Secure Quantum Extraction Protocols

Prabhanjan Ananth[1](✉) and Rolando L. La Placa[2]

[1] University of California, Santa Barbara, USA
prabhanjan@cs.ucsb.edu
[2] MIT, Cambridge, USA
rlaplaca@mit.edu

Abstract. Knowledge extraction, typically studied in the classical setting, is at the heart of several cryptographic protocols. The prospect of quantum computers forces us to revisit the concept of knowledge extraction in the presence of quantum adversaries.

We introduce the notion of secure quantum extraction protocols. A secure quantum extraction protocol for an NP relation $\mathcal{R}$ is a classical interactive protocol between a sender and a receiver, where the sender gets as input the instance $\mathbf{y}$ and witness $\mathbf{w}$ while the receiver only gets the instance $\mathbf{y}$ as input. There are two properties associated with a secure quantum extraction protocol: (a) *Extractability*: for any efficient quantum polynomial-time (QPT) adversarial sender, there exists a QPT extractor that can extract a witness $\mathbf{w}'$ such that $(\mathbf{y}, \mathbf{w}') \in \mathcal{R}$ and, (b) *Zero-Knowledge*: a malicious receiver, interacting with the sender, should not be able to learn any information about $\mathbf{w}$.

We study and construct two flavors of secure quantum extraction protocols.

- **Security against QPT malicious receivers**: First we consider the setting when the malicious receiver is a QPT adversary. In this setting, we construct a secure quantum extraction protocol for NP assuming the existence of quantum fully homomorphic encryption satisfying some mild properties (already satisfied by existing constructions [Mahadev, FOCS'18, Brakerski CRYPTO'18]) and quantum hardness of learning with errors. The novelty of our construction is a new non-black-box technique in the quantum setting. All previous extraction techniques in the quantum setting were solely based on quantum rewinding.
- **Security against classical PPT malicious receivers**: We also consider the setting when the malicious receiver is a classical probabilistic polynomial time (PPT) adversary. In this setting, we construct a secure quantum extraction protocol for NP solely based on the quantum hardness of learning with errors. Furthermore, our construction satisfies *quantum-lasting security*: a malicious receiver cannot later, long after the protocol has been executed, use a quantum computer to extract a valid witness from the transcript of the protocol.

Both the above extraction protocols are *constant round* protocols.

We present an application of secure quantum extraction protocols to zero-knowledge (ZK). Assuming quantum hardness of learning with

R. Pass and K. Pietrzak (Eds.): TCC 2020, LNCS 12552, pp. 123–152, 2020.
https://doi.org/10.1007/978-3-030-64381-2_5

errors, we present the first construction of ZK argument systems for NP in constant rounds based on the quantum hardness of learning with errors with: (a) zero-knowledge against QPT malicious verifiers and, (b) soundness against classical PPT adversaries. Moreover, our construction satisfies the stronger (quantum) auxiliary-input zero knowledge property and thus can be composed with other protocols secure against quantum adversaries.

1 Introduction

Knowledge extraction is a quintessential concept employed to argue the security of classical zero-knowledge systems and secure two-party and multi-party computation protocols. The seminal work of Feige, Lapidot and Shamir [19] shows how to leverage knowledge extraction to construct zero-knowledge protocols. The ideal world-real world paradigm necessarily requires the simulator to be able to extract the inputs of the adversaries to argue the security of secure computation protocols.

Typically, knowledge extraction is formalized by defining a knowledge extractor that given access to the adversarial machine, outputs the input of the adversary. The prototypical extraction technique employed in several cryptographic protocols is rewinding. In the rewinding technique, the extractor, with oracle access to the adversary, rewinds the adversary to a previous state to obtain more than one protocol transcript which in turn gives the ability to the extractor to extract from the adversary. While rewinding has proven to be quite powerful, it has several limitations [22]. Over the years, cryptographers have proposed novel extraction techniques to circumvent the barriers of rewinding. Each time a new extraction technique was invented, it has advanced the field of zero-knowledge and secure computation. As an example, the breakthrough work of Barak [7] proposed a non-black-box extraction technique – where the extractor crucially uses the code of the verifier for extraction – and used this to obtain the first feasibility result on constant-round public-coin zero-knowledge argument system for NP. Another example is the work of Pass [35] who introduced the technique of super-polynomial time extraction and presented the first feasibility result on 2-round concurrent ZK argument system albeit under a weaker simulation definition.

Extracting from Quantum Adversaries. The prospect of quantum computers introduces new challenges in the design of zero-knowledge and secure computation protocols. As a starting step towards designing these protocols, we need to address the challenge of knowledge extraction against quantum adversaries. So far, the only technique used to extract from quantum adversaries is quantum rewinding [42], which has already been studied by a few works [3,27,38,40,42] in the context of quantum zero-knowledge protocols.

Rewinding a quantum adversary, unlike its classical counterpart, turns out to be tricky due to two reasons, as stated in Watrous [42]: firstly, intermediate quantum states of the adversary cannot be copied (due to the universal no-cloning theorem) and secondly, if the adversary performs some measurements then this adversary cannot be rewound since measurements in general are irreversible processes. As a result, the existing quantum rewinding techniques tend to be "oblivious" [38], to rewind the adversary back to an earlier point, the extraction should necessarily forget all the information it has learnt from that point onwards. As a result of these subtle issues, the analysis of quantum rewinding turns out to be quite involved making it difficult to use it in the security proofs. Moreover, existing quantum rewinding techniques [38,42] pose a bottleneck towards achieving a constant round extraction technique; we will touch upon this later.

In order to advance the progress of constructing quantum-secure (or post-quantum) cryptographic protocols, it is necessary that we look beyond quantum rewinding and explore new quantum extraction techniques.

1.1 Results

We introduce and study new techniques that enable us to extract from quantum adversaries.

Our Notion: Secure Quantum Extraction Protocols. We formalize this by first introducing the notion of secure quantum extraction protocols. This is a classical interactive protocol between a sender and a receiver and is associated with a NP relation. The sender has an NP instance and a witness while the receiver only gets the NP instance. In terms of properties, we require the following to hold:

- *Extractability*: An extractor, implemented as a quantum polynomial time algorithm, can extract a valid witness from an adversarial sender. We model the adversarial sender as a quantum polynomial time algorithm that follows the protocol but is allowed to choose its randomness; in the classical setting, this is termed as *semi-malicious* and we call this semi-malicious quantum adversaries[1].

  We also require *indistinguishability of extraction*: that is, the adversarial sender cannot distinguish whether it's interacting with the honest receiver or an extractor. In applications, this property is used to argue that the adversary cannot distinguish whether it's interacting with the honest party or the simulator.
- *Zero-Knowledge*: A malicious receiver should not be able to extract a valid witness after interacting with the sender. The malicious receiver can either be a classical probabilistic polynomial time algorithm or a quantum polynomial time algorithm. Correspondingly, there are two notions of quantum extraction protocols we study: quantum extraction protocols secure against quantum adversarial receivers (qQEXT) and quantum extraction protocols secure against classical adversarial receivers (cQEXT).

[1] In the literature, this type of semi-malicious adversaries are also referred to as *explainable* adveraries.

There are two reasons why we only study extraction against semi-malicious adversaries, instead of malicious adversaries (who can arbitrarily deviate from the protocol): first, even extracting from semi-malicious adversaries turns out to be challenging and we view this as a first step towards extraction from malicious adversaries and second, in the classical setting, there are works that show how to leverage extraction from semi-malicious adversaries to achieve zero-knowledge protocols [9,11] or secure two-party computation protocols [4].

Quantum extraction protocols are interesting even if we only consider classical adversaries, as they present a new method for proving zero-knowledge. For instance, to demonstrate zero-knowledge, we need to demonstrate a simulator that has a computational capability that a malicious prover doesn't have. Allowing quantum simulators in the classical setting [28] is another way to achieve this asymmetry between the power of the simulator and the adversary besides the few mentioned before (rewinding, superpolynomial, or non-black-box). Furthermore, quantum simulators capture the notion of knowledge that could be learnt if a malicious verifier had access to a quantum computer.

Quantum-Lasting Security. A potential concern regarding the security of cQEXT protocols is that the classical malicious receiver participating in the cQEXT protocol could later, long after the protocol has been executed, use a quantum computer to learn the witness of the sender from the transcript of the protocol and its own private state. For instance, the transcript could contain an ElGamal encryption of the witness of the sender; while a malicious classical receiver cannot break it, after the protocol is completed, it could later use a quantum computer to learn the witness. This is especially interesting in the event (full-fledged) quantum computers might become available in the future. First introduced by Unruh [39], we study the concept of quantum-lasting security; any quantum polynomial time (QPT) adversary given the transcript and the private state of the malicious receiver, should not be able to learn the witness of the sender. Our construction will satisfy this security notion and thus our protocol is resilient against the possibility of quantum computers being accessible in the future.

Result #1: Constant Round qQEXT protocols. We show the following result.

Theorem 1 (Informal). *Assuming quantum hardness of learning with errors and a quantum fully homomorphic encryption scheme (for arbitrary poly-time computations)*[2]*, satisfying, (1) perfect correctness for classical messages and, (2) ciphertexts of poly-sized classical messages have a poly-sized classical description, there exists a constant round quantum extraction protocol secure against quantum poly-time receivers.*

We clarify what we mean by perfect correctness. For every public key, every valid fresh ciphertext of a classical message can always be decrypted correctly. Moreover, we require that for every valid fresh ciphertext, of a classical message, the evaluated ciphertext can be decrypted correctly with probability negligibly

[2] As against leveled quantum FHE, which can be based on quantum hardness of LWE.

close to 1. We note that the works of [14,31] give candidates for quantum fully homomorphic encryption schemes satisfying both the above properties.

En route to proving the above theorem, we introduce a new non black extraction technique in the quantum setting building upon a *classical* non-black extraction technique of [11]. We view identifying the appropriate classical non-black-box technique to also be a contribution of our work. A priori it should not be clear whether classical non-black-box techniques are useful in constructing their quantum analogues. For instance, it is unclear how to utilize the well known non-black-box technique of Barak [7]; at a high level, the idea of Barak [7] is to commit to the code of the verifier and then prove using a succinct argument system that either the instance is in the language or it has the code of the verifier. In our setting, the verifier is a quantum circuit which means that we would require succinct arguments for quantum computations which we currently don't know how to achieve.

Non-black-box extraction overcomes the disadvantage quantum rewinding poses in achieving constant round extraction; the quantum rewinding employed by [42] requires polynomially many rounds (due to sequential repetition) or constant rounds with non-negligible gap between extraction and verification error [38].

This technique was concurrently developed by Bitansky and Shmueli [12] (see "Comparison with [12]" paragraph) and they critically relied upon this to construct a constant-round zero-knowledge argument system for NP and QMA, thus resolving a long-standing open problem in the round complexity of quantum zero-knowledge.

Subsequent Work. Many followup works have used the non-black-box extraction technique we introduce in this work to resolve other open problems in quantum cryptography. For instance, our technique was adopted to prove that quantum copy-protection is impossible [6]; resolving a problem that was open for more than a decade. It was also used to prove that quantum VBB for classical circuits is impossible [2,6]. In yet another exciting follow up work, this technique was developed further to achieve the first constant round post-quantum secure MPC protocol [1].

Result #2: Constant Round cQEXT protocols. We also present a construction of quantum extraction protocols secure against classical adversaries (cQEXT). This result is incomparable to the above result; on one hand, it is a weaker setting but on the other hand, the security of this construction can solely be based on the hardness of learning with errors.

Theorem 2 (Informal). *Assuming quantum hardness of learning with errors, there exists a constant round quantum extraction protocol secure against classical PPT adversaries and satisfying quantum-lasting security.*

Our main insight is to turn the "test of quantumness" protocol introduced in [15] into a quantum extraction protocol using cryptographic tools. In fact, our techniques are general enough that they might be useful to turn any protocol that can verify a quantum computer versus a classical computer into a quantum extraction protocol secure against classical adversaries; the transformation additionally assumes quantum hardness of learning with errors. Our work presents a new avenue for using "test of quantumness" protocols beyond using them just to test whether the server is quantum or not.

We note that it is conceivable to construct "test of quantumness" protocols from DDH (or any other quantum-**in**secure assumption). The security of the resulting extraction protocol would then be based on DDH and quantum hardness of learning with errors – the latter needed to argue quantum-lasting security. However, the security of our protocol is solely based on the quantum hardness of learning with errors.

Result #3: Constant Round QZK for NP with Classical Soundness. As an application, we show how to construct constant quantum zero-knowledge argument systems secure against quantum verifiers based on quantum hardness of learning with errors; however, the soundness is still against classical PPT adversaries.

Moreover, our protocol satisfies zero-knowledge against quantum verifiers with arbitrary quantum auxiliary state. Such protocols are also called auxiliary-input zero-knowledge protocols [24] and are necessary for composition. Specifically, our ZK protocol can be composed with other protocols to yield new protocols satisfying quantum security.

Theorem 3 (Constant Round Quantum ZK with Classical Soundness; Informal). *Assuming quantum hardness of learning with errors, there exists a constant round black box quantum zero-knowledge system with negligible soundness against classical PPT algorithms. Moreover, our protocol satisfies (quantum) auxiliary-input zero-knowledge property.*

A desirable property from a QZK protocol is if the verifier is classical then the simulator is also classical. While our protocol doesn't immediately satisfy this property, we show, nonetheless, that there is a simple transformation that converts into another QZK protocol that has this desirable property.

Application: Authorization with Quantum Cloud. Suppose Eva wants to convince the cloud services offered by some company that she has the authorization to access a document residing in the cloud. Since the authorization information could leak sensitive information about Eva, she would rather use a zero-knowlede protocol to prove to the cloud that she has the appropriate authorization. While we currently don't have scalable implementations of quantum computers, this could change in the future when organizations (e.g. governments, IBM, Microsoft, etc.) could be the first ones to develop a quantum computer. They could in principle then use this to break the zero-knowledge property of Eva's protocol and learn sensitive information about her. In this case, it suffices to use a QZK protocol but only requiring soundness against malicious classical users; in the nearby future, it is reasonable to assume that even if organizations with enough resources get to develop full-fledged quantum computers, it'll take a while before everyday users will have access to one.

1.2 Related Work

Quantum Rewinding. Watrous [42] introduced the quantum analogue of the rewinding technique. Later, Unruh [38] introduced yet another notion of quantum rewinding with the purpose of constructing quantum zero-knowledge proofs of knowledge. Unruh's rewinding does have extractability, but it requires that the underlying protocol to satisfy *strict soundness*. Furthermore, the probability that the extractor succeeds is not negligibly close to 1. The work of [3] shows that relative to an oracle, many classical zero-knowledge protocols are quantum insecure, and that the strict soundness condition from [38] is necessary in order for a sigma protocol to be a quantum proofs of knowledge.

Quantum and Classical Zero-Knowledge. Zero-knowledge against quantum adversaries was first studied by Watrous [42]. He showed how the GMW protocol [23] for graph 3-colorability is still zero-knowledge against quantum verifiers. Other works [18,26,27,29,33,38] have extended the study of classical protocols that are quantum zero-knowledge, and more recently, Broadbent et al. [17] extended the notion of zero-knowledge to QMA languages. By using ideas from [32] to classically verify quantum computation, the protocol in [17] was adapted to obtained classical argument systems for quantum computation in [41]. All known protocols, with non-negligible soundness error, take non-constant rounds.

On the other hand, zero knowledge proof and argument systems have been extensively studied in classical cryptography. In particular, a series of recent works [8–11] resolved the round complexity of zero knowledge argument systems.

Comparison with [12]. In a recent exciting work, [12] construct a constant round QZK with soundness against quantum adversaries for NP and QMA.

- The non-black-box techniques used in their work was concurrently developed and are similar to the techniques used in our QEXT protocol secure against quantum receivers[3].
- Subsequent to their posting, using completely different techniques, we developed QEXT secure against classical receivers and used it to build a constant round QZK system with classical soundness. There are a few crucial differences between our QZK argument system and theirs:
  1. Our result is based on quantum hardness of learning with errors while their result is based on the existence of quantum fully homomorphic encryption for arbitrary polynomial computations and quantum hardness of learning with errors.
  2. The soundness of their argument system is against quantum polynomial time algorithms while ours is only against classical PPT adversaries.

[3] A copy of our QEXT protocol secure against quantum receivers was privately communicated to the authors of [12] on the day of their public posting and our paper was posted online in about two weeks from then [5].

1.3 Quantum Extraction with Security Against Classical Receivers: Overview

We start with the overview of quantum extraction protocols with security against classical receivers.

Starting Point: Noisy Trapdoor Claw-Free Functions. Our main idea is to turn the "test of quantumness" from [15] into an extraction protocol. Our starting point is a noisy trapdoor claw-free function (NTCF) family [15,31,32], parameterized by key space $\mathcal{K}$, input domain $\mathcal{X}$ and output domain $\mathcal{Y}$. Using a key $\mathbf{k} \in \mathcal{K}$, NTCFs allows for computing the functions, denoted by $f_{\mathbf{k},0}(x) \in \mathcal{Y}$ and $f_{\mathbf{k},1}(x) \in \mathcal{Y}$ [4], where $x \in \mathcal{X}$. Using a trapdoor td associated with a key $\mathbf{k}$, any y in the support of $f_{\mathbf{k},b}(x)$, can be efficiently inverted to obtain x. Moreover, there are "claw" pairs (x_0, x_1) such that $f_{\mathbf{k},0}(x_0) = f_{\mathbf{k},1}(x_1)$. Roughly speaking, the security property states that it is computationally hard even for a quantum computer to simultaneously produce $y \in \mathcal{Y}$, values (b, x_b) and (d, u) such that $f_{\mathbf{k},b}(x_b) = y$ and $\langle d, J(x_0) \oplus J(x_1) \rangle = u$, where $J(\cdot)$ is an efficienctly computable injective function mapping $\mathcal{X}$ into bit strings. What makes this primitive interesting is its quantum capability that we will discuss when we recall below the test of [15].

Test of Quantumness [15]. Using NTCFs, [15] devised the following test[5]:

- The classical client, who wants to test whether the server it's interacting with is quantum or classical, first generates a key $\mathbf{k}$ along with a trapdoor td associated with a noisy trapdoor claw-free function (NTCF) family. It sends $\mathbf{k}$ to the server.
- The server responds back with $y \in \mathcal{Y}$.
- The classical client then sends a **challenge** bit $\mathbf{a}$ to the server.
- If $\mathbf{a} = 0$, the server sends a pre-image x_b along with bit b such that $f_{\mathbf{k},b}(x_b) = y$. If $\mathbf{a} = 1$, the server sends a vector d along with a bit u satisfying the condition $\langle d, J(x_0) \oplus J(x_1) \rangle = u$, where x_0, x_1 are such that $f_{\mathbf{k},0}(x_0) = f_{\mathbf{k},1}(x_1) = y$.

The client can check if the message sent by the server is either a valid pre-image or a valid d that is correlated with respect to both the pre-images.

Intuitively, since the (classical) server does not know, at the point when it sends y, whether it will be queried for (b, x_b) or (d, u), by the security of NTCFs, it can only answer one of the queries. While the quantum capability of NTCFs allows for a quantum server to maintain a superposition of a claw at the time it sent y and depending on the query made by the verifier it can then perform the appropriate quantum operations to answer the client; thus it will always pass the test.

[4] The efficient implementation of f only approximately computes f and we denote this by f'. We ignore this detail for now.

[5] As written, this test doesn't have negligible soundness but we can achieve negligible soundness by parallel repetition.

From Test of Quantumness to Extraction. A natural attempt to achieve extraction is the following: the sender takes the role of the client and the receiver takes the role of the server and if the test passes, the sender sends the witness to the receiver. We sketch this attempt below.

- Sender on input instance-witness pair $(\mathbf{y}, \mathbf{w})$ and receiver on input instance $\mathbf{y}$ run a "test of quantumness" protocol where the receiver (taking the role of the server) needs to convince the sender (taking the role of the classical client) that it is a quantum computer.
- If the receiver succeeds in the "test of quantumness" protocol then the sender $\mathbf{w}$, else it aborts.

Note that a quantum extractor can indeed succeed in the test of quantumness protocol and hence, it would receive $\mathbf{w}$ while a malicious classical adversary will not.

However, the above solution is not good enough for us. It does not satisfy indistinguishability of extraction: the sender can detect whether it's interacting with a quantum extractor or an honest receiver.

Achieving Indistinguishability of Extraction. To ensure indistinguishability of extraction, we rely upon a tool called secure function evaluation [9,21] that satisfies quantum security. A secure function evaluation (SFE) allows for two parties P_1 and P_2 to securely compute a function on their inputs in a such a way that only one of the parties, say P_2, receives the output of the function. In terms of security, we require that: (i) P_2 doesn't get information about P_1's input beyond the output of the function and, (ii) P_1 doesn't get any information about P_2's input (in fact, even the output of the protocol is hidden from P_1).

The hope is that by combining SFE and test of quantumness protocol, we can guarantee that a quantum extractor can still recover the witness by passing the test of quantumness as before but the sender doesn't even know whether the receiver passed or not. To implement this, we assume a structural property from the underlying test of quantumness protocol: until the final message of the protocol, the client cannot distinguish whether it's talking to a quantum server or a classical server. This structural property is satisfied by the test of quantumness protocol [15] sketched above.

Using this structural property and SFE, here is another attempt to construct a quantum extraction protocol: let the test of quantumness protocol be a k-round protocol.

- Sender on input instance-witness pair $(\mathbf{y}, \mathbf{w})$ and receiver on input instance $\mathbf{y}$ run the first $(k-1)$ rounds of the test of quantumness protocol where the receiver (taking the role of the server) needs to convince the sender (taking the role of the receiver) that it can perform quantum computations.

- Sender and receiver then run a SFE protocol for the following functionality G: it takes as input $\mathbf{w}$ and the first $(k-1)$ rounds of the test of quantumness protocol from the sender, the k^{th} round message from the receiver[6] and outputs $\mathbf{w}$ if indeed the test passed, otherwise output $\perp$. Sender will take the role of P_1 and the receiver will take the role of P_2 and thus, only the receiver will receive the output of G.

Note that the security of SFE guarantees that the output of the protocol is hidden from the sender and moreover, the first $(k-1)$ messages of the test of quantumness protocol doesn't reveal the information about whether the receiver is a quantum computer or not. These two properties ensure the sender doesn't know whether the receiver passed the test or not. Furthermore, the quantum extractor still succeeds in extracting the witness $\mathbf{w}$ since it passes the test.

The only remaining property to prove is zero-knowledge.

Challenges in Proving Zero-Knowledge. How do we ensure that a malicious classical receiver was not able to extract the witness? The hope would be to invoke the soundness of the test of quantumness protocol to argue this. However, to do this, we need all the k messages of the test of quantumness protocol.

To understand this better, let us recall how the soundness of the test of quantumness works: the client sends a challenge bit $\mathbf{a} = 0$ to the server who responds back with (b, x_b), then the client rewinds the server and instead sends the challenge bit $\mathbf{a} = 1$ and it receives (d, u): this contradicts the security of NTCFs since a classical PPT adversary cannot simultaneously produce both a valid pre-image (b, x_b) and a valid correlation vector along with the prediction bit (d, u).

Since the last message is fed into the secure function evaluation protocol and inaccessible to the simulator, we cannot use this rewinding strategy to prove the zero-knowledge of the extraction protocol.

Final Template: Zero-Knowledge via Extractable Commitments [36,37]. To overcome this barrier, we force the receiver to commit, using an extractable commitment scheme, to the k^{th} round of the test of quantumness protocol before the SFE protocol begins. An extractable commitment scheme is one where there is an extractor who can extract an input x being committed from the party committing to x. Armed with this tool, we give an overview of our construction below.

- Sender on input instance-witness pair $(\mathbf{y}, \mathbf{w})$ and receiver on input instance $\mathbf{y}$ run the first $(k-1)$ rounds of the test of quantumness protocol where the receiver (taking the role of the server) needs to convince the sender (taking the role of the receiver) that it can perform quantum computations.

[6] It follows without loss of generality that the server (and thus, the receiver of the quantum extraction protocol) computes the final message of the test of quantumness protocol.

- The k^{th} round of the test of quantumness protocol is then committed by the receiver, call it $\mathbf{c}$, using the extractable commitment scheme[7].
- Finally, the sender and the receiver then run a SFE protocol for the following functionality G: it takes as input $\mathbf{w}$ and the first $(k-1)$ rounds of the test of quantumness protocol from the sender, the decommitment of $\mathbf{c}$ from the receiver and outputs $\mathbf{w}$ if indeed the test passed, otherwise output $\perp$. Sender will take the role of P_1 and the receiver will take the role of P_2 and thus, only the receiver will receive the output of G.

Let us remark about zero-knowledge since we have already touched upon the other properties earlier. To argue zero-knowledge, construct a simulator that interacts honestly with the malicious receiver until the point the extraction protocol is run. Then, the simulator runs the extractor of the commitment scheme to extract the final message of the test of quantumness protocol. It then rewinds the test of quantumness protocol to the point where the simulator sends a different challenge bit (see the informal description of [15] given before) and then runs the extractor of the commitment scheme once again to extract the k^{th} round message of the test of quantumness protocol. Recall that having final round messages corresponding to two different challenge bits is sufficient to break the security of NTCFs; the zero-knowledge property then follows.

A couple of remarks about our simulator. Firstly, the reason why our simulator is able to rewind the adversary is because the adversary is a classical PPT algorithm. Secondly, our simulator performs *double rewinding* – not only does the extractor of the commitment scheme perform rewinding but also the test of quantumness protocol is rewound.

1.4 Constant Round QZK Argument Systems with Classical Soundness

We show how to use the above quantum extraction protocol secure against classical receivers (cQEXT) to construct an interactive argument system satisfying classical soundness and quantum ZK.

From Quantum Extraction to Quantum Zero-Knowledge. As a starting point, we consider the quantum analogue of the seminal FLS technique [19] to transform a quantum extraction protocol into a quantum ZK protocol. A first attempt to construct quantum ZK is as follows: let the input to the prover be instance $\mathbf{y}$ and witness $\mathbf{w}$ while the input to the verifier is $\mathbf{y}$.

- The verifier commits to some trapdoor td. Call the commitment $\mathbf{c}$ and the corresponding decommitment $\mathbf{d}$.
- The prover and verifier then execute a quantum extraction protocol with the verifier playing the role of the sender, on input $(\mathbf{c}, \mathbf{d})$, while the prover plays the role of the receiver on input $\mathbf{c}$.

[7] In the technical sections, we use a specific construction of extractable commitment scheme by [36,37] since we additionally require security against quantum adversaries.

- The prover and the verifier then run a witness-indistinguishable protocol where the prover convinces the verifier that either $\mathbf{y}$ belongs to the language or it knows td.

At first sight, it might seem that the above template should already give us the result we want; unfortunately, the above template is insufficient. The verifier could behave maliciously in the quantum extraction protocol but the quantum extraction protocol only guarantees security against semi-malicious senders. Hence, we need an additional mechanism to protect against malicious receivers. Of course, we require witness-indistinguishability to hold against quantum verifiers and we do know candidates satisfying this assuming quantum hardness of learning with errors [13,30].

Handling Malicious Behavior in QEXT. To check that the verifier behaved honestly in the quantum extraction protocol, we ask the verifier to reveal the inputs and random coins used in the quantum extraction protocol. At this point, the prover can check if the verifier behaved honestly or not. Of course, this would then violate soundness: the malicious prover upon receiving the random coins from the verifier can then recover td and then use this to falsely convince the verifier to accept its proof. We overcome this by forcing the prover to commit (we again use the extractable commitment scheme of [36]) to some string td' just before the verifier reveals the inputs and random coins used in the quantum extraction protocol. Then we force the prover to use the committed td' in the witness-indistinguishable protocol; the prover does not gain any advantage upon seeing the coins of the verifier and thus, ensuring soundness.

One aspect we didn't address so far is the aborting issue of the verifier: if the verifier aborts in the quantum extraction protocol, the simulator still needs to produce a transcript indistinguishable from that of the honest prover. Luckily for us, the quantum extraction protocol we constructed before already allows for simulatability of aborting adversaries.

To summarise, our ZK protocol consists of the following steps: (i) first, the prover and the verifier run the quantum extraction protocol, (ii) next the prover commits to a string td' using [36], (iii) the verifier then reveals the random coins used in the extraction protocol and, (iv) finally, the prover and the verifier run a quantum WI protocol where the prover convinces the verifier that it either knows a trapdoor td' or that $\mathbf{y}$ is a YES instance.

1.5 Quantum Extraction with Security Against Quantum Receivers: Overview

We show how to construct extraction protocols where we prove security against quantum receivers. At first sight, it might seem that quantum extraction and quantum zero-knowledge properties are contradictory since the extractor has the same computational resources as the malicious receiver. However, we provide more power to the extractor by giving the extractor non-black-box access to the semi-malicious sender. There is a rich literature on non-black-box techniques in the classical setting starting with the work of [7].

Quantum Extraction via Circular **In***security of QFHE.* The main tool we employ in our protocol is a fully homomorphic encryption qFHE scheme[8] that allows for public homomorphic evaluation of quantum circuits. Typically, we require a fully homomorphic encryption scheme to satisfy semantic security. However, for the current discussion, we require that qFHE to satisfy a stronger security property called 2-circular **in**security:

> Given $\mathsf{qFHE.Enc}(\mathsf{PK}_1, SK_2)$ (i.e., encryption of SK_2 under PK_1), $\mathsf{qFHE.Enc}(PK_2, \mathsf{SK}_1)$, where $(\mathsf{PK}_1, \mathsf{SK}_1)$ and (PK_2, SK_2) are independently generated public key-secret key pairs, we can efficiently recover SK_1 and SK_2.

Later, we show how to get rid of 2-circular **in**security property by using lockable obfuscation [25,43]. Here is our first attempt to construct the extraction protocol:

- The sender, on input instance $\mathbf{y}$ and witness $\mathbf{w}$, sends three ciphertexts: $\mathsf{CT}_1 \leftarrow \mathsf{qFHE.Enc}(\mathsf{PK}_1, \mathsf{td})$, $\mathsf{CT}_2 \leftarrow \mathsf{qFHE.Enc}(\mathsf{PK}_1, \mathbf{w})$ and $\mathsf{CT}_3 \leftarrow \mathsf{qFHE.Enc}(PK_2, \mathsf{SK}_1)$.
- The receiver sends td'.
- If $\mathsf{td}' = \mathsf{td}$ then the sender sends SK_2.

A quantum extractor with non-black-box access to the private (quantum) state of the semi-malicious sender S does the following:

- It first encrypts the private (quantum) state of S under public key PK_1.
- Here is our main insight: the extractor can homomorphically evaluate the next message function of S on CT_1 and the encrypted state of S. The result is $\mathsf{CT}_1^* = \mathsf{qFHE.Enc}(\mathsf{PK}_1, S(\mathsf{td}))$. But note that $S(\mathsf{td})$ is nothing but SK_2; note that S upon receiving $\mathsf{td}' = \mathsf{td}$ outputs SK_2. Thus, we have $\mathsf{CT}_1^* = \mathsf{qFHE.Enc}(\mathsf{PK}_1, SK_2)$.
- Now, the extractor has both $\mathsf{CT}_3 = \mathsf{qFHE.Enc}(PK_2, \mathsf{SK}_1)$ and $\mathsf{CT}_1^* = \mathsf{qFHE.Enc}(\mathsf{PK}_1, SK_2)$. It can then use the circular **in**security of qFHE to recover SK_1, SK_2.
- Finally, it decrypts CT_2 to obtain the witness $\mathbf{w}$!

The correctness of extraction alone is not sufficient; we need to argue that the sender cannot distinguish whether it's interacting with the honest receiver or the extractor. This is not true in our protocol since the extractor will always compute the next message function of S on $\mathsf{td}' = \mathsf{td}$ whereas an honest receiver will send $\mathsf{td}' = \mathsf{td}$ only with negligible probability.

Indistinguishability of Extraction: SFE Strikes Again. We already encountered a similar issue when we were designing extraction protocols with security against classical receivers and the tool we used to solve that issue was secure function evaluation (SFE); we will use the same tool here as well.

Using SFE, we make another attempt at designing the quantum extraction protocol.

[8] Recall that a classical FHE scheme [16,20] allows for publicly evaluating an encryption of a message x using a circuit C to obtain an encryption of $C(x)$.

- The sender, on input instance $\mathbf{y}$ and witness $\mathbf{w}$, sends three ciphertexts: $\mathsf{CT}_1 \leftarrow \mathsf{qFHE.Enc}(\mathsf{PK}_1, \mathsf{td})$, $\mathsf{CT}_2 \leftarrow \mathsf{qFHE.Enc}(\mathsf{PK}_1, \mathbf{w})$ and $\mathsf{CT}_3 \leftarrow \mathsf{qFHE.Enc}(PK_2, \mathsf{SK}_1)$.
- The sender and the receiver executes a secure two-party computation protocol, where the receiver feeds td' and the sender feeds in $(\mathsf{td}, \mathbf{w})$. After the protocol finishes, the receiver recovers $\mathbf{w}$ if $\mathsf{td}' = \mathsf{td}$, else it recovers $\perp$. The sender doesn't receive any output.

The above template guarantees indistinguishability of extraction property[9].

We next focus on zero-knowledge. To do this, we need to argue that the td' input by the malicious receiver can never be equal to td. One might falsely conclude that the semantic security of qFHE would imply that td is hidden from the sender and hence the argument follows. This is not necessarily true; the malicious receiver might be able to "maul" the ciphertext CT_1 into the messages of the secure function evaluation protocol in such a way that the implicit input committed by the receiver is td'. We need to devise a mechanism to prevent against such mauling attacks.

Preventing Mauling Attacks. We prevent the mauling attacks by forcing the receiver to commit to random strings $(r_1, \ldots, r_\ell)$ in the first round, where $|\mathsf{td}| = \ell$, even before it receives the ciphertexts $(\mathsf{CT}_1, \mathsf{CT}_2, \mathsf{CT}_3)$ from the sender. Once it receives the ciphertexts, the receiver is supposed to commit to every bit of the trapdoor using the randomness $r_1, \ldots, r_\ell$; that is, the i^{th} bit of td is committed using r_i.

Using this mechanism, we can then provably show that if the receiver was able to successfully maul the qFHE ciphertext then it violates the semantic security of qFHE using a non-uniform adversary.

Replacing Circular **In***security with Lockable Obfuscation* [25,43]. While the above protocol is a candidate for quantum extraction protocol secure against quantum receivers; it is still unsatisfactory since we assume a quantum FHE scheme satisfying 2-circular **in**security. We show how to replace 2-circular insecure QFHE with *any* QFHE scheme (satisfying some mild properties already satisfied by existing candidates) and lockable obfuscation for classical circuits. A lockable obfuscation scheme is an obfuscation scheme for a specific class of functionalities called compute-and-compare functionalities; a compute-and-compare functionality is parameterized by C, α (lock), β such that on input x, it outputs β if $C(x) = \alpha$. As long as α is sampled uniformly at random and independently of C, lockable obfuscation completely hides the circuit C, α and β. The idea to replace 2-circular insecure QFHE with lockable obfuscation[10] is as follows:

[9] There is a subtle point here that we didn't address: the transcript generated by the extractor is encrypted under qFHE. But after recovering the secret keys, the extractor could decrypt the encrypted transcript.

[10] It shouldn't be too surprising that lockable obfuscation can be used to replace circular insecurity since one of the applications [25,43] of lockable obfuscation was to demonstrate counter-examples for circular security.

obfuscate the circuit, with secret key SK_2, ciphertext $\mathsf{qFHE.Enc}(SK_2, r)$ hardwired, that takes as input $\mathsf{qFHE.Enc}(\mathsf{PK}_1, SK_2)$, decrypts it to obtain SK_2', then decrypts $\mathsf{qFHE.Enc}(SK_2, r)$ to obtain r' and outputs SK_1 if $r' = r$. If the adversary does not obtain $\mathsf{qFHE.Enc}(\mathsf{PK}_1, SK_2)$ then we can first invoke the security of lockable obfuscation to remove SK_1 from the obfuscated circuit and then it can replace $\mathsf{qFHE.Enc}(\mathsf{PK}_1, \mathbf{w})$ with $\mathsf{qFHE.Enc}(\mathsf{PK}_1, \perp)$. The idea of using fully homomorphic encryption along with lockable obfuscation to achieve non-black-box extraction was first introduced, in the classical setting, by [11].

Unlike our cQEXT construction, the non-black-box technique used for qQEXT does not directly give us a constant round quantum zero-knowledge protocol for NP. This is because an adversarial verifier that aborts can distinguish between the extractor or the honest prover (receiver in qQEXT). The main issue is that the extractor runs the verifier homomorphically, so it cannot detect if the verifier aborted at any point in the protocol without decrypting. But if the verifier aborted, the extractor wouldn't be able to decrypt in the first place – it could attempt to rewind but then this would destroy the initial quantum auxiliary state.

2 Preliminaries

We denote the security parameter by λ. We denote (classical) computational indistiguishability of two distributions $\mathcal{D}_0$ and $\mathcal{D}_1$ by $\mathcal{D}_0 \approx_{c,\varepsilon} \mathcal{D}_1$. In the case when ε is negligible, we drop ε from this notation.

Languages and Relations. A language $\mathcal{L}$ is a subset of $\{0,1\}^*$. A relation $\mathcal{R}$ is a subset of $\{0,1\}^* \times \{0,1\}^*$. We use the following notation:

- Suppose $\mathcal{R}$ is a relation. We define $\mathcal{R}$ to be *efficiently decidable* if there exists an algorithm A and fixed polynomial p such that $(x, w) \in \mathcal{R}$ if and only if $A(x, w) = 1$ and the running time of A is upper bounded by $p(|x|, |w|)$.
- Suppose $\mathcal{R}$ is an efficiently decidable relation. We say that $\mathcal{R}$ is a NP relation if $\mathcal{L}(\mathcal{R})$ is a NP language, where $\mathcal{L}(\mathcal{R})$ is defined as follows: $x \in \mathcal{L}(R)$ if and only if there exists w such that $(x, w) \in \mathcal{R}$ and $|w| \leq p(|x|)$ for some fixed polynomial p.

2.1 Learning with Errors

In this work, we are interested in the decisional learning with errors (LWE) problem. This problem, parameterized by n, m, q, χ, where $n, m, q \in \mathbb{N}$, and for a distribution χ supported over $\mathbb{Z}$ is to distinguish between the distributions $(\mathbf{A}, \mathbf{A}\mathbf{s}{+}\mathbf{e})$ and $(\mathbf{A}, \mathbf{u})$, where $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{m\times n}, \mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^{n\times 1}, \mathbf{e} \xleftarrow{\$} \chi^{m\times 1}$ and $\mathbf{u} \leftarrow \mathbb{Z}_q^{m\times 1}$. Typical setting of m is $n\log(q)$, but we also consider $m = \mathrm{poly}(n\log(q))$.

We base the security of our constructions on the quantum hardness of learning with errors problem.

2.2 Notation and General Definitions

For completeness, we present some of the basic quantum definitions, for more details see [34].

Quantum States and Channels. Let $\mathcal{H}$ be any finite Hilbert space, and let $L(\mathcal{H}) := \{\mathcal{E} : \mathcal{H} \rightarrow \mathcal{H}\}$ be the set of all linear operators from $\mathcal{H}$ to itself (or endomorphism). Quantum states over $\mathcal{H}$ are the positive semidefinite operators in $L(\mathcal{H})$ that have unit trace. Quantum channels or quantum operations acting on quantum states over $\mathcal{H}$ are completely positive trace preserving (CPTP) linear maps from $L(\mathcal{H})$ to $L(\mathcal{H}')$ where $\mathcal{H}'$ is any other finite dimensional Hilbert space.

A state over $\mathcal{H} = \mathbb{C}^2$ is called a qubit. For any $n \in \mathbb{N}$, we refer to the quantum states over $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$ as n-qubit quantum states. To perform a standard basis measurement on a qubit means projecting the qubit into $\{|0\rangle, |1\rangle\}$. A quantum register is a collection of qubits. A classical register is a quantum register that is only able to store qubits in the computational basis.

A unitary quantum circuit is a sequence of unitary operations (unitary gates) acting on a fixed number of qubits. Measurements in the standard basis can be performed at the end of the unitary circuit. A (general) quantum circuit is a unitary quantum circuit with 2 additional operations: (1) a gate that adds an ancilla qubit to the system, and (2) a gate that discards (trace-out) a qubit from the system. A quantum polynomial-time algorithm (QPT) is a uniform collection of quantum circuits $\{C_n\}_{n\in\mathbb{N}}$.

Quantum Computational Indistinguishability. When we talk about quantum distinguishers, we need the following definitions, which we take from [42].

Definition 1 (Indistinguishable collections of states). *Let I be an infinite subset $I \subset \{0,1\}^*$, let $p : \mathbb{N} \rightarrow \mathbb{N}$ be a polynomially bounded function, and let ρ_x and σ_x be $p(|x|)$-qubit states. We say that $\{\rho_x\}_{x\in I}$ and $\{\sigma_x\}_{x\in I}$ are* ***quantum computationally indistinguishable collections of quantum states*** *if for every QPT $\mathcal{E}$ that outputs a single bit, any polynomially bounded $q : \mathbb{N} \rightarrow \mathbb{N}$, and any auxiliary $q(|x|)$-qubits state ν, and for all $x \in I$, we have that*

$$|\Pr\left[\mathcal{E}(\rho_x \otimes \nu) = 1\right] - \Pr\left[\mathcal{E}(\sigma_x \otimes \nu) = 1\right]| \leq \epsilon(|x|)$$

for some negligible function $\epsilon : \mathbb{N} \rightarrow [0,1]$. We use the following notation

$$\rho_x \approx_{Q,\epsilon} \sigma_x$$

and we ignore the ϵ when it is understood that it is a negligible function.

Definition 2 (Indistinguishability of channels). *Let I be an infinite subset $I \subset \{0,1\}^*$, let $p, q : \mathbb{N} \rightarrow \mathbb{N}$ be polynomially bounded functions, and let $\mathcal{D}_x, \mathcal{F}_x$ be quantum channels mapping $p(|x|)$-qubit states to $q(|x|)$-qubit states. We say that $\{\mathcal{D}_x\}_{x\in I}$ and $\{\mathcal{F}_x\}_{x\in I}$ are* ***quantum computationally indistinguishable***

collection of channels *if for every QPT $\mathcal{E}$ that outputs a single bit, any polynomially bounded $t : \mathbb{N} \to \mathbb{N}$, any $p(|x|) + t(|x|)$-qubit quantum state ρ, and for all $x \in I$, we have that*

$$|\Pr[\mathcal{E}((\mathcal{D}_x \otimes \mathsf{Id})(\rho)) = 1] - \Pr[\mathcal{E}((\mathcal{F}_x \otimes \mathsf{Id})(\rho)) = 1]| \leq \epsilon(|x|)$$

for some negligible function $\epsilon : \mathbb{N} \to [0, 1]$. We will use the following notation

$$\mathcal{D}_x(\cdot) \approx_{Q,\epsilon} \mathcal{F}_x(\cdot)$$

and we ignore the ϵ when it is understood that it is a negligible function.

Interactive Models. We model an interactive protocol between a prover, Prover, and a verifier, Verifier, as follows. There are 2 registers $\mathrm{R}_{\mathsf{Prover}}$ and $\mathrm{R}_{\mathsf{Verifier}}$ corresponding to the prover's and the verifier's private registers, as well as a message register, R_{M}, which is used by both Prover and Verifier to send messages. In other words, both prover and verifier have access to the message register. We denote the size of a register R by $|\mathsf{R}|$ – this is the number of bits or qubits that the register can store. We will have 2 different notions of interactive computation. Our honest parties will perform classical protocols, but the adversaries will be allowed to perform quantum protocols with classical messages.

1. **Classical protocol:** An interactive protocol is classical if $\mathrm{R}_{\mathsf{Prover}}$, $\mathrm{R}_{\mathsf{Verifier}}$, and R_{M} are classical, and Prover and Verifier can only perform classical computation.
2. **Quantum protocol with classical messages:** An interactive protocol is quantum with classical messages if either one of $\mathrm{R}_{\mathsf{Prover}}$ or $\mathrm{R}_{\mathsf{Verifier}}$ is a quantum register, and R_{M} is classical. Prover and Verifier can perform quantum computations if their respective private register is quantum, but they can only send classical messages.

When a protocol has classical messages, we can assume that the adversarial party will also send classical messages. This is without loss of generality, because the honest party can enforce this condition by always measuring the message register in the computational basis before proceeding with its computations.

Non-Black-Box Access. Let S be a QPT party (e.g. either prover or verifier in the above descriptions) involved in specific quantum protocol. In particular, S can be seen as a collection of QPTs, $S = (S_1, ..., S_\ell)$, where ℓ is the number of rounds of the protocol, and S_i is the quantum operation that S performs on the ith round of the protocol.

We say that a QPT Q has *non-black-box access* to S, if Q has access to an efficient classical description for the operations that S performs in each round, $(S_1, ..., S_\ell)$, as well as access to the initial auxiliary inputs of S.

Interaction Channel. For a particular protocol $(\mathsf{Prover}, \mathsf{Verifier})$, the interaction between Prover and $\mathsf{Verifier}$ on input $\mathbf{y}$ induces a quantum channel $\mathcal{E}_{\mathbf{y}}$ acting on their private input states, ρ_{Prover} and $\sigma_{\mathsf{Verifier}}$. We denote the view of $\mathsf{Verifier}$ when interacting with Prover by

$$\mathsf{View}_{\mathsf{Verifier}}\left(\langle \mathsf{Prover}\left(\mathbf{y}, \rho_{\mathsf{Prover}}\right), \mathsf{Verifier}\left(\mathbf{y}, \sigma_{\mathsf{Verifier}}\right)\rangle\right),$$

and this view is defined as the verifiers output. Specifically,

$$\mathsf{View}_{\mathsf{Verifier}}\left(\langle \mathsf{Prover}\left(\mathbf{y}, \rho_{\mathsf{Prover}}\right), \mathsf{Verifier}\left(\mathbf{y}, \sigma_{\mathsf{Verifier}}\right)\rangle\right) := \mathsf{Tr}_{\mathrm{R}_{\mathsf{Prover}}}\left[\mathcal{E}_{\mathbf{y}}\left(\rho_{\mathsf{Prover}} \otimes \sigma_{\mathsf{Verifier}}\right)\right].$$

From the verifier's point of view, the interaction induces the channel $\mathcal{E}_{\mathbf{y},V}(\sigma) = \mathcal{E}_{\mathbf{y}}(\sigma \otimes \rho_{\mathsf{Prover}})$ on its private input state.

3 Secure Quantum Extraction Protocols

We define the notion of quantum extraction protocols below. An extraction protocol, associated with an NP relation, is a *classical* interactive protocol between a sender and a receiver. The sender has an NP instance and a witness; the receiver only has the NP instance.

In terms of properties, we require the property that there is a QPT extractor that can extract the witness from a semi-malicious sender (i.e., follows the protocol but is allowed to choose its own randomness) even if the sender is a QPT algorithm. Moreover, the semi-malicious sender should not be able to distinguish whether it's interacting with the extractor or the honest receiver.

In addition, we require the following property (zero-knowledge): the interaction of any malicious receiver with the sender should be simulatable without the knowledge of the witness. The malicious receiver can either be classical or quantum and thus, we have two notions of quantum extraction protocols corresponding to both of these cases.

In terms of properties required, this notion closely resembles the concept of zero-knowledge argument of knowledge (ZKAoK) systems. There are two important differences:

- Firstly, we do not impose any completeness requirement on our extraction protocol.
- In ZKAoK systems, the prover can behave maliciously (i.e., deviates from the protocol) and the argument of knowledge property states that the probability with which the extractor can extract is negligibly close to the probability with which the prover can convince the verifier. In our definition, there is no guarantee of extraction if the sender behaves maliciously.

Definition 3 (Quantum extraction protocols secure against quantum adversaries). *A* ***quantum extraction protocol secure against quantum adversaries****, denoted by* qQEXT *is a classical protocol between two classical PPT algorithms, sender* S *and a receiver* R *and is associated with an NP relation* $\mathcal{R}$*. The input to both the parties is an instance* $\mathbf{y} \in \mathcal{L}(\mathcal{R})$*. In addition, the sender also gets as input the witness* $\mathbf{w}$ *such that* $(\mathbf{y}, \mathbf{w}) \in \mathcal{R}$*. At the end of the protocol, the receiver gets the output* $\mathbf{w}'$*. The following properties are satisfied by* qQEXT*:*

- **Quantum Zero-Knowledge**: *Let $p : \mathbb{N} \rightarrow \mathbb{N}$ be any polynomially bounded function. For every $(\mathbf{y}, \mathbf{w}) \in \mathcal{R}$, for any QPT algorithm R^* with private quantum register of size $|\mathrm{R}_{\mathsf{R}^*}| = p(\lambda)$, for any large enough security parameter $\lambda \in \mathbb{N}$, there exists a QPT simulator Sim such that,*

$$\mathsf{View}_{\mathsf{R}^*}\left(\langle \mathsf{S}(1^\lambda, \mathbf{y}, \mathbf{w}), \mathsf{R}^*(1^\lambda, \mathbf{y}, \cdot)\rangle\right) \approx_Q \mathsf{Sim}(1^\lambda, \mathsf{R}^*, \mathbf{y}, \cdot).$$

- **Semi-Malicious Extractability**: *Let $p : \mathbb{N} \rightarrow \mathbb{N}$ be any polynomially bounded function. For any large enough security parameter $\lambda \in \mathbb{N}$, for every $(\mathbf{y}, \mathbf{w}) \in \mathcal{L}(\mathcal{R})$, for every semi-malicious*[11] *QPT S^* with private quantum register of size $|\mathrm{R}_{\mathsf{S}^*}| = p(\lambda)$, there exists a QPT extractor $\mathsf{Ext} = (\mathsf{Ext}_1, \mathsf{Ext}_2)$ (possibly using the code of S^* in a non-black box manner), the following holds:*
  - **Indistinguishability of Extraction**: $\mathsf{View}_{\mathsf{S}^*}\left(\langle \mathsf{S}^*(1^\lambda, \mathbf{y}, \mathbf{w}, \cdot), \mathsf{R}(1^\lambda, \mathbf{y})\rangle\right) \approx_Q \mathsf{Ext}_1\left(1^\lambda, \mathsf{S}^*, \mathbf{y}, \cdot\right)$
  - *The probability that Ext_2 outputs $\mathbf{w}'$ such that $(\mathbf{y}, \mathbf{w}') \in \mathcal{R}$ is negligibly close to 1.*

Definition 4 (Quantum extraction protocols secure against classical adversaries). *A* ***quantum extraction protocol secure against classical adversaries*** *cQEXT is defined the same way as in Definition 3 except that instead of quantum zero-knowledge, cQEXT satisfies classical zero-knowledge property defined below:*

- **Classical Zero-Knowledge**: *Let $p : \mathbb{N} \rightarrow \mathbb{N}$ be any polynomially bounded function. For any large enough security parameter $\lambda \in \mathbb{N}$, for every $(\mathbf{y}, \mathbf{w}) \in \mathcal{R}$, for any classical PPT algorithm R^* with auxiliary information $\mathsf{aux} \in \{0,1\}^{\mathrm{poly}(\lambda)}$, there exists a classical PPT simulator Sim such that*

$$\mathsf{View}_{\mathsf{R}^*}\left(\langle \mathsf{S}(1^\lambda, \mathbf{y}, \mathbf{w}), \mathsf{R}^*(1^\lambda, \mathbf{y}, \mathsf{aux})\rangle\right) \approx_c \mathsf{Sim}(1^\lambda, \mathsf{R}^*, \mathbf{y}, \mathsf{aux}).$$

Quantum-Lasting Security. A desirable property of cQEXT protocols is that a classical malicious receiver, long after the protocol has been executed cannot use a quantum computer to learn the witness of the sender from the transcript of the protocol along with its own private state. We call this property *quantum-lasting security*; first introduced by Unruh [39]. We formally define quantum-lasting security below.

Definition 5 (Quantum-Lasting Security). *A cQEXT protocol is said to be* **quantum-lasting secure** *if the following holds: for any large enough security parameter $\lambda \in \mathbb{N}$, for any classical PPT R^*, for any QPT adversary $\mathcal{A}^*$, for any auxiliary information $\mathsf{aux} \in \{0,1\}^{\mathrm{poly}(\lambda)}$, for any auxiliary state of polynomially many qubits, ρ, there exist a QPT simulator Sim^* such that:*

$$\mathcal{A}^*\left(\mathsf{View}_{\mathsf{R}^*}\left\langle \mathsf{S}(1^\lambda, \mathbf{y}, \mathbf{w}), \mathsf{R}^*(1^\lambda, \mathbf{y}, \mathsf{aux})\right\rangle, \rho\right) \approx_Q \mathsf{Sim}^*(1^\lambda, \mathbf{y}, \mathsf{aux}, \rho)$$

[11] A QPT algorithm is said to be semi-malicious in the quantum extraction protocol if it follows the protocol but is allowed to choose the randomness for the protocol.

4 QEXT Secure Against Classical Receivers

In this section, we show how to construct quantum extraction protocols secure against classical adversaries based solely on the quantum hardness of learning with errors.

Tools

- Quantum-secure computationally-hiding and perfectly-binding non-interactive commitments, Comm.

  We instantiate the underlying commitment scheme in [36] using Comm to obtain a quantum-secure extractable commitment scheme. Instead of presenting a definition of quantum-secure extractable commitment scheme and then instantiating it, we directly incorporate the construction of [36] in the construction of the extraction protocol.
- Noisy trapdoor claw-free functions $\{f_{\mathbf{k},b} : \mathcal{X} \rightarrow D_{\mathcal{Y}}\}_{\mathbf{k}\in\mathcal{K},b\in\{0,1\}}$.
- Quantum-secure secure function evaluation protocol $\mathsf{SFE} = (\mathsf{SFE.S}, \mathsf{SFE.R})$.

Construction. We present the construction of the quantum extraction protocol (S, R) in Fig. 2 for an NP language $\mathcal{L}$.

We prove the following lemma in the full version.

Lemma 1. *Assuming the quantum security of* $\mathsf{Comm}, \mathsf{SFE}$ *and NTCFs, the protocol* (S, R) *is a quantum extraction protocol secure against classical adversaries for NP. Moreover,* (S, R) *satisfies quantum-lasting security.*

5 Application: Classical ZK Arguments Secure Against Quantum Verifiers

In this section, we show how to construct a quantum zero-knowledge, classical prover, argument system for NP secure against quantum verifiers; that is, the protocol is classical, the malicious prover is also a classical adversary but the malicious verifier can be a polynomial time quantum algorithm. To formally define this notion, consider the following definition.

Definition 6 (Classical arguments for NP). *A classical interactive protocol* $(\mathsf{Prover}, \mathsf{Verifier})$ *is a* ***classical ZK argument system*** *for an NP language* $\mathcal{L}$*, associated with an NP relation* $\mathcal{L}(\mathcal{R})$*, if the following holds:*

- **Completeness***: For any* $(\mathbf{y}, \mathbf{w}) \in \mathcal{L}(\mathcal{R})$*, we have that* $\Pr[\langle \mathsf{Prover}(1^\lambda, \mathbf{y}, \mathbf{w}), \mathsf{Verifier}(1^\lambda, \mathbf{y})\rangle = 1] \geq 1 - \mathsf{negl}(\lambda)$*, for some negligible function* negl*.*
- **Soundness***: For any* $\mathbf{y} \notin \mathcal{L}$*, any classical PPT adversary* Prover^**, and any polynomial-sized auxiliary information* aux*, we have that* $\Pr[\langle \mathsf{Prover}^*(1^\lambda, \mathbf{y}, \mathsf{aux}), \mathsf{Verifier}(1^\lambda, \mathbf{y})\rangle = 1] \leq \mathsf{negl}(\lambda)$*, for some negligible function* negl*.*

F

Input of sender: $\left(\left\{\mathbf{c}_{i,0}^{(j)}, \mathbf{c}_{i,1}^{(j)}, (sh_{i,w_i}^{(j)})', (\mathbf{d}_{i,w_i}^{(j)})', \mathsf{td}_i, \mathbf{k}_i, y_i, v_i, w_i^{(j)}\right\}_{i,j\in[k]}, \mathbf{w}\right)$
Input of receiver: $\left(\left\{sh_{i,\overline{w_i}}^{(j)}, \mathbf{d}_{i,\overline{w_i}}^{(j)}\right\}_{i,j\in[k]}\right)$

- If for any $i, j \in [k]$, $\mathbf{c}_{i,w_i}^{(j)} \neq \mathsf{Comm}\left(1^\lambda, (sh_{i,w_i}^{(j)})'; (\mathbf{d}_{i,w_i}^{(j)})'\right)$ or $\mathbf{c}_{i,\overline{w_i}}^{(j)} \neq \mathsf{Comm}\left(1^\lambda, sh_{i,\overline{w_i}}^{(j)}; \mathbf{d}_{i,\overline{w_i}}^{(j)}\right)$, output $\perp$.
- For every $i \in [k]$, let $(x_{i,0}, x_{i,1}) \leftarrow \mathsf{Inv}(\mathbf{k}_i, \mathsf{td}_i, y_i)$.
  - *Check if the commitments commit to the same message*: Output $\perp$ if the following does not hold: for every $j, j' \in [k]$, we have $\left(sh_{i,w_i}^{(j)}\right)' \oplus sh_{i,w_i}^{(j)} = \left(sh_{i,w_i}^{(j')}\right)' \oplus sh_{i,w_i}^{(j')}$.
  - If $v_i = 0$: let $(b_i, J(x'_{i,b_i})) = (sh_{i,w_i}^{(j)})' \oplus sh_{i,\overline{w_i}}^{(j)}$, where $J(\cdot)$ is the injection in the definition of NTCF. Since $J(\cdot)$ can be efficiently inverted, recover x'_{i,b_i}. If $x'_{i,b_i} \neq x_{i,b_i}$, output $\perp$.
  - If $v_i = 1$: let $(u_i, d_i) = \left(sh_{i,w_i}^{(j)}\right)' \oplus sh_{i,\overline{w_i}}^{(j)}$. If $\langle d_i, J(x_{i,0}) \oplus J(x_{i,1})\rangle \neq u_i$, or if $d_i \notin G_{\mathbf{k}_i,0,x_{i,0}} \cap G_{\mathbf{k}_i,1,x_{i,1}}$ output $\perp$.
- Otherwise, output $\mathbf{w}$.

Fig. 1. Description of the function **F** associated with the SFE.

We say that a classical argument system for NP is a QZK (quantum zero-knowledge) classical argument system for NP if in addition to the above properties, a classical interactive protocol satisfies zero-knowledge against malicious receivers.

Definition 7 (QZK classical argument system for NP). *A classical interactive protocol* $(\mathsf{Prover}, \mathsf{Verifier})$ *is a **quantum zero-knowledge classical argument system** for a language $\mathcal{L}$, associated with an NP relation $\mathcal{L}(\mathcal{R})$ if both of the following hold.*

- $(\mathsf{Prover}, \mathsf{Verifier})$ *is a classical argument for $\mathcal{L}$ (Definition 6).*
- **Quantum Zero-Knowledge***: Let $p : \mathbb{N} \rightarrow \mathbb{N}$ be any polynomially bounded function. For any QPT* $\mathsf{Verifier}^*$ *that on instance $\mathbf{y} \in \mathcal{L}$ has private register of size $|\mathsf{R}_{\mathsf{Verifier}^*}| = p(|\mathbf{y}|)$, there exist a QPT* Sim *such that the following two collections of quantum channels are quantum computationally indistinguishable,*

Input of sender: $(\mathbf{y}, \mathbf{w})$.
Input of receiver: $\mathbf{y}$

- S: Compute $\forall i \in [k], (\mathbf{k}_i, \mathsf{td}_i) \leftarrow \mathsf{Gen}(1^\lambda; r_i)$, where $k = \lambda$. Send $\left(\{\mathbf{k}_i\}_{i \in [k]}\right)$.
- R: For every $i \in [k]$, choose a random bit $b_i \in \{0,1\}$ and sample a random $y_i \leftarrow f'_{\mathbf{k}_i, b_i}(x_{i,b_i})$, where $x_{i,b_i} \xleftarrow{\$} \mathcal{X}$. Send $\{y_i\}_{i \in [k]}$. (Recall that $f'_{\mathbf{k},b}(x)$ is a distribution over $\mathcal{Y}$.)
- S: Send bits $(v_1, \ldots, v_k)$, where $v_i \xleftarrow{\$} \{0,1\}$ for $i \in [k]$.
- R: For every $i, j \in [k]$, compute the commitments $\mathbf{c}_{i,0}^{(j)} \leftarrow \mathsf{Comm}(1^\lambda, sh_{i,0}^{(j)}; \mathbf{d}_{i,0}^{(j)})$ and $\mathbf{c}_{i,1}^{(j)} \leftarrow \mathsf{Comm}(1^\lambda, sh_{i,1}^{(j)}; \mathbf{d}_{i,1}^{(j)})$, where $sh_{i,0}^{(j)}, sh_{i,1}^{(j)} \xleftarrow{\$} \{0,1\}^{\mathrm{poly}(\lambda)}$ for $i, j \in [k]$. Send $\left(\left\{\mathbf{c}_{i,0}^{(j)}, \mathbf{c}_{i,1}^{(j)}\right\}_{i,j \in [k]}\right)$.
  Note: The reason why we have k^2 commitments above is because we repeat (in parallel) the test of quantumness protocol k times and for each repetition, the response of the receiver is committed using k commitments; the latter is due to [36].
- S: For every $i, j \in [k]$, send random bits $w_i^{(j)} \in \{0,1\}$.
- R: Send $\left(\left\{(sh_{i,w_i}^{(j)})', (\mathbf{d}_{i,w_i}^{(j)})'\right\}_{i,j \in [k]}\right)$.
- S and R run SFE, associated with the two-party functionality $\mathbf{F}$ defined in Figure 1; S takes the role of $\mathsf{SFE.S}$ and R takes the role of $\mathsf{SFE.R}$. The input to $\mathsf{SFE.S}$ is $\left(\left\{\mathbf{c}_{i,0}^{(j)}, \mathbf{c}_{i,1}^{(j)}, (sh_{i,w_i}^{(j)})', (\mathbf{d}_{i,w_i}^{(j)})', \mathsf{td}_i, \mathbf{k}_i, y_i, v_i, w_i^{(j)}\right\}_{i,j \in [k]}, \mathbf{w}\right)$ and the input to $\mathsf{SFE.R}$ is $\left(\left\{sh_{i,\overline{w_i}}^{(j)}, \mathbf{d}_{i,\overline{w_i}}^{(j)}\right\}_{i,j \in [k]}\right)$.

Fig. 2. Quantum extraction protocol (S, R) secure against classical receivers.

- $\{\mathsf{Sim}(\mathbf{y}, \mathsf{Verifier}^*, \cdot)\}_{\mathbf{y} \in \mathcal{L}}$
- $\{\mathsf{View}_{\mathsf{Verifier}^*}(\langle \mathsf{Prover}(\mathbf{y}, \mathsf{aux}_1), \mathsf{Verifier}^*(\mathbf{y}, \cdot)\rangle)\}_{\mathbf{y} \in \mathcal{L}}$.

In other words, that for every $\mathbf{y} \in \mathcal{L}$, for any bounded polynomial $q : \mathbb{N} \to \mathbb{N}$, for any QPT distinguisher $\mathcal{D}$ that outputs a single bit, and any $p(|\mathbf{y}|) + q(|\mathbf{y}|)$-qubits quantum state ρ,

$$\Big| \Pr\left[\mathcal{D}\left(\mathsf{Sim}(\mathbf{y}, \mathsf{Verifier}^*, \cdot) \otimes I\right)(\rho)) = 1\right]$$
$$- \Pr\left[\mathcal{D}\left((\mathsf{View}_{\mathsf{Verifier}^*}(\langle \mathsf{Prover}(\mathbf{y}, \mathsf{aux}_1), \mathsf{Verifier}^*(\mathbf{y}, \cdot)\rangle) \otimes I)(\rho)\right) = 1\right] \Big| \leq \epsilon(|\mathbf{y}|)$$

Witness-Indistinguishability Against Quantum Verifiers. As a building block, we also consider witness indistinguishable (WI) argument systems for NP languages secure against quantum verifiers. We define this formally below.

Definition 8 (Quantum WI for an $\mathcal{L} \in$ NP). *A classical protocol* $(\mathsf{Prover}, \mathsf{Verifier})$ *is a* ***quantum witness indistinguishable argument system*** *for an NP language* $\mathcal{L}$ *if both of the following hold.*

- $(\mathsf{Prover}, \mathsf{Verifier})$ *is a classical argument for* $\mathcal{L}$ *(Definition 6).*
- **Quantum WI***: Let* $p : \mathbb{N} \rightarrow \mathbb{N}$ *be any polynomially bounded function. For every* $\mathbf{y} \in \mathcal{L}$*, for any two valid witnesses* $\mathbf{w}_1$ *and* $\mathbf{w}_2$*, for any QPT* $\mathsf{Verifier}^*$ *that on instance* $\mathbf{y}$ *has private quantum register of size* $|\mathsf{R}_{\mathsf{Verifier}^*}| = p(|\mathbf{y}|)$*, we require that*

$$\mathsf{View}_{\mathsf{Verifier}^*}(\langle \mathsf{Prover}(\mathbf{y}, \mathbf{w}_1), \mathsf{Verifier}^*(\mathbf{y}, \cdot)\rangle) \approx_Q \mathsf{View}_{\mathsf{Verifier}^*}(\langle \mathsf{Prover}(\mathbf{y}, \mathbf{w}_2), \mathsf{Verifier}^*(\mathbf{y}, \cdot)\rangle).$$

If $(\mathsf{Prover}, \mathsf{Verifier})$ *is a quantum proof system (sound against unbounded provers), we say that* $(\mathsf{Prover}, \mathsf{Verifier})$ *is a* ***quantum witness indistinguishable proof system*** *for* $\mathcal{L}$*.*

Instantiation. By suitably instantiating the constant round WI argument system of Blum [13] with perfectly binding quantum computational hiding commitments, we achieve a constant round quantum WI classical argument system assuming quantum hardness of learning with errors.

Construction. We present a construction of constant round quantum zero-knowledge classical argument system for NP.

Tools

- Perfectly-binding and quantum-computational hiding non-interactive commitments Comm.
- Quantum extraction protocol secure against classical adversaries $\mathsf{cQEXT} = (\mathsf{S}, \mathsf{R})$ associated with the relation $\mathcal{R}_{\mathrm{EXT}}$ as constructed in Sect. 6.
- Quantum witness indistinguishable classical argument system $\Pi_{\mathsf{WI}} = (\Pi_{\mathsf{WI}}.\mathsf{Prover}, \Pi_{\mathsf{WI}}.\mathsf{Verifier})$ (Definition 8) for the relation $\mathcal{R}_{\mathsf{wi}}$ (Fig. 3).

Construction. Let $\mathcal{L}$ be an NP language. We describe a classical interactive protocol $(\mathsf{Prover}, \mathsf{Verifier})$ for $\mathcal{L}$ in Fig. 4.

We prove following lemma in the full version.

Lemma 2. *Assuming the security of* Comm*,* cQEXT *and* Π_{WI}*, the classical interactive protocol* $(\mathsf{Prover}, \mathsf{Verifier})$ *is a quantum zero-knowledge classical argument system for NP.*

Instance: $\left(\mathbf{y}, \mathsf{td}, \left\{(\mathbf{c}_0^{(j)})^*, (\mathbf{c}_1^{(j)})^*\right\}_{j\in[k]}\right)$

Witness: $\left(\mathbf{w}, \left\{(sh_0^{(j)}, \mathbf{d}_0^{(j)}, sh_1^{(j)}, \mathbf{d}_1^{(j)})\right\}_{j\in[k]}\right)$

NP verification: Accept if one of the following two conditions are satisfied:

- $(\mathbf{y}, \mathbf{w}) \in \mathcal{R}$.
- If for every $j \in [k]$, it holds that
  - $(\mathbf{c}_0^{(j)})^* = \mathsf{Comm}(1^\lambda, sh_0^{(j)}; \mathbf{d}_0^{(j)})$
  - $(\mathbf{c}_1^{(j)})^* = \mathsf{Comm}(1^\lambda, sh_1^{(j)}; \mathbf{d}_1^{(j)})$
  - $\mathsf{td} = sh_0^{(j)} \oplus sh_1^{(j)}$

Fig. 3. Relation $\mathcal{R}_{\mathsf{wi}}$ associated with Π_{WI}.

6 QEXT Secure Against Quantum Adversaries

6.1 Construction of QEXT

We present a construction of quantum extraction protocols secure against quantum adversaries, denoted by qQEXT. First, we describe the tools used in this construction.

- Quantum-secure
  computationally-hiding and perfectly-binding non-interactive commitments Comm.
- Quantum fully homomorphic encryption scheme with some desired properties, (qFHE.Gen, qFHE.Enc, qFHE.Dec, qFHE.Eval).
  - It admits homomorphic evaluation of arbitrary computations,
  - It admits perfect correctness,
  - The ciphertext of a classical message is also classical.

  We show in the full version that there are qFHE schemes satisfying the above properties.
- Quantum-secure two-party secure computation SFE with the following properties:
  - Only one party receives the output. We designate the party receiving the output as the receiver SFE.R and the other party to be SFE.S.
  - Security against quantum passive senders.
  - IND-Security against quantum malicious receivers.

- **Trapdoor Committment by Verifier**: Verifier: sample $\mathsf{td} \leftarrow \{0,1\}^{\lambda}$. Compute $\mathbf{c} \leftarrow \mathsf{Comm}(1^{\lambda}, \mathsf{td}; \mathbf{d})$, where $\mathbf{d} \leftarrow \{0,1\}^{\mathrm{poly}(\lambda)}$ is the randomness used in the commitment. Send $\mathbf{c}$ to Prover.

- **Trapdoor Extraction Phase**: Prover and Verifier run the quantum extraction protocol cQEXT with Verifier taking the role of the sender cQEXT.S and Prover taking the role of the receiver cQEXT.R. The input of cQEXT.S is $(1^{\lambda}, \mathbf{c}, \mathbf{d}; \mathbf{r}_{\mathrm{qext}})$ and the input of cQEXT.R is $\left(1^{\lambda}, \mathbf{c}\right)$, where $\mathbf{r}_{\mathrm{qext}}$ is the randomness used by the sender in cQEXT. Let the transcript generated during the execution of cQEXT be $\mathcal{T}_{\mathsf{Verifier} \rightarrow \mathsf{Prover}}$.
  Note: The trapdoor extraction phase will be used by the simulator, while proving zero-knowledge, to extract the trapdoor from the malicious verifier.

- **Trapdoor Commitment by Prover**:
  - Let $k = \lambda$. For every $j \in [k]$, Prover sends $(\mathbf{c}_0^{(j)})^* = \mathsf{Comm}(1^{\lambda}, sh_0^{(j)}; \mathbf{d}_0^{(j)})$ and $(\mathbf{c}_1^{(j)})^* = \mathsf{Comm}(1^{\lambda}, sh_1^{(j)}; \mathbf{d}_1^{(j)})$, where $sh_0^{(j)}, sh_1^{(j)} \xleftarrow{\$} \{0,1\}^{\mathrm{poly}(\lambda)}$.
  - For every $j \in [k]$, Verifier sends bit $b^{(j)} \xleftarrow{\$} \{0,1\}$ to Prover.
  - Prover sends $\left\{ \left(sh_{b^{(j)}}^{(j)}, \mathbf{d}_{b^{(j)}}^{(j)} \right)_{j \in [k]} \right\}$ to Verifier.

- **Check if Verifier cheated in Trapdoor Extraction Phase**: Verifier sends $\mathbf{r}_{\mathrm{qext}}, \mathbf{d}, \mathsf{td}$ to Prover. Then Prover checks the following:
  - Let $\mathcal{T}_{\mathsf{Verifier} \rightarrow \mathsf{Prover}}$ be $(m_1^S, m_1^R, \ldots, m_{t'}^S, m_{t'}^R)$, where the message m_i^R (resp., m_i^S) is the message sent by the receiver (resp., sender) in the i^{th} round and t' is the number of rounds of cQEXT. Let the message produced by $\mathsf{S}\left(1^{\lambda}, \mathbf{c}, \mathbf{d}; \mathbf{r}_{\mathrm{qext}}\right)$ in the i^{th} round be $\widetilde{m}_i^S$.
  - If for any $i \in [t']$, $\widetilde{m}_i^S \neq m_i^S$ then Prover aborts. If $\mathbf{c} \neq \mathsf{Comm}(1^{\lambda}, \mathsf{td}; \mathbf{d})$ then Prover aborts.

- **Quantum WI**: Prover and Verifier run Π_{WI} with Prover taking the role of Π_{WI} prover Π_{WI}.Prover and Verifier taking the role of Π_{WI} verifier Π_{WI}.Verifier. The input to Π_{WI}.Prover is the security parameter 1^{λ}, instance $\left(\mathbf{y}, \mathsf{td}, \left\{ (\mathbf{c}_0^{(j)})^*, (\mathbf{c}_1^{(j)})^* \right\}_{j \in [k]} \right)$ and witness $(\mathbf{w}, \perp)$. The input to Π_{WI}.Verifier is the security parameter 1^{λ} and instance $\left(\mathbf{y}, \mathsf{td}, \left\{ (\mathbf{c}_0^{(j)})^*, (\mathbf{c}_1^{(j)})^* \right\}_{j \in [k]} \right)$.

- **Decision step**: Verifier computes the decision step of Π_{WI}.Verifier.

Fig. 4. (Classical Prover) Quantum zero-knowledge argument systems for NP.

$\mathbf{C}$

Input: CT
Hardwired values: $\mathbf{r}$ (lock)$, \mathbf{k}, \mathsf{SK}_1, \mathsf{CT}^*$.

- $\mathsf{SK}_2' \leftarrow \mathsf{qFHE.Dec}(\mathsf{SK}_1, \mathsf{CT})$
- $\mathbf{r}' \leftarrow \mathsf{qFHE.Dec}(\mathsf{SK}_2', \mathsf{CT}^*)$
- If $\mathbf{r}' = \mathbf{r}$, output $\mathbf{k}$. Else, output $\bot$.

Fig. 5. Circuits used in the lockable obfuscation

f

Input of sender: $(\mathsf{td}, \mathbf{c}, \mathbf{c}_1^*, \ldots, \mathbf{c}_\ell^*, \mathsf{SK}_2)$
Input of receiver: $(\mathbf{d}, r_1, \ldots, r_\ell)$

- If $\left(\mathbf{c} \leftarrow \mathsf{Comm}\left(1^\lambda, (r_1, \ldots, r_\ell); \mathbf{d}\right)\right) \bigwedge \left(\forall i \in [\ell], \mathbf{c}_i^* \leftarrow \mathsf{Comm}\left(1^\lambda, \mathsf{td}_i; r_i\right)\right)$, output SK_2. Here, td_i denotes the i^{th} bit of td.
- Otherwise, output $\bot$.

Fig. 6. Description of the function f associated with the SFE.

- Quantum-secure lockable obfuscation $\mathbf{LObf} = (\mathsf{Obf}, \mathsf{ObfEval})$ for $\mathcal{C}$, where every circuit $\mathbf{C}$, parameterized by $(\mathbf{r}, \mathbf{k}, \mathsf{SK}_1, \mathsf{CT}^*)$, in $\mathcal{C}$ is defined in Fig. 5. Note that $\mathcal{C}$ is a compute-and-compare functionality.

Construction. We construct a protocol (S, R) in Fig. 7 for a NP language $\mathcal{L}$, and the following lemma shows that (S, R) is a quantum extraction protocol.

We prove the following lemma in the full version.

Lemma 3. *Assuming the quantum security of* Comm, SFE, qFHE *and* $\mathbf{LObf}$, (S, R) *is a quantum extraction protocol for* $\mathcal{L}$ *secure against quantum adversaries.*

Input of sender: $(\mathbf{y}, \mathbf{w})$.
Input of receiver: $\mathbf{y}$

- R: sample $(r_1, \ldots, r_\ell) \xleftarrow{\$} \{0,1\}^{\ell \cdot \mathrm{poly}(\lambda)}$. Compute $\mathbf{c} \leftarrow \mathsf{Comm}\left(1^\lambda, (r_1, \ldots, r_\ell); \mathbf{d}\right)$, where $\ell = \lambda$ and $\mathbf{d}$ is the randomness used to compute $\mathbf{c}$. Send $\mathbf{c}$ to S.
- S:
  - Compute the $\mathsf{qFHE.Setup}$ twice; $(\mathsf{PK}_i, \mathsf{SK}_i) \leftarrow \mathsf{qFHE.Setup}(1^\lambda)$ for $i \in \{1, 2\}$.
  - Compute $\mathsf{CT}_1 \leftarrow \mathsf{qFHE.Enc}(\mathsf{PK}_1, (\mathsf{td}||\mathbf{w}))$, where $\mathsf{td} \xleftarrow{\$} \{0,1\}^\lambda$.
  - Compute $\widetilde{\mathbf{C}} \leftarrow \mathsf{Obf}(1^\lambda, \mathbf{C}[\mathbf{r}, \mathbf{k}, \mathsf{SK}_1, \mathsf{CT}^*])$, where $\mathbf{r} \xleftarrow{\$} \{0,1\}^\lambda$ and $\mathbf{k} \xleftarrow{\$} \{0,1\}^\lambda$, CT^* is defined below and $\mathbf{C}[\mathbf{r}, \mathbf{k}, \mathsf{SK}_1, \mathsf{CT}^*]$ is defined in Figure 5.
    - $\mathsf{CT}^* \leftarrow \mathsf{qFHE.Enc}\left(\mathsf{PK}_2, \mathbf{r}\right)$

  Send $\mathsf{msg}_1 = \left(\mathsf{CT}_1, \widetilde{\mathbf{C}}, \mathsf{otp} := \mathbf{k} \oplus \mathsf{SK}_1\right)$.
- R: compute $\mathbf{c}_i^* \leftarrow \mathsf{Comm}\left(1^\lambda, 0; r_i\right)$ for $i \in [\ell]$. Send $(\mathbf{c}_1^*, \ldots, \mathbf{c}_\ell^*)$ to S.
- S and R run SFE, associated with the two-party functionality f defined in Figure 6; S takes the role of $\mathsf{SFE.S}$ and R takes the role of $\mathsf{SFE.R}$. The input to $\mathsf{SFE.S}$ is $(\mathsf{td}, \mathbf{c}, \mathbf{c}_1^*, \ldots, \mathbf{c}_\ell^*, \mathsf{SK}_2)$ and the input to $\mathsf{SFE.R}$ is $(\mathbf{d}, r_1, \ldots, r_\ell)$.

Fig. 7. Quantum extraction protocol (S, R)

Acknowledgements. We are grateful to Kai-Min Chung for many clarifications regarding quantum zero-knowledge proof and argument systems. We thank Thomas Vidick and Urmila Mahadev for answering questions about noisy trapdoor claw-free functions. We thank Abhishek Jain for helpful discussions and pointing us to the relevant references.

References

1. Agarwal, A., Bartusek, J., Goyal, V., Khurana, D., Malavolta, G.: Post-quantum multi-party computation in constant rounds (2020)
2. Alagic, G., Brakerski, Z., Dulek, Y., Schaffner, C.: Impossibility of quantum virtual black-box obfuscation of classical circuits (2020)
3. Ambainis, A., Rosmanis, A., Unruh, D.: Quantum attacks on classical proof systems: the hardness of quantum rewinding. In: FOCS (2014)
4. Ananth, P., Jain, A.: On secure two-party computation in three rounds. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017. LNCS, vol. 10677, pp. 612–644. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70500-2_21

5. Ananth, P., La Placa, R.L.: Secure quantum extraction protocols. arXiv preprint arXiv:1911.07672 (2019)
6. Ananth, P., La Placa, R.L.: Secure software leasing. arXiv preprint arXiv:2005.05289 (2020)
7. Barak, B.: How to go beyond the black-box simulation barrier. In: Proceedings 42nd IEEE Symposium on Foundations of Computer Science, pp. 106–115. IEEE (2001)
8. Bitansky, N., Brakerski, Z., Kalai, Y., Paneth, O., Vaikuntanathan, V.: 3-message zero knowledge against human ignorance. In: Hirt, M., Smith, A. (eds.) TCC 2016. LNCS, vol. 9985, pp. 57–83. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53641-4_3
9. Bitansky, N., Canetti, R., Paneth, O., Rosen, A.: On the existence of extractable one-way functions. SIAM J. Comput. **45**(5), 1910–1952 (2016)
10. Bitansky, N., Kalai, Y.T., Paneth, O.: Multi-collision resistance: a paradigm for keyless hash functions. In: Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, pp. 671–684. ACM (2018)
11. Bitansky, N., Khurana, D., Paneth, O.: Weak zero-knowledge beyond the black-box barrier. In: Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, pp. 1091–1102. ACM (2019)
12. Bitansky, N., Shmueli, O.: Post-quantum zero knowledge in constant rounds. In: STOC (2020)
13. Blum, M.: How to prove a theorem so no one else can claim it. In: Proceedings of the International Congress of Mathematicians, vol. 1, p. 2. Citeseer (1986)
14. Brakerski, Z.: Quantum FHE (almost) as secure as classical. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018. LNCS, vol. 10993, pp. 67–95. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-96878-0_3
15. Brakerski, Z., Christiano, P., Mahadev, U., Vazirani, U., Vidick, T.: A cryptographic test of quantumness and certifiable randomness from a single quantum device. In: 2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS), pp. 320–331. IEEE (2018)
16. Brakerski, Z., Vaikuntanathan, V.: Efficient fully homomorphic encryption from (standard) LWE. SIAM J. Comput. **43**(2), 831–871 (2014)
17. Broadbent, A., Ji, Z., Song, F., Watrous, J.: Zero-knowledge proof systems for QMA. In: 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS), pp. 31–40. IEEE (2016)
18. Chailloux, A., Ciocan, D.F., Kerenidis, I., Vadhan, S.: Interactive and noninteractive zero knowledge are equivalent in the help model. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 501–534. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78524-8_28
19. Feige, U., Lapidot, D., Shamir, A.: Multiple noninteractive zero knowledge proofs under general assumptions. SIAM J. Comput. **29**(1), 1–28 (1999)
20. Gentry, C., et al.: Fully homomorphic encryption using ideal lattices. In: STOC, vol. 9, pp. 169–178 (2009)
21. Gentry, C., Halevi, S., Vaikuntanathan, V.: i-hop homomorphic encryption and rerandomizable yao circuits. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 155–172. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14623-7_9
22. Goldreich, O., Krawczyk, H.: On the composition of zero-knowledge proof systems. SIAM J. Comput. **25**(1), 169–192 (1996)

23. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity and a methodology of cryptographic protocol design. In: 27th Annual Symposium on Foundations of Computer Science, 1986, pp. 174–187. IEEE (1986)
24. Goldreich, O., Oren, Y.: Definitions and properties of zero-knowledge proof systems. J. Cryptol. **7**(1), 1–32 (1994). https://doi.org/10.1007/BF00195207
25. Goyal, R., Koppula, V., Waters, B.: Lockable obfuscation. In: 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), pp. 612–621. IEEE (2017)
26. Hallgren, S., Kolla, A., Sen, P., Zhang, S.: Making classical honest verifier zero knowledge protocols secure against quantum attacks. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008. LNCS, vol. 5126, pp. 592–603. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70583-3_48
27. Jain, R., Kolla, A., Midrijanis, G., Reichardt, B.W.: On parallel composition of zero-knowledge proofs with black-box quantum simulators. arXiv preprint quant-ph/0607211 (2006)
28. Kalai, Y.T., Khurana, D.: Non-interactive non-malleability from quantum supremacy. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019. LNCS, vol. 11694, pp. 552–582. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26954-8_18
29. Kobayashi, H.: General properties of quantum zero-knowledge proofs. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 107–124. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78524-8_7
30. Lombardi, A., Schaeffer, L.: A note on key agreement and non-interactive commitments. IACR Cryptol. ePrint Arch. **2019**, 279 (2019)
31. Mahadev, U.: Classical homomorphic encryption for quantum circuits. In: 2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS), pp. 332–338. IEEE (2018)
32. Mahadev, U.: Classical verification of quantum computations. In: 2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS), pp. 259–267. IEEE (2018)
33. Matsumoto, K.: A simpler proof of zero-knowledge against quantum attacks using Grover's amplitude amplification. arXiv preprint quant-ph/0602186 (2006)
34. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information (2002)
35. Pass, R.: Simulation in quasi-polynomial time, and its application to protocol composition. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 160–176. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-39200-9_10
36. Pass, R., Wee, H.: Black-box constructions of two-party protocols from one-way functions. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 403–418. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00457-5_24
37. Prabhakaran, M., Rosen, A., Sahai, A.: Concurrent zero knowledge with logarithmic round-complexity. In: FOCS (2002)
38. Unruh, D.: Quantum proofs of knowledge. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 135–152. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29011-4_10
39. Unruh, D.: Everlasting multi-party computation. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8043, pp. 380–397. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40084-1_22
40. Unruh, D.: Computationally binding quantum commitments. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016. LNCS, vol. 9666, pp. 497–527. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49896-5_18

41. Vidick, T., Zhang, T.: Classical zero-knowledge arguments for quantum computations. arXiv preprint arXiv:1902.05217 (2019)
42. Watrous, J.: Zero-knowledge against quantum attacks. SIAM J. Comput. **39**(1), 25–58 (2009)
43. Wichs , D., Zirdelis, G.: Obfuscating compute-and-compare programs under LWE. In: 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), pp. 600–611. IEEE (2017)