQ for the LWE-based scheme [BCM+18].

Theorem 2 (ZKPoQ for the LWE-based scheme [BCM+18]). *Given an extractable-NIZK proof, the LWE-based scheme in [BCM+18] can be transformed into ZKPoQ one.*

Last but not least, we emphasize that using classically secure extractable-NIZK proof in the aforementioned ZKPoQ schemes is sufficient. Moreover, if we want to apply our LWE-based ZKPoQ scheme for other purpose such as certifiable randomness from quantum device [BCM+18] or key leasing [CGJL23], post-quantum secure extractable-NIZK seems to be necessary. Because in that case, we want to ensure that even the quantum prover could not gain any extra advantage from the extractable-NIZK proof. As shown in Theorem 4, there exists a post-quantum secure extractable-NIZK proof based on the LWE assumption.

Difficulty of having a generic transformation for proofs of quantumness to be zero-knowledge. In this work, we have examinated two examples based on factoring and LWE, respectively. These two protocols shares the same challenge-response type of procedure between the prover and verifier. Under this category, there exist more LWE-based protocols, but additional assumptions are involved. For example, the PoQ scheme in [BKVV20] requires random oracle and the ones in [KMCVY22,BGKM$^+$23] further assume Bell's inequality. Therefore, it is not clear whether the idea of requiring an extractable-NIZK from the verifier's side is also applicable to the other LWE-based protocols. Other than the challenge-response style protocols, there is also non-interactive PoQ protocol such as the Bosonsampling [AA11,BJS11]. Therefore, though our idea seems to be quite general, different methods may still be required for other protocols to achieve zero-knowledge.

1.2 Related works

Proofs of quantumness (PoQ) represent a milestone for the field of quantum computation, serving as a classically-verifiable demonstration of non-classical behavior from a single quantum device. To date, there are three mainstream approaches for demonstrating proofs of quantumness:

1. **Factoring-Based PoQ**: This approach leverages the well-known computational hardness of factoring large integers, a task presumed to be intractable for classical computers but efficiently solvable by quantum computers using algorithms such as Shor's algorithm [Sho94]. The quantum demonstration involves producing factors of large composite numbers, which can be verified classically. However, implementing factoring-based PoQ requires fault-tolerant scalable quantum computer, which is unimplementable on short-term NISQ devices.
2. **Sampling-Based PoQ**: This approach is based on the classical hardness of sampling problems, such as boson sampling [AA11,MLA$^+$22] or random circuit sampling [AAB$^+$19,WBC$^+$21]. Although these methods can be implemented with NISQ hardware, they are not efficiently verifiable since the classical verification of these methods typically demands exponential time.
3. **LWE-Based PoQ**: Another PoQ approach introduced by Brkerski et al. [BCM$^+$18], utilizes the quantum hardness of the Learning With Errors (LWE) problem to execute a cryptographic interactive protocol between a polynomial-time classical verifier and an ostensibly quantum polynomial-time prover. Wherein, the verifier presents challenges to the prover and checks the correctness of the prover's responses. A crucial aspect of this method is that an effective quantum strategy should enable the prover to accurately respond to the verifier's challenges with a high likelihood of success, while any effective classical strategy would only achieve a low probability of success, assuming the hardness of LWE assumption. In terms of implementation, the LWE-based PoQ appears to require significantly fewer resources compared to the Factoring-based PoQ, yet it still demands more than the Sampling-based PoQ. Regarding verification, both LWE-based and Factoring-based PoQ can be efficiently verified classically, unlike Sampling-based PoQ, which does not allow for such efficient classical verification. Following on the breakthrough work of [BCM$^+$18], numerous studies such as [BKVV20,ACGH20,KMCVY22,KLVY23,AMMW24,BGKM$^+$23] have developed progressively more efficient proofs of quantumness. The goal of these efforts is to simplify these tests so that they can be implemented on current NISQ quantum devices.

The above PoQ protocols, along with follow-up works, have not addressed scenarios that involve malicious verifiers, an area yet to be explored in the current research. This paper introduces zero-knowledge proofs of quantumness, presenting them as an advanced security concept for proofs of quantumness. Given that sampling-based PoQ schemes do not align with the cryptographic interactive protocol framework and lack the capability for efficient classical verification, our work specifically focuses the formalization of zero-knowledge proofs on classically verifiable PoQ protocols, with particular emphasis on those based on factoring and LWE.

1.3 Open Problems

Although our results naturally transform PoQ into ZKPoQ using extractable-NIZK, there remain many intriguing open problems regarding the general transformation of PoQ into ZKPoQ. Here, we mention some of them.

- Is it possible to provide a more general transformation framework that formalizes the required characteristics of the PoQ? In our current work, we have demonstrated that both the factoring-based PoQ and LWE-based PoQ protocols can be directly upgraded to ZKPoQ using extractable-NIZK techniques. However, for sampling-based PoQ schemes, they do not align with the cryptographic interactive protocol framework and lack the capability for efficient classical verification, which makes such a general transformation challenging. This remains an open problem, and we will consider exploring this direction in future work. For other LWE-based PoQ protocols, namely [BKVV20,ACGH20,KMCVY22,KLVY23,AMMW24,BGKM$^+$23], we note that these protocols often rely on different heuristic assumptions. Therefore, it is not trivial to apply our approach to these schemes. However, these protocols closely resemble the one we considered, making it interesting to explore potential common characteristics that satisfy the transformation. This would be another future research direction.
- Can interactive solutions in the plain model replace NIZKs in both ZKPoQ constructions, thereby eliminating the need for the CRS model? In our current construction, we chose to use post-quantum NIZK to minimize interaction between parties, but this comes at the cost of relying on a CRS. In order to remove the CRS and work in the plain model, one can instead resort to the interactive zero-knowledge proofs [GMW86,BG92]. We will consider it as future work to formally explore the feasibility of this replacement.

2 Preliminaries

2.1 Notation

We use the acronyms PPT and QPT for probabilistic polynomial time and quantum polynomial time respectively. For a classical probabilistic algorithm $\mathcal{A}$, we write $\mathcal{A}(x;r)$ to denote running $\mathcal{A}$ on input x, with input randomness r. For a finite set S, we use $x \overset{\$}{\leftarrow} S$ to denote uniform sampling of x from the set S. We denote $[n] = \{1, 2, \cdots, n\}$. For clarity, unless otherwise stated, the terms "Verifier ($\mathcal{V}$)" and "Prover ($\mathcal{P}$)" refer specifically to their roles in the PoQ scheme.

Conceptually, trapdoor claw-free functions (TCF) consist of a pair of injective functions $\{f_{k,0}, f_{k,1}\}_k$ that share the same image. With access to a secret trapdoor td, it becomes easy to determine the two preimages $\mathbf{x}_0$ and $\mathbf{x}_1$ of the same image y, such that $f_0(\mathbf{x}_0) = f_1(\mathbf{x}_1) = y$. However, it is computationally difficult to invert $\{f_{k,0}, f_{k,1}\}_k$ without the trapdoor td. Such a pair of $(\mathbf{x}_0, \mathbf{x}_1)$ is known as a claw, hence the name is claw-free. In the LWE-based scheme [BCM$^+$18], they use the Noisy TCF (NTCF) informally described by $f'_{k,b}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{e}' + b \cdot (\mathbf{A}\mathbf{s} + \mathbf{e})$ for $b \in \{0,1\}$ and $k = (\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e})$. In the following, we recall the definition of NTCF as well as its realization under LWE.

Definition 2 (NTCF family, [BCM$^+$18]). *Let λ be a security parameter. Let $\mathcal{X}$ and $\mathcal{Y}$ be finite sets. Let $\mathcal{K}_{\mathcal{F}}$ be a finite set of keys. A family of functions*

$$\mathcal{F} = \{f_{k,b} : \mathcal{X} \to \mathcal{D}_{\mathcal{Y}}\}_{k \in \mathcal{K}_{\mathcal{F}}, b \in \{0,1\}}$$

is called a noisy trapdoor claw-free (NTCF) family if the following conditions hold:

1. ***Efficient Function Generation.*** *There exists an efficient probabilistic algorithm $\textsc{Gen}_{\mathcal{F}}$ which generates a key $k \in \mathcal{K}_{\mathcal{F}}$ together with a trapdoor t_k:*
$$(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{F}}(1^\lambda) .$$
2. ***Trapdoor Injective Pair.***
   (a) *Trapdoor: There exists an efficient deterministic algorithm $\textsc{Inv}_{\mathcal{F}}$ such that with overwhelming probability over the choice of $(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{F}}(1^\lambda)$, the following holds:*
$$\text{for all } b \in \{0,1\}, x \in \mathcal{X} \text{ and } y \in \textsc{Supp}(f_{k,b}(x)), \textsc{Inv}_{\mathcal{F}}(t_k, b, y) = x.$$
   (b) *Injective pair: For all keys $k \in \mathcal{K}_{\mathcal{F}}$, there exists a perfect matching $\mathcal{R}_k \subseteq \mathcal{X} \times \mathcal{X}$ such that $f_{k,0}(x_0) = f_{k,1}(x_1)$ if and only if $(x_0, x_1) \in \mathcal{R}_k$.*
3. ***Efficient Range Superposition.*** *For all keys $k \in \mathcal{K}_{\mathcal{F}}$ and $b \in \{0,1\}$ there exists a function $f'_{k,b} : \mathcal{X} \to \mathcal{D}_{\mathcal{Y}}$ such that the following hold.*

(a) *For all* $(x_0, x_1) \in \mathcal{R}_k$ *and* $y \in \textsc{Supp}(f'_{k,b}(x_b))$, $\textsc{Inv}_{\mathcal{F}}(t_k, b, y) = x_b$ *and* $\textsc{Inv}_{\mathcal{F}}(t_k, b \oplus 1, y) = x_{b\oplus 1}$*.*

(b) *There exists an efficient deterministic procedure* $\textsc{Chk}_{\mathcal{F}}$ *that, on input* k*,* $b \in \{0,1\}$*,* $x \in \mathcal{X}$ *and* $y \in \mathcal{Y}$*, returns* 1 *if* $y \in \textsc{Supp}(f'_{k,b}(x))$ *and* 0 *otherwise. Note that* $\textsc{Chk}_{\mathcal{F}}$ *is not provided the trapdoor* t_k*.*

(c) *For every* k *and* $b \in \{0,1\}$*,*

$$\mathbb{E}_{x\leftarrow\mathcal{X}}\big[\, H^2(f_{k,b}(x),\, f'_{k,b}(x))\,\big] \;\leq\; \mu(\lambda)\ .$$

for some negligible function μ*. Here* H^2 *is the Hellinger distance. Moreover, there exists an efficient procedure* $\textsc{Samp}_{\mathcal{F}}$ *that on input* k *and* $b \in \{0,1\}$ *prepares the state*

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

4. ***Adaptive Hardcore Bit.*** *For all keys* $k \in \mathcal{K}_{\mathcal{F}}$ *the following conditions hold, for some integer* w *that is a polynomially bounded function of* λ*.*

(a) *For all* $b \in \{0,1\}$ *and* $x \in \mathcal{X}$*, there exists a set* $G_{k,b,x} \subseteq \{0,1\}^w$ *such that* $\Pr_{d\leftarrow_U\{0,1\}^w}[d \notin G_{k,b,x}]$ *is negligible, and moreover there exists an efficient algorithm that checks for membership in* $G_{k,b,x}$ *given* k, b, x *and the trapdoor* t_k*.*

(b) *There is an efficiently computable injection* $\mathcal{J} : \mathcal{X} \to \{0,1\}^w$*, such that* $\mathcal{J}$ *can be inverted efficiently on its range, and such that the following holds. If*

$$\begin{aligned} H_k &= \big\{(b, x_b, d, d\cdot(\mathcal{J}(x_0)\oplus\mathcal{J}(x_1))) \mid b \in \{0,1\}, (x_0, x_1) \in \mathcal{R}_k,\\ &\qquad d \in G_{k,0,x_0} \cap G_{k,1,x_1}\big\},\\ \overline{H}_k &= \{(b, x_b, d, c) \mid (b, x, d, c\oplus 1) \in H_k\big\}, \end{aligned}$$

then for any quantum polynomial-time procedure $\mathcal{A}$ *there exists a negligible function* $\mathsf{negl}(\cdot)$ *such that*

$$\left| \Pr_{(k,t_k)\leftarrow GEN_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(k) \in H_k] - \Pr_{(k,t_k)\leftarrow GEN_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(k) \in \overline{H}_k] \right| \leq \mathsf{negl}(\lambda).$$

Lemma 1 ([BCM$^+$18, Theorem 4.1]). *Assuming the quantum hardness of LWE, there exists an LWE-based NTCF family with an adaptive hardcore bit property.*

Here, we recall an inverting algorithm with trapdoor in [MP13], which is used in the LWE-based PoQ scheme [BCM$^+$18].

Theorem 3 ([MP13, Theorem 5.1]). *There is an efficient algorithm* $\mathsf{GenTrap}$ *that, on input* $1^n, q, m = \Omega(n \log q)$*, outputs a matrix* $\mathbf{A}$ *distributed statistically close to uniformly on* $\mathbb{Z}_q^{n\times m}$*, and a* $O(m)$*-good lattice trapdoor* td *for* $\mathbf{A}$*. Moreover, there is an efficient algorithm* Invert *that, on input* $\mathbf{A}, \mathsf{td}$ *and* $\mathbf{sA} + \mathbf{e}$ *where* $\|\mathbf{e}\| \leq q/(C_T\sqrt{n\log q})$ *and* C_T *is a universal constant, returns* $\mathbf{s}$ *and* $\mathbf{e}$ *with overwhelming probability over* $(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$*.*

2.2 Extractable-NIZK for NP in the CRS Model

Notice that, in the following definition from [JK25], we additionally consider a weaker notion of extractability, where the adversary is not allowed to query any simulated proof under the targeted CRS. We will denote the model simply by extractable-NIZK, where the simulation-extractability is replaced by extractability. It is easy to see that the simulation-extractable NIZK implies the extractable-NIZK. The latter one is sufficient for our purpose.

Definition 3 (Post-quantum (simulation-)extractable NIZK for NP in CRS Model [JK25]). *Let* NP *relation* $\mathcal{R}$ *with corresponding language* $\mathcal{L}$ *be given such that they can be indexed by a security parameter* $\lambda \in \mathbb{N}$*.*

$\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ *is a post-quantum (quantum) non-interactive simulation-extractable zero-knowledge argument for* NP *in the CRS model if it has the following syntax and properties.*
Syntax: *The input* 1^λ *is left out when it is clear from context.*

- $\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)$*: The probabilistic polynomial-size algorithm* Setup *on input* 1^λ *outputs a common reference string* crs*.*
- $\pi \leftarrow \mathsf{P}(1^\lambda, \mathsf{crs}, x, w)$*: The probabilistic (quantum) polynomial-size algorithm* P *on input a common reference string* crs *and instance and witness pair* $(x, w) \in \mathcal{R}_\lambda$*, outputs a proof* π*.*
- $\mathsf{V}(1^\lambda, \mathsf{crs}, x, \pi) \in \{0, 1\}$*: The probabilistic (quantum) polynomial-size algorithm* V *on input a common reference string* crs*, an instance* x*, and a proof* π *outputs* 1 *iff* π *is a valid proof for* x*.*

Properties: *A post-quantum simulation-extractable NIZK is secure if the following properties hold:*

- **Perfect Completeness.** *For every* $\lambda \in \mathbb{N}$ *and every* $(x, w) \in \mathcal{R}_\lambda$*,*

$$\Pr_{\substack{\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda) \\ \pi \leftarrow \mathsf{P}(\mathsf{crs}, x, w)}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1] = 1.$$

- **Adaptive Multi-Theorem Computational Zero-Knowledge.** *There exists a probabilistic (quantum) polynomial-size algorithm* $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ *and a negligible function* $\mathsf{negl}(\cdot)$ *such that for every polynomial-size quantum algorithm* $\mathcal{A}$*, and every sufficiently large* $\lambda \in \mathbb{N}$*,*

$$\mathsf{adv}_{\mathsf{zk}} = \left| \Pr_{\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)} [\mathcal{A}^{\mathsf{P}(\mathsf{crs}, \cdot, \cdot)}(\mathsf{crs}) = 1] - \Pr_{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda)} [\mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, \cdot)}(\mathsf{crs}) = 1] \right| \leq \mathsf{negl}(\lambda).$$

- **Simulation Extractability.** *Let* $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ *be the simulator given by the adaptive multi-theorem computational zero-knowledge property. There exists a polynomial-time extractor* Ext *and a negligible function* $\mathsf{negl}(\cdot)$ *such that for every oracle-aided polynomial-size quantum algorithm* $\mathcal{A}$ *and every* $\lambda \in \mathbb{N}$*,*

$$\Pr_{\substack{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}^{\mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, \cdot)}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, \pi)}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge x \notin Q \wedge (x, w) \notin \mathcal{R}] \leq \mathsf{negl}(\lambda),$$

  where Q *is the list of queries from* $\mathcal{A}$ *to* Sim_1*.*
- **Extractability.** *Let* Sim *be the simulator given by the adaptive multi-theorem computational zero-knowledge property. There exists a polynomial-time extractor* Ext *and a negligible function* $\mathsf{negl}(\cdot)$ *such that for every polynomial-size quantum algorithm* $\mathcal{A}$ *and every* $\lambda \in \mathbb{N}$*,*

$$\Pr_{\substack{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, \pi)}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge (x, w) \notin \mathcal{R}] \leq \mathsf{negl}(\lambda).$$

We emphasize that the extractor in the definition of extractability of the extractable-NIZK proof is purely classical, which is sufficient for our purpose. Looking ahead, one can notice that the classically secure extractable-NIZK proof is already enough for our transformation. However, we still recall the post-quantum secure version of extractable NIZK proof aiming for its potential applications in more functionalities such as certifying qubits and key leasing (as discussed in Section 1.1).

Next, we recall a result from [JK25] about an LWE-based construction satisfying the simulation-extractable NIZK definition.

Theorem 4 ([JK25, Corollary 4.4]). *Assuming the polynomial quantum hardness of LWE, there exists a post-quantum non-interactive simulation-extractable, adaptively multi-theorem computationally zero-knowledge argument for* NP *in the common reference string model (Definition 3).*

Note that the construction provided by [JK25] achieves a stronger notion than the one we need, but it is the closest one that we are aware of. Therefore, we take it as an example of extractable-NIZK for instantiating our transformation.

$\Sigma_{\mathsf{Factoring}}$: PoQ for Factoring N'

Fix a security parameter λ. Let $\mathcal{P}$ denote a quantum prover and $\mathcal{V}$ denote a classical verifier, and $\langle \mathcal{P}, \mathcal{V} \rangle$ be a QPIP system for a PoQ scheme. Let N' be the product of two primes p and q.

$\mathcal{V}$: Prepare two random large primes (p, q) and compute N'. Send N' to the $\mathcal{P}$.

$\mathcal{P}$: Compute $(p, q) \leftarrow \mathsf{Q.factoring}(N')$ by using Shor's quantum polynomial time algorithm $\mathsf{Q.factoring}(N')$. Send (p, q) to the $\mathcal{V}$.

$\mathcal{V}$: Check the validity of (p, q) and output $\langle \mathcal{P}, \mathcal{V} \rangle = 1$ if it passes; else output 0 and abort.

Fig. 5. $\Sigma_{\mathsf{Factoring}}$: Factoring-based Proofs of Quantumness Scheme.

2.3 Recall Approaches for Proofs of Quantumness

We first recall the definition of the quantum-prover interactive proof.

Definition 4 (QPIP: Quantum-prover interactive proof). *A quantum-prover interactive proof (QPIP) system is an interactive scheme between two polynomially bounded parties, a quantum prover and a classical verifier interacting over a classical channel.*

1. *A semi-honest QPIP system is described by a pair of algorithms: the PPT algorithm of the honest-but-curious verifier $\mathcal{V}$ and the QPT algorithm of prover $\mathcal{P}$;*
2. *A malicious QPIP system described by a pair of algorithms: the PPT algorithm of the malicious verifier $\mathcal{V}^*$ and the QPT algorithm of prover $\mathcal{P}$;*

We first state the straightforward quantum completeness and classical soundness of the proofs of quantumness (PoQ) scheme based on factoring.

Lemma 2 (Quantum Completeness). *Due to Shor's quantum polynomial-time factoring algorithm, a QPT prover, $\mathcal{P}$, following the honest strategy in the scheme $\Sigma_{\mathsf{Factoring}}$ shown in Figure 5 is accepted with probability* $1 - \mathsf{negl}(\lambda)$.

Lemma 3 (Classical Soundness). *Assuming large integer factoring is classically intractable for any PPT prover $\mathcal{P}'$, the prover $\mathcal{P}'$ can convince the $\mathcal{V}$ to accept with only negligible probability in scheme $\Sigma_{\mathsf{Factoring}}$ shown in Figure 5.*

Now we recall the quantum completeness and classical soundness of the PoQ scheme in [BCM$^+$18] as follows.

Lemma 4 (Quantum Completeness, [BCM$^+$18]). *A QPT prover, $\mathcal{P}$, following the honest strategy in the scheme Σ_{BCMVV} (as shown in Figure 6) is accepted with probability* $1 - \mathsf{negl}(\lambda)$.

Lemma 5 (Classical Soundness, [BCM$^+$18]). *Assume that LWE is classically intractable. For any PPT prover, $\mathcal{P}'$, in the parallel repetition version of the scheme Σ_{BCMVV} (as shown in Figure 6), the prover $\mathcal{P}'$ convinces the $\mathcal{V}$ to accept with negligible probability.*

3 Zero-Knowledge Proofs of Quantumness Based on Extractable-NIZK

In this section, we initially present the formal definition of zero-knowledge proofs of quantumness. Subsequently, we introduce two constructions that can be provably secure under this definition.

3.1 Zero-Knowledge Proofs of Quantumness

In the following definition, we will supplement the zero-knowledge notion to the existing proofs of quantumness definition. Our zero-knowledge property is also formalized by indistinguishability between real transcript and simulated one. In particular, our zero-knowledge protocol requires indistinguishability between the communication initiated by a quantum prover and the one by a classical prover when interacting with the same verifier.

Σ_{BCMVV}: LWE-based PoQ Scheme (Parallel repetition version)

Fix a security parameter λ and an NTCF family $\mathcal{F} = \{f'_{k,b} : \mathcal{X} \to \mathcal{D}_{\mathcal{Y}}\}_{k \in \mathcal{K}, b \in \{0,1\}}$ described by algorithms ($\textsc{Gen}_{\mathcal{F}}, \textsc{Samp}_{\mathcal{F}}, \textsc{Inv}_{\mathcal{F}}, \textsc{Chk}_{\mathcal{F}}$), assuming that LWE is classically intractable. Let $\mathcal{P}$ denote a quantum prover and $\mathcal{V}$ denote a classical verifier, and $\langle \mathcal{P}, \mathcal{V} \rangle$ be a QPIP system for a PoQ scheme. Repeat the following steps λ times:

$\mathcal{V}$: Prepare $(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{F}}(1^\lambda)$ and send k to the prover $\mathcal{P}$, where $k = (\mathbf{A}', \mathbf{A}'\mathbf{s}' + \mathbf{e}')$ and $t_k = \mathsf{td}_{\mathbf{A}'}$ is the trapdoor.

$\mathcal{P}$: Run $\textsc{Samp}_{\mathcal{F}}(1^\lambda)$ and measure the image register to yield a string $\mathbf{y}$, send $\mathbf{y}$ to the verifier $\mathcal{V}$.

$\mathcal{V}$: Sample a uniformly random challenge bit $c \overset{\$}{\leftarrow} \{0,1\}$ and send c to the prover $\mathcal{P}$.

$\mathcal{P}$: Take in the challenge c, do:

- Preimage test (if $c = 0$): Perform a standard basis measurement, return a pair $(b, \mathbf{x}) \in \{0,1\} \times \{0,1\}^n$ as the proof $\sigma_0 = (b, \mathbf{x})$.
- Equation test (if $c = 1$): Perform a Hadamard basis measurement, return a pair $(u, \mathbf{d}) \in \{0,1\} \times \{0,1\}^n$ as the proof $\sigma_1 = (u, \mathbf{d})$.
- Send σ_c to the verifier $\mathcal{V}$.

$\mathcal{V}$: Take in $(t_k, \mathbf{y}, c, \sigma_c)$ do:

- Compute $(\mathbf{x}_0, \mathbf{x}_1) \leftarrow \textsc{Inv}_{\mathcal{F}}(1^\lambda, t_k, \mathbf{y})$, where $\mathbf{x}_1 = \mathbf{x}_0 - \mathbf{s}' \mod q$.
- Check the validity of σ_c and output $\langle \mathcal{P}, \mathcal{V} \rangle$, which is defined as

$$\langle \mathcal{P}, \mathcal{V} \rangle := \begin{cases} 1 & \text{if } c = 0 \text{ and } \textsc{Chk}_{\mathcal{F}}(k, \mathbf{y}, b, \mathbf{x}) = 1, \\ 1 & \text{if } c = 1, d \in G_{k,0,x_0} \cap G_{k,1,x_1} \text{ and } \mathbf{d}^\top \cdot (\mathbf{x}_0 \oplus \mathbf{x}_1) \bmod 2 = u, \\ 0 & \text{otherwise.} \end{cases}$$

At the end of the λ rounds, if the verifier $\mathcal{V}$ has not aborted it accepts.

Fig. 6. Σ_{BCMVV}: LWE-based Proofs of Quantumness Scheme [BCM+18].

Definition 5 (Zero-knowledge proofs of quantumness, ZKPoQ). *Let $\mathcal{P}$ denote a quantum prover and $\mathcal{V}$ denote a classical verifier. we say that $\langle \mathcal{P}, \mathcal{V} \rangle$ is a QPIP system for zero-knowledge proofs of quantumness if the following properties are satisfied:*

- ***Quantum Completeness**: Let $\lambda \in \mathbb{N}$ be the security parameter. Given the QPT prover $\mathcal{P}$, there exists a negligible function in λ such that:*

$$\Pr[\langle \mathcal{V}(1^\lambda), \mathcal{P}(1^\lambda) \rangle = 1] \geq 1 - \operatorname{negl}(\lambda)$$

- ***Classical Soundness**: For any PPT prover $\mathcal{P}'$, there exists a negligible function in λ such that:*

$$\Pr[\langle \mathcal{V}(1^\lambda), \mathcal{P}'(1^\lambda) \rangle = 1] \leq \operatorname{negl}(\lambda)$$

- ***Computational Zero-Knowledge**: For any PPT verifier $\mathcal{V}^*$ and QPT prover $\mathcal{P}$, there exists a PPT simulator algorithm $\mathsf{Sim}_{\mathcal{V}^*}$ has assess to oracle $\mathcal{V}^*$, such that:*

$$\mathsf{Sim}_{\mathcal{V}^*}(1^\lambda) \approx_c \mathsf{View}[\langle \mathcal{V}^*(1^\lambda) \leftrightarrow \mathcal{P}(1^\lambda) \rangle]$$

where $\approx_c$ represents computational indistinguishability.

3.2 Construction of a Factoring ZKPoQ Scheme

We refer to Figure 7 for our factoring-based ZKPoQ scheme.

Theorem 5. *Assuming that the LWE problem is classically intractable, the protocol $\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$ described in Figure 7 is a ZKPoQ (Definition 5) scheme satisfying quantum completeness, classical soundness, and computational zero-knowledge.*

$\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$: ZKPoQ for Factoring N

Fix a security parameter λ. Let $\mathcal{P}$ denote a quantum prover and $\mathcal{V}$ denote a classical verifier. Let $\langle \mathcal{P}, \mathcal{V} \rangle$ be a QPIP system for ZKPoQ scheme described by algorithms (Setup, Nizk, Prove, Verify). Let N be the product of two primes p and q. Let $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ be an extractable non-interactive, adaptively multi-theorem computationally zero-knowledge (extractable-NIZK) scheme for factoring N.

Setup(1^λ):

- $(\mathsf{crs}, \mathsf{td}) \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$.

$\mathcal{V}$: Nizk(crs):

- Prepare large primes (p, q) and compute N.
- Compute proof $\pi \leftarrow \Pi.\mathsf{P}(\mathsf{crs}, N, (p, q))$ for factoring N.
- Send (N, π) to quantum prover $\mathcal{P}$.

$\mathcal{P}$: Prove(crs, N, π):

- Compute $\Pi.\mathsf{V}(\mathsf{crs}, N, \pi)$ and continue if it passes; else output $\perp$ and abort.
- Compute quantumness proof $\sigma \leftarrow \mathsf{Q.factoring}(N)$.
- Send $\sigma = (p, q)$ to classical verifier $\mathcal{V}$.

$\mathcal{V}$: Verify(σ):

- Check the validity of σ and output $\langle \mathcal{P}, \mathcal{V} \rangle = 1$ if it passes; else output 0 and abort.

Fig. 7. $\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$: Factoring-based Zero-knowledge Proofs of Quantumness Scheme.

The Theorem 5 follows from the following Lemmata 6, 7 and 8. The completeness of our factoring-based ZKPoQ scheme is stated below.

Lemma 6 (Quantum Completeness). *The scheme* $\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$ *satisfies quantum completeness of ZKPoQ in Definition 5.*

Proof. Completeness follows from completeness of the Extractable-NIZK, and completeness of PoQ Scheme $\Sigma_{\mathsf{Factoring}}$ shown in Figure 5.

The classical soundness of our factoring-based ZKPoQ scheme is given as follows. In particular, the classical soundness of our factoring-based ZKPoQ scheme is based on the zero-knowledge property of the extractable-NIZK proof as well as the classical soundness of the original factoring-based PoQ scheme.

Lemma 7 (Classical Soundness). *The scheme* $\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$ *satisfies the classical soundness of ZKPoQ in Definition 5.*

Proof. We proceed by contradiction. Assuming that there exists a PPT adversary, denoted as $\mathcal{A}$, who can break the classical soundness of scheme $\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$ with a non-negligible advantage ε_0. We define the following three games. Let $\mathsf{adv}_i(\mathcal{A})$ denote the advantage of the adversary $\mathcal{A}$ in the Game_i for $i = \{0, 1, 2\}$, respectively.

- $\underline{\mathsf{Game}_0}$: Let Game_0 be the same as the real scheme $\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$ in Figure 7.
- $\underline{\mathsf{Game}_1}$: Let Game_1 be the same as Game_0, except the generation of $(\mathsf{crs}, \mathsf{td}) \leftarrow \textsc{Setup}(1^\lambda)$ is replaced by $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda)$, and the proof π generated by the classical verifier $\mathcal{V}$ is also replaced by the simulated one $\pi \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, N)$ correspondingly, where $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ is the simulator corresponding to the adaptive multi-theorem computational zero-knowledge property.
- $\underline{\mathsf{Game}_2}$: Let Game_2 be the same as Game_1, except that the N is replaced by the N' generated by the challenger for the soundness of the scheme $\Sigma_{\mathsf{Factoring}}$. We will also forward the response (p, q) from the prover to the challenger.

$\underline{\mathsf{Game}_0 \approx_c \mathsf{Game}_1}$: This follows directly from the property of the adaptive multi-theorem computational zero-knowledge (Definition 3). Thus, $\mathsf{adv}_0(\mathcal{A}) \leq \mathsf{adv}_1(\mathcal{A}) + \mathsf{adv}_{\mathsf{zk}} \leq \mathsf{adv}_1(\mathcal{A}) + \mathsf{negl}(\lambda)$.

$\underline{\mathsf{Game}_1 \equiv \mathsf{Game}_2}$: The Game_1 and Game_2 are identical as the distributions of N in the ZKPoQ scheme and N' from the (honest) challenger for the original PoQ scheme are the same. Thus, we have $\mathsf{adv}_2(\mathcal{A}) = \mathsf{adv}_1(\mathcal{A})$.

First, we claim that $\mathsf{adv}_2(\mathcal{A}) \leq \varepsilon'$, where the latter one denotes the advantage of any classical PPT adversary $\mathcal{A}'$ against the soundness of the original PoQ scheme $\Sigma_{\mathsf{Factoring}}$. This is because a correct response (p, q) (as factorization of N') the adversary $\mathcal{A}$ in Game_2 will also be the correct solution to the challenge provided by the challenger for the soundness of the original PoQ scheme. Thus, we have $\varepsilon_0 = \mathsf{adv}_0(\mathcal{A}) \leq \mathsf{adv}_1(\mathcal{A}) + \mathsf{negl}(\lambda) = \mathsf{adv}_2(\mathcal{A}) + \mathsf{negl}(\lambda) \leq \varepsilon' + \mathsf{negl}(\lambda)$. Based on the soundness of the original PoQ scheme, we have $\varepsilon' \leq \mathsf{negl}(\lambda)$ and therefore $\varepsilon_0 \leq \mathsf{negl}(\lambda)$. The lemma can be concluded by contradiction.

Next, we show that our factoring-based ZKPoQ scheme also enjoys the zero-knowledge property. Note that in this security notion, we will consider a malicious verifier who will try to extract unexpected information from the prover. Notably, we can not assume the number N sent by the malicious verifier will be honestly chosen, which makes the construction of the simulator for zero-knowledge slightly more involved. As mentioned in Section 1.1, the zero-knowledge property of our factoring-based ZKPoQ scheme is based on the extractability of the extractable-NIZK proof.

Lemma 8 (Computational Zero-Knowledge). *The scheme* $\mathsf{ZK}.\Sigma_{\mathsf{Factoring}}$ *satisfies computational zero-knowledge of ZKPoQ in Definition 5.*

Proof. In the following, we provide the construction of the PPT simulator $\mathsf{Sim}_{\mathcal{V}^*}$ depending solely on the classical verifier $\mathcal{V}^*$.

$\mathsf{Sim}_{\mathcal{V}^*}(1^\lambda)$:

- Run $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}(1^\lambda)$ and store the td.
- Run verifier $\mathcal{V}^*$ on crs to get N and extractable-NIZK proof π for proving the NP relation $(N, (p, q))$.
- Run polynomial-time (classical) extractor Ext to compute $w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, \pi)$, where the witness $w = (p, q)$ and $x = N$. If this fails, then we abort.
- Feed the verifier $\mathcal{V}^*$ with (p, q) and the simulation is finished.

First, the distributions crs from the setup algorithm $\mathsf{Setup}(1^\lambda)$ and the simulation algorithm $\mathsf{Sim}(1^\lambda)$ are computationally indistinguishable (that is implicit in the zero-knowledge property of the extractable-NIZK proof). In addition, since the probability of extractable-NIZK's extractor Ext failing to extract (p, q) is negligible, the probability of simulator $\mathsf{Sim}_{\mathcal{V}^*}$ abort is also negligible. Thus, we have $\mathsf{Sim}_{\mathcal{V}^*}(1^\lambda) \approx_c \mathsf{View}[\langle \mathcal{V}^*(1^\lambda) \leftrightarrow \mathcal{P}(1^\lambda) \rangle]$.

3.3 Construction of the LWE-based ZKPoQ Scheme

We refer to Figure 8 for our LWE-based ZKPoQ scheme.

Theorem 6. *Assuming the post-quantum LWE problem, the protocol* $\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}$ *described in Figure 8 is a ZKPoQ (Definition 5) scheme satisfying quantum completeness, classical soundness, and computational zero-knowledge.*

The Theorem 6 follows from the Lemmata 9, 10 and 11. The completeness is given as follows.

Lemma 9 (Quantum Completeness). *The scheme* $\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}$ *satisfies quantum completeness of ZKPoQ in Definition 5.*

Proof. Completeness follows from completeness of the extractable-NIZK, and completeness of PoQ scheme Σ_{BCMVV} shown in Figure 6.

The classical soundness of our LWE-based ZKPoQ scheme is given as follows. To prove the classical soundness of the LWE-based ZKPoQ scheme, we need to rely on both the zero-knowledge property of the extractable-NIZK proof as well as the classical soundness of the original LWE-based PoQ scheme.

$\underline{\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}\text{: ZKPoQ for LWE-based Scheme (Parallel repetition version)}}$

Fix a security parameter λ and an NTCF family $\mathcal{F} = \{f'_{k,b} : \mathcal{X} \to \mathcal{D}_{\mathcal{Y}}\}_{k \in \mathcal{K}, b \in \{0,1\}}$ described by algorithms ($\textsc{Gen}_{\mathcal{F}}, \textsc{Samp}_{\mathcal{F}}, \textsc{Inv}_{\mathcal{F}}, \textsc{Chk}_{\mathcal{F}}$), assuming the post-quantum hardness of LWE. Let $\mathcal{P}$ denote a quantum prover and $\mathcal{V}$ denote a classical verifier. Let $\langle \mathcal{P}, \mathcal{V} \rangle$ be a QPIP system for ZKPoQ scheme described by algorithms ($\textsc{Setup}, \textsc{Nizk}, \textsc{Prove}, \textsc{Verify}$). Let $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ be an extractable non-interactive, adaptively multi-theorem computationally zero-knowledge (extractable-NIZK) scheme for factoring N. Repeat the following steps λ times:
$\underline{\textsc{Setup}}(1^\lambda)$: $(\mathsf{crs}, \mathsf{td}) \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$.
$\mathcal{V}$: $\underline{\textsc{Nizk}}(\mathsf{crs})$:

- Prepare $(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{F}}(1^\lambda)$, where $k = (\mathbf{A}, \mathbf{As} + \mathbf{e})$ and $t_k = \mathsf{td}_{\mathbf{A}}$.
- Recover LWE secret $\mathbf{s}$ from k via t_k using algorithm from Lemma 3.
- Compute proof $\pi \leftarrow \Pi.\mathsf{P}(\mathsf{crs}, (\mathbf{A}, \mathbf{As} + \mathbf{e}), \mathbf{s})$.
- Send (k, π) to the prover $\mathcal{P}$.

$\mathcal{P}$: $\underline{\textsc{Prove}_1}(\mathsf{crs}, k, \pi)$:

- Compute $\Pi.\mathsf{V}(\mathsf{crs}, k, \pi)$ and continue if it passes; else output $\perp$ and abort.
- Run $\textsc{Samp}_{\mathcal{F}}(1^\lambda)$ and measure the image register to yield an string $\mathbf{y}$, send $\mathbf{y}$ to the verifier $\mathcal{V}$. Note that the $\mathbf{y} = f'_{k,b}(\mathbf{x})$ is distributed over random $b \xleftarrow{\$} \{0,1\}$ and $\mathbf{x} \xleftarrow{\$} \mathcal{X}$.

$\mathcal{V}$: Sample a uniformly random challenge bit $c \xleftarrow{\$} \{0,1\}$ and send c to the prover $\mathcal{P}$
$\mathcal{P}$: $\underline{\textsc{Prove}_2}(c, \rho)$:

- Preimage test (if $c = 0$): Perform a standard basis measurement, return a pair $(b, \mathbf{x}) \in \{0,1\} \times \{0,1\}^n$ as the proof $\sigma_0 = (b, \mathbf{x})$. In this case, the response of the prover is a random one of the two preimages $(b = 0, \mathbf{x}_0 = \mathbf{x})$ and $(b = 1, \mathbf{x}_1 = \mathbf{x}_0 - \mathbf{s})$ of $\mathbf{y}$ under the function $f'_{k,b}(\mathbf{x}_b)$.
- Equation test (if $c = 1$): Perform a Hadamard basis measurement, return a pair $(u, \mathbf{d}) \in \{0,1\} \times \{0,1\}^n$ as the proof $\sigma_1 = (u, \mathbf{d})$. In this case, the response of the prover is $(u, \mathbf{d})$ such that $\mathbf{d}$ is random and $u = \mathbf{d}^\top \cdot (\mathbf{x}_0 \oplus \mathbf{x}_1) \bmod 2$.
- Send σ_c to the verifier $\mathcal{V}$.

$\mathcal{V}$: $\underline{\textsc{Verify}}(t_k, \mathbf{y}, c, \sigma_c)$:

- Compute $(\mathbf{x}_0, \mathbf{x}_1) \leftarrow \textsc{Inv}_{\mathcal{F}}(1^\lambda, t_k, \mathbf{y})$, where $\mathbf{x}_1 = \mathbf{x}_0 - \mathbf{s} \mod q$.
- Check the validity of σ_c and Output $\langle \mathcal{P}, \mathcal{V} \rangle$, which is defined as

$$\langle \mathcal{P}, \mathcal{V} \rangle := \begin{cases} 1 & \text{if } c = 0 \text{ and } \textsc{Chk}_{\mathcal{F}}(k, \mathbf{y}, b, \mathbf{x}) = 1, \\ 1 & \text{if } c = 1, d \in G_{k,0,x_0} \cap G_{k,1,x_1} \text{ and } \mathbf{d}^\top \cdot (\mathbf{x}_0 \oplus \mathbf{x}_1) \bmod 2 = u, \\ 0 & \text{otherwise.} \end{cases}$$

At the end of the λ rounds, if the verifier $\mathcal{V}$ has not output 0 it accepts.

Fig. 8. $\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}$: LWE-based Zero-Knowledge Proofs of Quantumness Scheme.

Lemma 10 (Classical Soundness). *The scheme* $\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}$ *satisfies the classical soundness of ZKPoQ in Definition 5.*

Proof. Similarly, we proceed by contradiction. Assuming that there exists a probabilistic polynomial-time (PPT) adversary, denoted as $\mathcal{A}$, who can break the classical soundness of scheme $\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}$ with a non-negligible advantage ε_0. We define the following three games. Let $\mathsf{adv}_i(\mathcal{A})$ denote the advantage of the adversary $\mathcal{A}$ in the Game_i for $i = \{0, 1, 2\}$, respectively.

- $\underline{\mathsf{Game}_0}$: Let Game_0 be the same as the real scheme $\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}$ in Figure 8.
- $\underline{\mathsf{Game}_1}$: Let Game_1 be the same as Game_0, except that the generation of $(\mathsf{crs}, \mathsf{td}) \leftarrow \textsc{Setup}(1^\lambda)$ is replaced by $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda)$, and the proof π generated by the classical verifier $\mathcal{V}$ is also replaced by the simulated one $\pi \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, k)$ with $k = (\mathbf{A}, \mathbf{As} + \mathbf{e})$ correspondingly, where $\mathsf{Sim} = (\mathsf{Sim}_0, \mathsf{Sim}_1)$ is the simulator corresponding to the adaptive multi-theorem computational zero-knowledge property.
- $\underline{\mathsf{Game}_2}$: Let Game_2 be the same as Game_1, except that all transcripts other than the extractable-NIZK proof from the verifier to the prover are replaced by the corresponding transcripts from the challenger for the soundness of the original scheme Σ_{BCMVV}. These transcripts will include k and c. All transcripts from the prover are also forwarded to the challenger correspondingly. These transcripts will include $\mathbf{y}$ and σ_c.

$\underline{\mathsf{Game}_0 \approx_c \mathsf{Game}_1}$: This follows directly from the property of the adaptive multi-theorem computational zero-knowledge (Definition 3). Thus, $\mathsf{adv}_0(\mathcal{A}) \leq \mathsf{adv}_1(\mathcal{A}) + \mathsf{adv}_{\mathsf{zk}} \leq \mathsf{adv}_1(\mathcal{A}) + \mathsf{negl}(\lambda)$.

$\underline{\mathsf{Game}_1 \equiv \mathsf{Game}_2}$: The Game_1 and Game_2 are identical because the distributions all transcripts except the extractable-NIZK proof from the verifier to the prover in the ZKPoQ scheme and the ones from the (honest) challenger for the original PoQ scheme are the same. Thus, we have $\mathsf{adv}_2(\mathcal{A}) = \mathsf{adv}_1(\mathcal{A})$.

First, we claim that $\mathsf{adv}_2(\mathcal{A}) \leq \varepsilon'$, where the latter one denotes the advantage of any classical PPT adversary $\mathcal{A}'$ against the soundness of the original PoQ scheme Σ_{BCMVV}. This is because once the adversary $\mathcal{A}$ succeeds in Game_2, then we can also pass the challenge proposed by the challenger for the soundness of the original PoQ scheme Σ_{BCMVV}. Therefore, we have $\varepsilon' \geq \mathsf{adv}_2(\mathcal{A})$. Thus, $\mathsf{adv}_0(\mathcal{A}) \leq \mathsf{adv}_1(\mathcal{A}) + \mathsf{negl}(\lambda) = \mathsf{adv}_2(\mathcal{A}) + \mathsf{negl}(\lambda) \leq \varepsilon' + \mathsf{negl}(\lambda)$. Based on the soundness of the original PoQ scheme, we have $\varepsilon' \leq \mathsf{negl}(\lambda)$ and therefore $\varepsilon_0 \leq \mathsf{negl}(\lambda)$. The lemma can be concluded by contradiction.

Next, we proceed to prove that our LWE-based ZKPoQ scheme also enjoys the zero-knowledge property. We follow a similar manner as before to first extract the witness of the extractable-NIZK proof, which can then be used to properly simulate all responses from the quantum prover (as discussed in Section 1.1). Therefore, the zero-knowledge property of our factoring-based ZKPoQ scheme is also based on the extractability of the extractable-NIZK proof.

Lemma 11 (Computational Zero-Knowledge). *The scheme* $\mathsf{ZK}.\Sigma_{\mathsf{BCMVV}}$ *satisfies computational zero-knowledge of ZKPoQ in Definition 5.*

Proof. In the following, we provide the construction of the PPT simulator $\mathsf{Sim}_{\mathcal{V}^*}$ depending solely on the classical verifier $\mathcal{V}^*$.

$\mathsf{Sim}_{\mathcal{V}^*}(1^\lambda)$:

1. Run $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}(1^\lambda)$ and store the trapdoor td.
2. Run verifier $\mathcal{V}^*$ on crs to get $k = (\mathbf{A}, \mathbf{As} + \mathbf{e})$ and an extractable-NIZK proof π for proving NP relation $(k = (\mathbf{A}, \mathbf{As} + \mathbf{e}), \mathbf{s})$.
3. Run polynomial-time (classical) extractor Ext to compute $w \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, x, \pi)$, where the witness $w = \mathbf{s}$ and the statement $x = (\mathbf{A}, \mathbf{As} + \mathbf{e})$. If this fails, then we abort.
4. Pick a random pair $(b, \mathbf{x})$ where $b \xleftarrow{\$} \{0, 1\}$ and $\mathbf{x} \xleftarrow{\$} \mathcal{X}$. Compute $\mathbf{y} = f'_{k,b}(\mathbf{x})$ classically and feed the verifier $\mathcal{V}^*$ with the next input $\mathbf{y}$. This produces $\mathbf{y}$ with the same distribution as in the real scheme.
5. Receive from the verifier $\mathcal{V}^*$ the next output challenge c, if $c = 0$, feed the verifier $\mathcal{V}^*$ with next input $(b, \mathbf{x})$; otherwise sample a random $\mathbf{d} \in \{0, 1\}^n$, compute $u = \mathbf{d}^\top \cdot (\mathbf{x} \oplus (\mathbf{x} - \mathbf{s})) \bmod 2$, then feed the verifier $\mathcal{V}^*$ with the next input $(u, \mathbf{d})$ as the equation test result. Note that both the pair $(b, \mathbf{x} - b \cdot \mathbf{s})$ and the pair $(u, \mathbf{d})$ have the same distribution as in the real scheme.

6. Repeat the above Step 2–Step 5 procedures λ times.

First, the distributions of crs from the setup algorithm $\mathsf{Setup}(1^\lambda)$ and the simulation algorithm $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}(1^\lambda)$ are computationally indistinguishable. In addition, since the probability of extractable-NIZK's extractor Ext failing to extract $w = \mathbf{s}$ is negligible, the probability of simulator $\mathsf{Sim}_{\mathcal{V}^*}$ abort is also negligible.

One can also verify that the simulated transcripts from the prover to the verifier have exactly the same distribution as the ones in the real protocol. First, it is clear that the $\mathbf{y} = f'_{k,b}(\mathbf{x})$ with both b and $\mathbf{x}$ randomly chosen, has the same distribution as the one produced in the real protocol. Then, in the case of $c = 0$, the $(b, \mathbf{x})$ is a random one of two preimages $(0, \mathbf{x}_b)$ and $(1, \mathbf{x}_1)$ of $\mathbf{y}$ under the function $f'_{k,b}(\mathbf{x}_b)$, as the real case. The last case is when $c = 1$, the $(u, \mathbf{d})$ enjoys the relation $u = \mathbf{d}^\top \cdot (\mathbf{x} \oplus (\mathbf{x} - \mathbf{s})) \bmod 2$ for a random $\mathbf{d}$, therefore also has the same distribution as the real one. Thus, we have $\mathsf{Sim}_{\mathcal{V}^*}(1^\lambda) \approx_c \mathsf{View}[\langle \mathcal{V}^*(1^\lambda) \leftrightarrow \mathcal{P}(1^\lambda)\rangle]$.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China (No. 2022YFB2702700), by the European Union's Horizon Europe research and innovation programme under the project "Quantum Secure Networks Partnership" (QSNP, No. 101114043), by the France 2030 ANR Project ANR-22-PECY-003 SecureCompute, and by the ANR Project ANR-24-CE48-4293 HELO.

References

AA11. Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In 43rd Annual ACM Symposium on Theory of Computing, pages 333–342, 2011.

AAB+19. Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. Nature, 574(7779):505–510, 2019.

ACGH20. Gorjan Alagic, Andrew M Childs, Alex B Grilo, and Shih-Han Hung. Non-interactive classical verification of quantum computation. In Theory of Cryptography Conference, pages 153–180. Springer, 2020.

AMMW24. Yusuf Alnawakhtha, Atul Mantri, Carl A Miller, and Daochen Wang. Lattice-based quantum advantage from rotated measurements. Quantum, 8:1399, 2024.

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

BG92. Mihir Bellare and Oded Goldreich. On defining proofs of knowledge. In Advances in Cryptology - CRYPTO 1992, pages 390–420. Springer, 1992.

BGKM+23. Zvika Brakerski, Alexandru Gheorghiu, Gregory D. Kahanamoku-Meyer, Eitan Porat, and Thomas Vidick. Simple tests of quantumness also certify qubits. In Advances in Cryptology – CRYPTO 2023, pages 162–191, Cham, 2023. Springer Nature Switzerland.

BJS11. Michael J Bremner, Richard Jozsa, and Dan J Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 467(2126):459–472, 2011.

BKVV20. Zvika Brakerski, Venkata Koppula, Umesh Vazirani, and Thomas Vidick. Simpler proofs of quantumness. In 15th Conference on the Theory of Quantum Computation, Communication and Cryptography, 2020.

CGJL23. Orestis Chardouvelis, Vipul Goyal, Aayush Jain, and Jiahui Liu. Quantum key leasing for PKE and FHE with a classical lessor. Cryptology ePrint Archive, Paper 2023/1640, 2023.

GMW86. Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity and a methodology of cryptographic protocol design. In 27th Annual Symposium on Foundations of Computer Science, 1986.

JK25. Ruta Jawale and Dakshita Khurana. Unclonable non-interactive zero-knowledge. In Advances in Cryptology – ASIACRYPT 2024, pages 94–128, Singapore, 2025. Springer Nature Singapore.

KLVY23. Yael Kalai, Alex Lombardi, Vinod Vaikuntanathan, and Lisa Yang. Quantum advantage from any non-local game. In 55th Annual ACM Symposium on Theory of Computing, pages 1617–1628, 2023.

KMCVY22. Gregory D Kahanamoku-Meyer, Soonwon Choi, Umesh V Vazirani, and Norman Y Yao. Classically verifiable quantum advantage from a computational bell test. Nature Physics, 18(8):918–924, 2022.

MLA+22. Lars S Madsen, Fabian Laudenbach, Mohsen Falamarzi Askarani, Fabien Rortais, Trevor Vincent, Jacob FF Bulmer, Filippo M Miatto, Leonhard Neuhaus, Lukas G Helt, Matthew J Collins, et al. Quantum computational advantage with a programmable photonic processor. Nature, 606(7912):75–81, 2022.

MP13. Daniele Micciancio and Chris Peikert. Hardness of sis and lwe with small parameters. In Advances in Cryptology – CRYPTO 2013, pages 21–39, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

Reg24. Oded Regev. An efficient quantum factoring algorithm. Journal of the ACM, 2024.

Sho94. Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In 35th Annual Symposium on Foundations of Computer Science, pages 124–134. IEEE, 1994.

WBC+21. Yulin Wu, Wan-Su Bao, Sirui Cao, Fusheng Chen, Ming-Cheng Chen, Xiawei Chen, Tung-Hsun Chung, Hui Deng, Yajie Du, Daojin Fan, et al. Strong quantum computational advantage using a superconducting quantum processor. Physical Review Letters, 127(18):180501, 2021.