Post-quantum Zero Knowledge in Constant Rounds

Nir Bitansky
nirbitan@tau.ac.il
Tel Aviv University
Tel Aviv, Israel

Omri Shmueli
omrishmueli@mail.tau.ac.il
Tel Aviv University
Tel Aviv, Israel

ABSTRACT

We construct a constant-round zero-knowledge classical argument for **NP** secure against quantum attacks. We assume the existence of Quantum Fully-Homomorphic Encryption and other standard primitives, known based on the Learning with Errors Assumption for quantum algorithms. As a corollary, we also obtain a constant-round zero-knowledge quantum argument for **QMA**.

At the heart of our protocol is a new *no-cloning* non-black-box simulation technique.

CCS CONCEPTS

• **Theory of computation** → **Cryptographic protocols**; • **Security and privacy** → **Mathematical foundations of cryptography**.

KEYWORDS

post-quantum cryptography, zero-knowledge, non-black-box simulation

ACM Reference Format:
Nir Bitansky and Omri Shmueli. 2020. Post-quantum Zero Knowledge in Constant Rounds. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing (STOC '20), June 22–26, 2020, Chicago, IL, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3357713.3384324

1 INTRODUCTION

Zero-knowledge protocols allow to prove statements without revealing anything but the mere fact that they are true. Since their introduction by Goldwasser, Micali, and Rackoff [GMR89] they have had a profound impact on modern cryptography and theoretical computer science at large. Following more than three decades of exploration, zero-knowledge protocols are now quite well understood in terms of their expressiveness and round complexity. In particular, under standard computational assumptions, arbitrary **NP** statements can be proved in only a constant number of rounds [GMW86, GK96a].

In this work, we consider classical zero-knowledge protocols with *post-quantum security*, namely, protocols that can be executed by classical parties, but where both soundness and zero knowledge are guaranteed even against efficient quantum adversaries. Here our understanding is far more restricted than in the classical setting. Indeed, not only are we faced with stronger adversaries, but also have to deal with the fact that quantum information behaves in a fundamentally different way than classical information, which summons new challenges in the design of zero-knowledge protocols.

In his seminal work [Wat09], Watrous developed a new quantum simulation technique and used it to show that classical zero-knowledge protocols for **NP**, such as the Goldreich-Micali-Wigderson 3-coloring protocol [GMW86], are also zero knowledge against quantum verifiers, assuming commitments with post-quantum hiding. These protocols are, in fact, *proof systems* meaning that soundness holds against unbounded adversarial provers, let alone efficient quantum ones. As in the classical setting, to guarantee a negligible soundness error (the gold standard in cryptography) these protocols require a polynomial number of rounds.

Watrous' technique does not apply for classical constant-round protocols. In fact, constant-round zero-knowledge protocols with post-quantum security remains an open question, *even when the honest parties and communication are allowed to be quantum.* The gap between classical and quantum zero knowledge stems from fundamental aspects of quantum information such as the no-cloning theorem [WZ82] and quantum state disturbance [FP96]. These pose a substantial barrier for classical zero-knowledge simulation techniques, a barrier that has so far been circumvented only in specific settings (such as, [Wat09]). Overcoming these barriers in the context of constant-round zero-knowledge seems to require a new set of techniques.

1.1 Results

Under standard computational assumptions, we resolve the above open question — we construct a classical, post-quantumly secure, computational-zero-knowledge argument for **NP** in a constant number of rounds (with a negligible soundness error). That is, the honest verifier and prover (given a witness) are efficient classical algorithms. In terms of security, both zero-knowledge and soundness hold against polynomial-size quantum circuits with non-uniform quantum advice.

Our construction is based on fully-homomorphic encryption supporting the evaluation of quantum circuits (QFHE) as well as additional standard classical cryptographic primitives. All are required to be secure against efficient quantum algorithms with non-uniform quantum advice. QFHE was recently constructed [Mah18a, Bra18] based on the assumption that the Learning with Errors Problem [Reg09] is hard for the above class of algorithms (from hereon, called QLWE) and a circular security assumption (analogous to the assumptions required for multi-key FHE in the classical setting). All other required primitives can be based on the QLWE assumption.

STOC '20, June 22–26, 2020, Chicago, IL, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-6979-4/20/06…$15.00
https://doi.org/10.1145/3357713.3384324

Theorem 1.1 (informal). *Assuming QLWE and QFHE, there exist a classical, post-quantumly secure, computational-zero-knowledge argument in a constant number of rounds for any $\mathcal{L} \in \boldsymbol{NP}$.*

Combining our zero-knowledge protocol with previous work by Broadbent et al. [BJSW16, BG19], yields constant-round zero-knowledge arguments for **QMA** with quantum honest parties.

Corollary 1.1 (informal). *Assuming QLWE and QFHE, there exist a quantum, post-quantumly secure, computational-zero-knowledge argument in a constant number of rounds for any $\mathcal{L} \in \boldsymbol{QMA}$.*

Main Technical Contribution: Non-Black-Box Quantum Extraction. Our main technical contribution is a new technique for extracting information from quantum circuits in a constant number of rounds. The technique circumvents the quantum information barriers previously mentioned. A key feature that enables this is using the adversary's circuit representation in a non-black-box manner.

The technique, in particular, yields a constant round extractable commitment. In such a commitment protocol, the verifier can commit to a classical (polynomially long) string. This commitment is perfectly binding, and hiding against efficient quantum receivers. Furthermore, it guarantees the existence of a simulator, which given non-black-box access to the sender's code, can simulate its view while extracting the committed plaintext. Further details are given in the technical overview below.

1.2 Technical Overview

We next discuss the main challenges in the design of post-quantum zero knowledge in constant rounds, and our main technical ideas toward overcoming these challenges.

1.2.1 Classical Protocols and the Quantum Barrier.

To understand the challenges behind post-quantum zero knowledge, let us first recall how classical constant-round protocols work, and identify why they fail in the quantum setting. Classical constant-round protocols typically involve three main steps: (1) a prover commitment α to a set of bits, (2) a verifier challenge β, and (3) a prover response γ, in which it opens the commitments corresponding to the challenge β. For instance, in the 3-coloring protocol of [GMW86], the prover commits to the (randomly permuted) vertex colors, the verifier picks some challenge edge, and the prover opens the commitments corresponding to the vertices of that edge. To guarantee a negligible soundness error, this is repeated in parallel a polynomial number of times.

As describe so far, the protocol satisfies a rather weak zero-knowledge guarantee — a simulator can efficiently simulate the verifier's view in the protocol *if it knows the verifier's challenge β ahead of time.* To obtain an actual zero-knowledge protocol, we need to exhibit a simulator for any *malicious* verifier, including ones who may arbitrarily choose their challenge depending on the prover's message α. For this purpose, an initial step (0) is added where the verifier commits ahead of time to its challenge, later opening it in step (2) [GK96a].

The added step allows the simulator to obtain the verifier's challenges ahead of time by means of *rewinding*. Specifically, having obtained the verifier commitment, the simulator takes a snapshot of the verifier's state and then runs it twice: first it generates a bogus prover commitment, and obtains the verifier challenge, then with the challenges at hand, it returns to the snapshot (effectively rewinding the verifier) and runs the verifier again to generate the simulated execution. The binding of the verifier's commitment guarantees that it will never use a different challenge, and thus simulation succeeds.

Barriers to Post-Quantum Security. By appropriately instantiating the verifier commitment, the above protocol can be shown to be sound against unbounded provers, and in particular efficient quantum provers. One could expect that by instantiating the prover's commitments so to guarantee hiding against quantum adversaries, we would get post-quantum zero knowledge. However, we do not know how to prove that such a protocol is zero knowledge against quantum verifiers. Indeed, the simulation strategy described above fails due to two basic concepts of quantum information theory:

- **No Cloning:** General quantum states cannot be copied. In particular, the simulator cannot take a snapshot of the verifier's state.
- **Quantum State Disturbance:** General quantum circuits, which in particular perform measurements, are not reversible. Once the simulator evaluates the verifier's quantum circuit to obtain its challenge, the verifier's original state (prior to this bogus execution) has already been disturbed and cannot be recovered.

Watrous [Wat09] showed that in certain settings the rewinding barrier can be circumvented. He presents a *quantum rewinding lemma* that roughly, shows how *non-rewinding* simulators that succeed in simulating only with some noticeable probability can be amplified into full-fledged simulators. The quantum rewinding lemma allows proving that classical protocols, like the GMW protocol are post-quantum zero knowledge (assuming commitments with hiding against quantum adversaries). The technique is insufficient, however, to prove post-quantum zero knowledge of existing constant-round protocols *with a negligible soundness error*, such as the GK protocol described above. For such protocols, non-rewinding simulators with a noticeable success probability are not known.

Can Non-Black-Box Techniques Cross the Quantum Barriers? Rewinding is, in fact, often an issue *also in the classical setting*. Starting with the work of Goldreich and Krawczyk [GK96b], it was shown that constant-round zero-knowledge protocols with certain features, such as a public-coin verifier, cannot be obtained using simulators that only use the verifier's next message function as a black box. That is, simulators that are based solely on rewinding. Surprisingly, Barak [Bar01] showed that these barriers can be circumvented using *non-black-box techniques.* He constructed a constant-round public-coin zero-knowledge protocol where the simulator takes advantage of the explicit circuit representation of the verifier. Following Barak's work, different non-black-box techniques have been introduced to solve various problems in cryptography (c.f., [DGS09, CLP13, Goy13, BP15, CPS16]).

A natural question is whether we can leverage classical protocols with non-black-box simulators, such as Barak's, in order to circumvent the discussed barriers in the quantum setting. Trying to answer this question reveals several challenges. One inherent challenge is that classical non-black-box techniques naturally involve cryptographic tools that support classical computations. Obtaining zero knowledge against quantum verifiers would require analogous tools for quantum computations. As an example, Barak relies on the existence of constant-round succinct proof systems for the correctness of classical computations; to obtain post-quantum zero knowledge, such a protocol would need to support also quantum computations, while (honest) verification should remain classical. Existing protocols for classical verification of quantum computations [Mah18b] are neither constant round nor succinct.

Another family of non-black-box techniques [BP15, BKP19], different from that of Barak, is based on fully-homomorphic encryption. Here (as mentioned above) constructions for homomorphic evaluation of quantum computations exist [Mah18a, Bra18]. The problem is that the mentioned non-black-box techniques *do perform state cloning*. Roughly speaking, starting from the same state, they evaluate the verifier's computation (at least) twice: once homomorphically, under the encryption, and once in the clear.[1] An additional hurdle is proving soundness against quantum provers. Known non-black-box techniques are sound against efficient classical provers, and often use tools that are not known in the quantum setting, such as constant-round knowledge extraction (which is further discussed below).

Our main technical contribution is devising a non-black-box technique that copes with the above challenges. We next explain the main ideas behind the technique.

1.2.2 *Our Technique: A No-Cloning Extraction Procedure.*

Toward describing the technique, we restrict attention to a more specific problem. Specifically, constructing a constant-round post-quantum zero-knowledge protocol can be reduced to the problem of constructing constant-round *quantumly-extractable commitments*. We recall what such commitments are and why they are sufficient, and then move to discuss the commitments we construct.

A quantumly-extractable commitment is a classical protocol between a sender Sen and a receiver Rec. The protocol satisfies the standard (statistical) binding and post-quantum hiding, along with a plaintext extraction guarantee. Extraction requires that there exists an efficient quantum simulator Ext that given any malicious sender Sen^*, represented by a polynomial-size quantum circuit, can simulate the view of Sen^* in the commitment protocol while extracting the committed plaintext message. Specifically, $\mathsf{Ext}(\mathsf{Sen}^*)$ outputs a classical transcript $\widetilde{T}$, a quantum state $|\widetilde{\psi}\rangle$, and an extracted plaintext $\widetilde{m}$ that are computationally indistinguishable from a real transcript, state, and plaintext $(T, |\psi\rangle, m)$, where T and $|\psi\rangle$ are the transcript and sender state generated at the end of a real interaction between the receiver Rec and sender Sen^*, and m is the plaintext fixed by the commitment transcript T.

Such commitments allow enhancing the classical four-step protocol described before to satisfy post-quantum zero-knowledge. We simply instantiate the verifier's commitment to the challenge β in step (0) with a quantumly-extractable commitment. To simulate a malicious quantum verifier V^*, the zero-knowledge simulator can then invoke the commitment simulator $\mathsf{Ext}(\mathsf{V}^*)$, with V^* acting as the sender, to obtain a simulated commitment as well as the corresponding challenge β. Now the simulator knows the challenge ahead of time, before producing the prover message α in step (1), and using the (simulated) verifier state $|\widetilde{\psi}\rangle$, can complete the simulation, *without any state cloning*. (Proving soundness is actually tricky on its own due to malleability concerns. We remain focused on zero knowledge for now).

The challenge is of course to obtain constant-round commitments with *no-cloning extraction*. Indeed, classically-extractable commitments have been long known in constant rounds under minimal assumptions, based on rewinding (and thus state cloning) [PRS02]. We next describe our non-black-box technique and how it enables quantum extraction without state cloning.

The Non-Black-Box Quantum Extraction Technique: A Simple Case. To describe the technique, we first focus on a restricted class of adversarial senders that are *non-aborting and explainable*. The notion of non-aborting explainable senders considers senders Sen^* whose messages can always be *explained* as a behavior of the honest (classical) sender with respect to *some* plaintext and randomness (finding this explanation may be inefficient); in particular, they never abort. The notion further restricts that of *(aborting) explainable adversaries* from [BKP19], which also allows aborts. To even further simplify our exposition, we first address classical (rather than quantum) senders, but crucially, while avoiding any form of state cloning. Later on, we shall address general quantum adversaries.

Our protocol is inspired by [BP15, BKP19] and relies on two basic tools. The first is fully-homomorphic encryption (FHE) — an encryption scheme that allows to homomorphically apply any polynomial-size circuit C to an encryption of x to obtain a new encryption of $C(x)$, proportional in size to the result $|C(x)|$ (the size requirement is known as *compactness*). The second is *compute-and-compare program obfuscation* (CCO). A compute-and-compare program $\mathsf{CC}[f, u, z]$ is given by a function f (represented as a circuit), a target string u in its range, and a message z; it outputs z on every input x such that $f(x) = u$, and rejects all other inputs. A corresponding obfuscator compiles any such program into a program $\widetilde{\mathsf{CC}}$ with the same functionality. In terms of security, provided that the target u has high entropy conditioned on f and z, the obfuscated program is computationally indistinguishable from a simulated dummy program, independent of (f, u, z). Such post-quantumly-secure obfuscators are known under QLWE [GKW17, WZ17, GKVW19].

To commit to a message m, the protocol consists of three steps:

(1) The sender Sen samples:
- two random strings u and v,
- a secret key sk for an FHE scheme,
- an FHE encryption $\mathsf{ct}_v = \mathsf{FHE.Enc}_{\mathsf{sk}}(v)$ of v,
- an obfuscation $\widetilde{\mathsf{CC}}$ of $\mathsf{CC}[f, u, z]$, where $z = (m, \mathsf{sk})$ and $f = \mathsf{FHE.Dec}_{\mathsf{sk}}$ is the FHE decryption circuit.

It then sends $(\mathsf{ct}_v, \widetilde{\mathsf{CC}})$ to the receiver Rec.

(2) The receiver Rec sends a guess v'.

(3) Sen rewards a successful guess: if $v = v'$, it sends back u (and otherwise $\perp$).

The described commitment protocol comes close to our objective. First, it is binding — the obfuscation $\widetilde{\mathsf{CC}}$ uniquely determines $z = (m, \mathsf{sk})$. Second, it is hiding — a receiver (even if malicious) gains no information about the message m. To see this, we argue that no receiver sends $v' = v$ at the second message, but with negligible

[1] In fact, Barak's technique also seems to require state cloning. Roughly speaking, the same verifier state is used once for simulating the main verifier execution and once when computing the proof for the verifier's computation.

probability. Indeed, given only the first sender message $(\mathsf{ct}_v, \widetilde{\mathsf{CC}})$, the receiver obtains no information about u. Hence, we can invoke the CCO security and replace the obfuscation $\widetilde{\mathsf{CC}}$ with a simulated one, which is independent of the secret FHE key sk. This, in turn, allows us to invoke the security of encryption to argue that the first message $(\mathsf{ct}_v, \widetilde{\mathsf{CC}})$ hides v. It follows that the third sender message is $\perp$ (rather than the target u) with overwhelming probability, which again by CCO security implies that the entire view of the receiver can be simulated independently of m.

Lastly, a non-black-box simulator, given the circuit representation of an explainable sender Sen^*, can simulate the sender's view, while extracting m. It first runs the sender to obtain the first message $(\mathsf{ct}_v, \widetilde{\mathsf{CC}})$. At this point, it can use the sender's circuit Sen^* to continue the emulation of Sen^* *homomorphically under the encryption* ct_v. The key point is that, under the encryption, we do have v. We can (homomorphically) feed v to the sender, and obtain an encryption ct_u of u. Now, the simulator feeds ct_u to the obfuscation $\widetilde{\mathsf{CC}}$, and gets back $z = (m, \mathsf{sk})$. (Note that here the compactness of FHE is crucial — the sender Sen^* could be of arbitrary polynomial size, whereas $\widetilde{\mathsf{CC}}$ and thus also ct_u are of fixed size.)

Having extracted m, it remains to simulate the inner (for now, classical) state ψ of the sender S^* and the full interaction transcript T. These are actually available, but in encrypted form, as a result of the previous homomorphic computation. Here we use the fact that the extracted z also includes the decryption key sk, allowing us to obtain the state ψ and transcript T *in the clear.*

An essential difference between the above extraction procedure and previous non-black-box extraction techniques (e.g., [BP15, BKP19]) is that *it does not perform any state cloning.* As explained earlier, previous procedures would perform the same computation twice, once under the encryption, and once in the clear. Here we perform the computation once, partially in the clear, and partially homomorphically. Crucially, we have a mechanism to peel off the encryption at the end of second part so that we do not have to redo the computation in the clear.

Indistinguishability through Secure Function Evaluation. The described protocol does not quite achieve our objective. The simulated interaction is, in fact, easy to distinguish from a real one. Indeed, in a simulated interaction the simulator's guess in the second message is $v' = v$, whereas the receiver cannot produce this value. To cope with this problem, we augment the protocol yet again, and perform the second step under a *secure function evaluation* (SFE) protocol. This can be thought of as homomorphic encryption with an additional *circuit privacy* guarantee, which says that the result of homomorphic evaluation of a circuit, reveals nothing about the evaluated circuit to the decryptor, except of course from the result of evaluation.

The augmented protocol is similar to the previous one, except for the last two steps, now done using SFE:

(1) The sender Sen samples:
- two random strings u and v,
- a secret key sk for an FHE scheme,
- an FHE encryption $\mathsf{ct}_v = \mathsf{FHE.Enc}_{\mathsf{sk}}(v)$ of v,
- an obfuscation $\widetilde{\mathsf{CC}}$ of $\mathsf{CC}[f, u, z]$, where $z = (m, \mathsf{sk})$ and $f = \mathsf{FHE.Dec}_{\mathsf{sk}}$ is the FHE decryption circuit.

It then sends $(\mathsf{ct}_v, \widetilde{\mathsf{CC}})$ to the receiver Rec.

(2) The receiver Rec sends $\mathsf{ct}'_{v'}$, a guess v' encrypted using SFE. (The honest receiver sets v' arbitrarily.)

(3) Sen homomorphically evaluates the function that given input v, returns u (and otherwise $\perp$). Sen then returns the resulting ciphertext to Rec.

The homomorphic computation done by the simulator in the new protocol is augmented accordingly — instead of sending u and obtaining v directly, it now sends an SFE encryption of u and obtains back an SFE encryption of v, which it can then decrypt to obtain v. Thus, as before, the homomorphic computation results in an FHE encryption of v. Indistinguishability of the simulated sender view from the real sender view now follows since the SFE encryption $\mathsf{ct}'_{v'}$ hides v'. The SFE circuit privacy guarantees that the homomorphic SFE evaluation does not leak any information about the target u, as long as the receiver does not send an SFE encryption of v.

A Malleability Problem and its Resolution. While we could argue before that a malicious receiver cannot output v in the clear, arguing that it does not output an SFE encryption of v is more tricky. In particular, the receiver might be able to somehow maul the FHE encryption ct_v to get an SFE encryption ct'_v of the value v, without actually "knowing" the value v. Classically, such malleability problems are solved using *extraction.* If we could efficiently extract the value encrypted in the SFE encryption ct', then we could rely on the previous argument. However, as explained before, efficient extraction is classically achieved using rewinding and thus state cloning. While so far we have focused on avoiding state cloning for the sake of simulating the sender, we should also avoid state cloning when proving hiding of the commitment as we are dealing with quantum receivers. It seems like we are back to square one.

To circumvent the problem, we rely on the fact that the hiding requirement of the commitment is relatively modest — commitments to different plaintexts should be indistinguishable. This is in contrast the efficient simulation requirement for the sender (needed for efficient zero knowledge simulation). Here one commonly used solution is *complexity leveraging* — we can design the SFE, FHE, and CCO so that extraction from SFE encryptions can be done in brute force, without any state cloning, and without compromising the security of the FHE and CCO. This comes at the cost of assuming subexponential (rather than just polynomial) hardness of the primitives in use.

A different solution, which is also the one we use in the body of the paper, relies on hardness against efficient quantum adversaries with *non-uniform quantum advice* (instead of subexponential hardness). Specifically, the receiver sends a commitment to the SFE encryption key in the beginning of the protocol. The reduction establishing the hiding of the protocol gets as non-uniform advice the initial receiver (quantum) state that maximizes the probability of breaking hiding, along with the corresponding SFE key. This allows for easy extraction from SFE encryptions, without any state cloning.

The full solution contains additional steps meant to establish that the receiver's messages are appropriately structured (e.g., the receiver's commitment defines a valid SFE key, and the SFE encryption later indeed uses that key). This is done using standard techniques based on witness-indistinguishable proofs, which exist in a constant number of rounds [GMW86] assuming commitments with post-quantum hiding (and in particular, QLWE).

Dealing with Quantum Adversaries. Above, we have assumed for simplicity that the sender is classical and have shown a simulation strategy that requires no state cloning. We now explain how the protocol is augmented to deal with quantum senders (for now still restricting attention to non-aborting explainable senders). The first natural requirement in order to deal with quantum senders is that the cryptographic tools in use (e.g., SFE encryption) will be postqantum secure. This can be guaranteed assuming QLWE.

As already mentioned earlier in the introduction, post-quantum security alone is not enough — we need to make sure that our non-black-box extraction technique can also work with quantum, rather than classical, circuits representing the sender Sen^*. For this purpose, we use *quantum* fully-homomorphic encryption (QFHE). In a QFHE scheme, the encryption and decryption keys are (classical) strings and the encryption and decryption algorithms are classical provided that the plaintext is classical (and otherwise quantum). Most importantly, QFHE allows to homomorphically evaluate quantum circuits. Such QFHE schemes were recently constructed in [Mah18a, Bra18] based on QLWE and a circular security assumption (analogous to the assumptions required for multi-key FHE in the classical setting).

The augmented protocol simply replaces the FHE scheme with a QFHE scheme (other primitives, such as the SFE and compute-and-compare are completely classical in terms of functionality and only need to be post-quantum secure). In the augmented protocol, the honest sender and receiver still act classically. In contrast, the non-black-box simulator described before is now quantum — it homomorphically evaluates the quantum sender circuit Sen^*. A technical point is that QFHE should support the evaluation of a quantum circuit with an additional quantum auxiliary input — in our case the quantum sender Sen^* and its inner state after it sends the first message. This is achieved by existing QFHE schemes (for instance, by using their public key encryption mode, and encrypting the initial state prior to the computation).

Dealing with Aborts. So far, we have dealt with explainable senders that are non-aborting. This is indeed a strong restriction and in fact, quantumly-extractable commitments against this class of senders can be achieved using black-box techniques (see more in the related work section). However, considering an adversary who, with noticeable probability, may abort at some stage of the protocol, existing black-box techniques completely fail (even if the adversary is explainable up to the abort). In contrast, as we shall see, our non-black-box technique will enable simulation also for aborting senders.

In our protocol, an aborting sender Sen^* may refuse to perform the SFE evaluation in the last step of the protocol. In this case, the simulator will get stuck — the simulated transcript and sender state $|\psi\rangle$ will remain forever locked under the encryption (since the simulator cannot use the obfuscation $\widetilde{\mathsf{CC}}$ to get the decryption key sk). Accordingly, the described simulator successfully simulates senders that never abort, but fails to simulate senders that abort (noticeably often). We next observe that there is, in fact, a non-rewinding simulation strategy also for the other extreme, namely for senders Sen^* that (almost) always abort. Here the simulator would simply send *in the clear* (rather than under FHE) an SFE encryption $\mathsf{ct}'_{v'}$ of an arbitrary string v', just like the honest receiver Rec. In this case, the simulated sender view is identical to its view in a real interaction (and since the sender Sen^* aborts, there is no need to extract the plaintext message).

We show that the two simulators described, $\mathsf{Sim}_{\mathrm{na}}$ for never-aborting senders and $\mathsf{Sim}_{\mathrm{aa}}$ for always-aborting senders, can be combined into a simulator for general senders (which sometimes abort). This is enabled by the fact that simulated receiver messages $\mathsf{ct}'_{v'}$ generated by the two simulators are indistinguishable due to the hiding of SFE encryptions. Accordingly, the sender's choice of whether to abort or not is (computationally) independent of whether we are simulating using the first simulator $\mathsf{Sim}_{\mathrm{na}}$ or the second $\mathsf{Sim}_{\mathrm{aa}}$. This gives rise to a combined simulator $\mathsf{Sim}_{\mathrm{comb}}$, which flips a random coin $b \leftarrow \{\mathrm{na}, \mathrm{aa}\}$ to predict whether an abort will occur, and then runs Sim_b. The combined simulator $\mathsf{Sim}_{\mathrm{comb}}$ succeeds if it guessed correctly, which occurs with probability (negligibly close to) half.

Applying Watrous' Quantum Rewinding Lemma. The above is reminiscent of the simulation strategy in classical 3-message zero-knowledge protocols (with a large soundness error), such as the GMW 3-coloring protocol [GMW87]. In these protocols, for each possible verifier challenge β there exists a non-rewinding simulator Sim_β, and the combined simulator $\mathsf{Sim}_{\mathrm{comb}}$ tries to guess the challenge β and apply the corresponding simulator. Similarly to the combined simulator in our protocol, the verifier's choice of challenge β is (computationally) independent of $\mathsf{Sim}_{\mathrm{comb}}$'s guess, and thus the simulator $\mathsf{Sim}_{\mathrm{comb}}$ succeeds in simulating with some fixed noticeable probability (specifically $2^{-|\beta|}$).

The advantage of such simulators (non-rewinding and successful with fixed noticeable probability) is that they can be amplified to full-fledged simulators, both classically and quantumly. In the classical setting, a full-fledged simulator Sim can be obtained by rerunning $\mathsf{Sim}_{\mathrm{comb}}$ until it succeeds. We can, in fact, apply the same rerunning strategy also for quantum verifiers. However, this does not guarantee zero knowledge against verifiers with quantum auxiliary input (since each execution of $\mathsf{Sim}_{\mathrm{comb}}$ may disturb the verifier's auxiliary state). To obtain zero knowledge against verifiers with quantum auxiliary input, we apply Watrous' quantum rewinding lemma [Wat09], which shows how to faithfully amplify the combined simulator $\mathsf{Sim}_{\mathrm{comb}}$ in the presence of quantum auxiliary input.

From Explainable Adversaries to Malicious Ones. The only remaining gap is the assumption that senders are explainable; that is, the messages they send (up to the point that they possibly abort), can always be explained as messages that would be sent by the honest (classical) sender for some plaintext and randomness. The simulator $\mathsf{Sim}_{\mathrm{na}}$ (for never-aborting verifiers) crucially relies on this; in particular, the CCO $\widetilde{\mathsf{CC}}$ and the FHE ciphertext ct_v must be formed consistently with each other for the simulator to work. Importantly, it suffices that *there exists an explanation* for the messages, and we do not have to efficiently extract it as part of the simulation;[2]

[2] This is in contrast to other restrictions of the adversary considered in the literature, like semi-honest and semi-malicious adversaries [GMW87, HIK+11, BGJ+13].

indeed, efficient quantum extraction is exactly the problem we are trying to solve.

The commitment protocol against explainable senders naturally gives rise to a zero-knowledge protocol against explainable verifiers. As is often the case in the design of zero knowledge protocols (see discussion in [BKP19]), dealing with explainable verifiers is actually the hard part of designing zero-knowledge protocols. Indeed, we use a generic transformation of [BKP19], slightly adapted to our setting, which converts zero-knowledge protocols against explainable verifiers to ones against arbitrary malicious verifiers. The transformation is based on constant-round (post-quantumly-secure) witness-indistinguishable proofs, which as mentioned before can be obtained based on QLWE.

1.3 More Related Work on Post-Quantum Zero Knowledge

The study of post-quantum zero-knowledge (QZK) protocols was initiated by van de Graaf [VDGC97], who first observed that traditional zero-knowledge simulation techniques, based on rewinding, fail against quantum verifiers. Subsequent work has further explored different flavors of zero knowledge and their limitations [Wat02], and also demonstrated that relaxed notions such as zero-knowledge with a trusted common reference string can be achieved [Kob03, DFS04]. Watrous [Wat09] was the first to show that the barriers of quantum information theory can be crossed, demonstrating a post-quantum zero-knowledge protocol for **NP** (in a polynomial number of rounds).

*Zero Knowledge for **QMA**.* Another line of work aims at constructing quantum (rather than classical) protocols for **QMA** (rather than **NP**). Following a sequence of works [BOCG+06, Liu06, DNS10, DNS12, MHNF15], Broadbent, Ji, Song and Watrous [BJSW16] show a zero-knowledge quantum proof system for all of **QMA** (in a polynomial number of rounds).

Quantum Proofs and Arguments of Knowledge. Extracting knowledge from quantum adversaries was investigated in a sequence of works [Unr12, HSS11, LN11, ARU14]. A line of works considered different variants of quantum proofs and arguments of knowledge (of the witness), proving both feasibility results and limitations. In particular, Unruh [Unr12] shows that assuming post-quantum injective one-way functions, some existing systems are a quantum proof of knowledge. He identifies a certain *strict soundness* requirement that suffices for such an implication. Ambainis, Rosmanis and Unruh [ARU14] give evidence that this requirement may be necessary.

Based on QLWE, Hallgren, Smith, and Song [HSS11] and Lunemann and Nielsen [LN11] show argument of knowledge where it is also possible to simulate the prover's state (akin to our simulation requirement of the sender's state). Unruh further explores arguments of knowledge in the context of computationally binding quantum commitments [Unr16b, Unr16a]. All of the above require a polynomial number of rounds to achieve a negligible knowledge error.

Zero-Knowledge Multi-Prover Interactive Proofs. Two recent works by Chiesa et al. [CFGS18] and by Grilo, Slofstra, and Yuen [GSY19] show that **NEXP** and **MIP***, respectively, have *perfect* zero-knowledge multi-prover interactive proofs (against entangled quantum provers).

Concurrent Work. In concurrent and independent work, Ananth and La Placa [AP] developed a non-black-box quantum extraction protocol that share some of our ideas and is based on similar computational assumptions. They used it to obtain quantum zero-knowledge, but only against explainable non-aborting verifiers.

A Word on Strict Commitments and Non-Aborting Verifiers. In [Unr12], Unruh introduces a notion of *strict commitments*, which are commitments that fix not only the plaintext, but also the randomness (e.g. Blum-Micali [BM84]), and are known to exist based on injective one-way functions. As mentioned in our technical overview, using such commitments it is possible to obtain zero-knowledge in constant rounds *against non-aborting explainable verifiers* through the GK four-step template we discussed in the overview. Roughly speaking, this is because when considering verifiers that always open their (strict) commitments, we are assured that measuring their answer does not disturb the verifier state, as this answer is information-theoretically fixed. This effectively allows to perform rewinding.

2 CONSTANT-ROUND ZERO-KNOWLEDGE ARGUMENTS FOR NP

In this section we construct a classical argument system for an arbitrary NP language $\mathcal{L}$ (according to Definition 2.1), with a constant number of rounds, quantum soundness and quantum zero-knowledge. We describe the protocol and the simulation only. The complete proof, including soundness, appears in the full version.

Definition 2.1 (Post-Quantum Zero-Knowledge Argument for NP). *A classical protocol* (P,V) *with an honest PPT prover* P *and an honest PPT verifier* V *for a language* $\mathcal{L} \in \mathbf{NP}$ *is a post-quantum zero-knowledge argument if it satisfies the following:*

(1) **Perfect Completeness:** *For any* $\lambda \in \mathbb{N}, x \in \mathcal{L} \cap \{0,1\}^\lambda, w \in \mathcal{R}_\mathcal{L}(x)$,

$$\Pr[\mathsf{OUT}_\mathsf{V}\langle \mathsf{P}(w),\mathsf{V}\rangle(x)=1]=1 \ .$$

(2) **Quantum Soundness:** *For any quantum polynomial-size prover* $\mathsf{P}^* = \left\{\mathsf{P}^*_\lambda, \rho_\lambda\right\}_{\lambda\in\mathbb{N}}$*, there exists a negligible function* $\mu(\cdot)$ *such that for any security parameter* $\lambda \in \mathbb{N}$ *and any* $x \in \{0,1\}^\lambda \setminus \mathcal{L}$*,*

$$\Pr\left[\mathsf{OUT}_\mathsf{V}\langle \mathsf{P}^*_\lambda(\rho_\lambda),\mathsf{V}\rangle(x)=1\right] \le \mu(\lambda) \ .$$

(3) **Quantum Zero Knowledge** *There exists a quantum polynomial-time simulator* Sim*, such that for any quantum polynomial-size verifier* $\mathsf{V}^* = \left\{\mathsf{V}^*_\lambda, \rho_\lambda\right\}_{\lambda\in\mathbb{N}}$*,*

$$\{\mathsf{OUT}_{\mathsf{V}^*_\lambda}\langle \mathsf{P}(w),\mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x)\}_{\lambda,x,w} \approx_c \{\mathsf{Sim}(x,\mathsf{V}^*_\lambda,\rho_\lambda)\}_{\lambda,x,w} \ ,$$

where $\lambda \in \mathbb{N}$*,* $x \in \mathcal{L} \cap \{0,1\}^\lambda$*,* $w \in \mathcal{R}_\mathcal{L}(x)$*.*

- *If* V^* *is a classical circuit, then the simulator is computable by a classical polynomial-time algorithm.*

Below are the cryptographic ingredients used in the protocol. For the definition of each primitive see the full version.

Ingredients and notation:

- A non-interactive commitment scheme Com.
- A CC obfuscator Obf.
- A quantum fully homomorphic encryption scheme (QHE.Keygen, QHE.Enc, QHE.QEnc, QHE.Dec, QHE.QDec, QHE.Eval).
- A 2-message function-hiding secure function evaluation scheme (SFE.Gen, SFE.Enc, SFE.Eval, SFE.Dec).
- A public-coin 3-message WI proof (WI.P, WI.V).
- A 3-message sigma protocol $(\Sigma.\mathsf{P}, \Sigma.\mathsf{V})$ for the language $\mathcal{L}$.

We describe the protocol in Figure 1.

2.1 Quantum Zero-Knowledge

We construct a quantum polynomial-time universal simulator Sim that for a quantum verifier V^*, an arbitrary quantum auxiliary input ρ and an instance in the language $x \in \mathcal{L}$, simulates the output distribution of the verifier after the real interaction, $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}, \mathsf{V}^*(\rho)\rangle(x)$.

High-Level Description of Simulation. Our simulation is composed as follows. We first describe two simulators, $\mathsf{Sim}_{\mathrm{a}}$ and $\mathsf{Sim}_{\mathrm{na}}$ that try to simulate different types of transcripts, specifically, $\mathsf{Sim}_{\mathrm{a}}$ will try to simulate an aborting interaction, and $\mathsf{Sim}_{\mathrm{na}}$ will try to simulate a non-aborting interaction. By "aborting interaction" and "non-aborting interaction" we formally mean the following:

- **An aborting interaction** is one where the verifier V^* either aborts before the end of step 5, or fails to prove its WI statement in step 4.
- **A non-aborting interaction** is one that is not aborting. More precisely, a non-aborting interaction is one where the verifier did not abort before the end of step 5, and also succeeded in proving its WI statement in step 4.

Our next step will be to describe a unified simulator $\mathsf{Sim}_{\mathrm{comb}}$ that randomly chooses $b \leftarrow \{\mathrm{a}, \mathrm{na}\}$ and then uses Sim_b to simulate the interaction. We will prove that on input (x, V^*, ρ), $\mathsf{Sim}_{\mathrm{comb}}$ outputs a quantum state that is computationally indistinguishable from $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}, \mathsf{V}^*(\rho)\rangle(x)$, with the following exception: $\mathsf{Sim}_{\mathrm{comb}}$ outputs a quantum state $\widetilde{\mathsf{OUT}}$ that indistinguishable from the real verifier output $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}, \mathsf{V}^*(\rho)\rangle(x)$ conditioned on $\widetilde{\mathsf{OUT}} \neq \mathtt{Fail}$. Furthermore $\widetilde{\mathsf{OUT}} \neq \mathtt{Fail}$ with probability negligibly close to $1/2$. In other words, $\mathsf{Sim}_{\mathrm{comb}}$ is going to succeed simulating only with probability (negligibly close to) $\frac{1}{2}$.

We further show that $\mathsf{Sim}_{\mathrm{comb}}$ satisfies the required conditions for applying Watrous' quantum rewinding lemma so that the success probability can be amplified from $\approx 1/2$ to ≈ 1.

The Actual Simulators. We start by describing the above mentioned simulators.

$\mathsf{Sim}_{\mathrm{a}}(x, \mathsf{V}^*, \rho)$:

(1) **Simulation of Initial Commitments and Verifier Message:**
   (a) $\mathsf{Sim}_{\mathrm{a}}$ computes $\mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda)$ and sends to V^* the commitments $\mathsf{cmt}_1 \leftarrow \mathsf{Com}(1^\lambda, 0^{|w|})$, $\mathsf{cmt}_2 \leftarrow \mathsf{Com}(1^\lambda, \mathsf{dk})$.
   (b) V^* sends $\mathsf{pk}, \mathsf{ct}_{\mathsf{V}^*}, \widetilde{\mathsf{CC}}$.

Protocol 1

Common Input: An instance $x \in \mathcal{L} \cap \{0,1\}^\lambda$, for security parameter $\lambda \in \mathbb{N}$.

P's private input: A classical witness $w \in \mathcal{R}_{\mathcal{L}}(x)$ for x.

(1) **Prover Commitment:** P sends non-interactive commitments to the witness w and to a string of zeros in the length of an SFE secret key dk: $\mathsf{cmt}_1 \leftarrow \mathsf{Com}(1^\lambda, w)$, $\mathsf{cmt}_2 \leftarrow \mathsf{Com}(1^\lambda, 0^{|\mathsf{dk}|})$.

(2) **Extractable Commitment to Verifier Challenge:**
   (a) V computes a challenge $\beta \leftarrow \Sigma.\mathsf{V}$.
   (b) V computes $s \leftarrow \{0,1\}^\lambda$, $t \leftarrow \{0,1\}^\lambda$, $(\mathsf{pk}, \mathsf{sk}) = \mathsf{QHE.Keygen}(1^\lambda; r)$ where r is the sampled randomness for the QFHE key generation algorithm. V sends
   $$\mathsf{pk}, \ \ \mathsf{ct}_{\mathsf{V}} \leftarrow \mathsf{QHE.Enc}_{\mathsf{pk}}(t),$$
   $$\widetilde{\mathsf{CC}} \leftarrow \mathsf{Obf}\Big(\mathsf{CC}\big[\mathsf{QHE.Dec}_{\mathsf{sk}}(\cdot), s, (r, \beta)\big]\Big) \ .$$
   (c) P computes $\mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda)$ and sends $\mathsf{ct}_{\mathsf{P}} \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(0^\lambda)$.
   (d) V sends $\hat{\mathsf{ct}} \leftarrow \mathsf{SFE.Eval}\Big(\mathsf{CC}\big[\mathsf{Id}(\cdot), t, s\big], \mathsf{ct}_{\mathsf{P}}\Big)$, where $\mathsf{Id}(\cdot)$ is a circuit the on input t outputs s and $\perp$ otherwise.

(3) **Sigma Protocol Execution:**
   (a) P computes $\alpha \leftarrow \Sigma.\mathsf{P}(x, w)$ and sends α.
   (b) V sends the challenge β.

(4) **WI Proof by the Verifier:** V gives a WI proof of the following statement:
   - The transcript of the verifier so far is explainable.
   - **Or,** cmt_1 is a commitment to a non-witness $u \notin \mathcal{R}_{\mathcal{L}}(x)$.

   The witness that V uses for the proof is its randomness, that proves that the transcript is explainable.

(5) **WI Proof by the Prover:** P gives a WI proof of the following statement:
   - $x \in \mathcal{L}$.
   - **Or,** cmt_1, cmt_2 are both valid commitments and furthermore, ct_{P} is a valid SFE encryption and is encrypted with a key dk which is the content of the commitment cmt_2.

   The witness that P uses for the proof is w, that proves $x \in \mathcal{L}$.

(6) **Sigma Protocol Completion:** P sends $\gamma = \Sigma.\mathsf{P}(x, w; \alpha, \beta)$.

(7) **Acceptance:** V accepts if $\Sigma.\mathsf{V}(\alpha, \beta, \gamma) = 1$.

Figure 1: A classical constant-round zero-knowledge argument for $\mathcal{L} \in$ NP with quantum security.

(2) **Trying to get an Abort:** $\mathsf{Sim}_{\mathrm{a}}$ interacts with V^* as the honest prover P until the end of step 5 of the original protocol, with exactly 2 differences:
   - The message α at step 3a is generated by the first-message simulator $\alpha \leftarrow \Sigma.\mathsf{S}(x, 0^\lambda)$ (the simulator from the special zero-knowledge property of the sigma protocol), and not by the sigma protocol prover.

   - At step 5, the witness used to prove the WI statement is for the second statement in the OR expression (that the commitments $\mathsf{cmt}_1, \mathsf{cmt}_2$ are valid and consistent), and not the first (that $x \in \mathcal{L}$).

(3) **Simulation Verdict:** If at some point V^* aborts or fails in its WI proof, Sim_a outputs the aborting verifier's output. Otherwise, Sim_a outputs `Fail`.

$\mathsf{Sim}_\mathsf{na}(x, \mathsf{V}^*, \rho)$:

(1) **Simulation of Initial Commitments and Verifier Message:**
  (a) Sim_na computes $\mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda)$ and sends to V^* the commitments $\mathsf{cmt}_1 \leftarrow \mathsf{Com}(1^\lambda, 0^{|w|})$, $\mathsf{cmt}_2 \leftarrow \mathsf{Com}(1^\lambda, \mathsf{dk})$.
  (b) V^* sends $\mathsf{pk}, \mathsf{ct}_{\mathsf{V}^*}, \widetilde{\mathsf{CC}}$.

(2) **Non-Black-Box Extraction Attempt:**
  (a) Sim_na computes

$$r_1 \leftarrow \{0,1\}^*, \qquad \mathsf{ct}_t^{\mathsf{SFE}} = \mathsf{QHE.Eval}_\mathsf{pk}(\mathsf{SFE.Enc}_\mathsf{dk}(\cdot; r_1), \mathsf{ct}_{\mathsf{V}^*}) \ .$$

  Sim_na also encrypts $\rho^{(1)}$, the inner (quantum) state of the verifier after its first message:

$$\mathsf{ct}_{\rho^{(1)}} \leftarrow \mathsf{QHE.QEnc}_\mathsf{pk}(\rho^{(1)}) \ .$$

  (b) Sim_na performs a quantum homomorphic evaluation of the verifier's response. It computes,

$$\left(\mathsf{ct}_s^{\mathsf{SFE}}, \mathsf{ct}_{\rho^{(2)}}\right) \leftarrow \mathsf{QHE.Eval}_\mathsf{pk}\left(\mathsf{V}^*, \left(\mathsf{ct}_t^{\mathsf{SFE}}, \mathsf{ct}_{\rho^{(1)}}\right)\right) \ .$$

  (c) Sim_na computes $\mathsf{ct}_s \leftarrow \mathsf{QHE.Eval}_\mathsf{pk}\left(\mathsf{SFE.Dec}_\mathsf{dk}(\cdot), \mathsf{ct}_s^{\mathsf{SFE}}\right)$, and then computes $(r, \beta') = \widetilde{\mathsf{CC}}(\mathsf{ct}_s)$.
  (d) Sim_na checks validity: $(\mathsf{pk}', \mathsf{sk}) = \mathsf{QHE.Keygen}(1^\lambda; r)$, if $\mathsf{pk}' \neq \mathsf{pk}$ then it halts simulation and outputs $\perp$. Otherwise, Sim_na obtains the inner state of V^* by decryption: $\rho^{(2)} \leftarrow \mathsf{QHE.QDec}_\mathsf{sk}(\mathsf{ct}_{\rho^{(2)}})$. Additionally, Sim_na simulates the missing transcript (for the verifier to later prove that its messages were explainable): for the prover message at step 2c it inserts $\mathsf{ct}_t = \mathsf{SFE.Enc}_\mathsf{dk}(t; r_1)$, and for the verifier message at step 2d it inserts $\hat{\mathsf{ct}}_s = \mathsf{QHE.Dec}_\mathsf{sk}(\mathsf{ct}_s^{\mathsf{SFE}})$.

(3) **Sigma Protocol Messages Simulation:**
  (a) Sim_na executes $(\alpha, \gamma) \leftarrow \Sigma.\mathsf{S}(x, \beta')$ (the simulator from the special zero-knowledge property of the sigma protocol) and sends α to V^*.
  (b) V^* returns β.

(4) **WI Proof by the Malicious Verifier:** Sim_na takes the role of the honest prover P in the WI proof V^* gives. If V^* fails to prove the statement, the simulation fails and the output is $\perp$.

(5) **Simulation of the Prover's WI Proof and Information Reveal:** Sim_na gives V^* a WI proof using the witness that shows cmt_1, cmt_2 are both valid commitments (and that cmt_2 is a commitment to the SFE key dk used in step 2c). After the proof, Sim_na sends γ to V^*.

(6) **Simulation Verdict:** If V^* completed interaction without aborting and gave a convincing WI proof, Sim_a outputs the verifier's output. Otherwise, Sim_a outputs `Fail`.

$\mathsf{Sim}_\mathsf{comb}(x, \mathsf{V}^*, \rho)$: $\mathsf{Sim}_\mathsf{comb}$ samples $b \leftarrow \{\mathsf{a}, \mathsf{na}\}$ and executes $\mathsf{Sim}_b(x, \mathsf{V}^*, \rho)$.

$\mathsf{Sim}(x, \mathsf{V}^*, \rho)$:

(1) Generate the circuit $\mathsf{Sim}_{\mathsf{comb},x,\mathsf{V}^*}$, which is the circuit implementation of $\mathsf{Sim}_\mathsf{comb}$, with hardwired input x, V^*, that is, the only input to $\mathsf{Sim}_{\mathsf{comb},x,\mathsf{V}^*}$ is the quantum state ρ.
(2) Let R be Watrous' quantum rewinding algorithm. The output of the simulation is $\mathsf{R}(\mathsf{Sim}_{\mathsf{comb},x,\mathsf{V}^*}, \rho, \lambda)$.

Proof of Simulation Validity. We describe the simulation validity proof outline. The complete proof is in the full version. The proof that the simulated output $\mathsf{Sim}(x, \mathsf{V}^*, \rho)$ is computationally indistinguishable from $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}, \mathsf{V}^*(\rho) \rangle(x)$ is done in the following steps:

(1) **Simulating aborting interactions:** Let V_a^* be the augmented verifier that is identical to V^*, with the exception that if V^* does not abort, V_a^* outputs `Fail`. Then the output of Sim_a is indistinguishable from the output of V_a^* in a real interaction.
(2) **Simulating non-aborting interactions:** Let V_na^* be the augmented verifier that is identical to V^*, with the exception that if V^* aborts, V_na^* outputs `Fail`. Then the output of Sim_na is indistinguishable from the output of V_na^* in a real interaction.
(3) The above two statements imply:
   - $\mathsf{Sim}_\mathsf{comb} \neq$ `Fail` with probability negligibly close to $\frac{1}{2}$, for every verifier auxiliary input ρ.
   - The output of V^* in a real interaction is indistinguishable from the output of $\mathsf{Sim}_\mathsf{comb}$ conditioned on $\mathsf{Sim}_\mathsf{comb} \neq$ `Fail`.

   These in turn imply that we can use Watrous' quantum rewinding lemma in order to amplify $\mathsf{Sim}_\mathsf{comb}$ into a full-fledged simulator Sim.

3 QUANTUMLY-EXTRACTABLE CLASSICAL COMMITMENTS

In this section we show how to use any constant-round post-quantum zero-knowledge argument for NP (and additional cryptographic primitives) in order to construct a constant-round, quantumly-extractable classical commitment scheme. We start with the definition, and proceed to the construction. The proof of correctness is in the full version.

Definition 3.1 (Quantumly-Extractable Commitment). *A quantumly-extractable commitment scheme consists of three interactive PPT algorithms* $(\mathsf{Sen}, \mathsf{Rec}, \mathsf{VDcom})$ *with the following syntax.*

- $\mathsf{Sen}(1^\lambda, m)$: *The sender algorithm gets as input the public security parameter* 1^λ *and the secret message* m *to commit to.*
- $\mathsf{Rec}(1^\lambda)$: *The receiver algorithm gets only the public security parameter* 1^λ.
- *The algorithms* $\mathsf{Sen}, \mathsf{Rec}$ *interact and generate transcript* T.
- $\mathsf{VDcom}(T, m, r)$: *For a transcript, message and randomness, the decommitment verification algorithm outputs a bit.*

The scheme satisfies the following conditions.

- **Perfect Binding:** *Let $m_0, m_1, r_0, r_1 \in \{0,1\}^*$, and let T be a transcript. If $\mathsf{VDcom}(T, m_0, r_0) = \mathsf{VDcom}(T, m_1, r_1) = 1$, then $m_0 = m_1$. Accordingly, for a transcript T denote by m_T the (unique) string such that if there exist r s.t. $\mathsf{VDcom}(T, m, r) = 1$, then $m_T := m$, and $m_T := \perp$ otherwise.*
- **Computational Hiding:** *For every polynomial-size quantum receiver $\mathsf{Rec}^* = \{\mathsf{Rec}^*_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ and polynomial $\ell(\cdot)$,*

$$\{\mathsf{OUT}_{\mathsf{Rec}^*_\lambda}\langle \mathsf{Sen}(m_0), \mathsf{Rec}^*_\lambda(\rho_\lambda)\rangle(1^\lambda)\}_{\lambda, m_0, m_1} \approx_c$$

$$\{\mathsf{OUT}_{\mathsf{Rec}^*_\lambda}\langle \mathsf{Sen}(m_1), \mathsf{Rec}^*_\lambda(\rho_\lambda)\rangle(1^\lambda)\}_{\lambda, m_0, m_1} \ ,$$

where $\lambda \in \mathbb{N}$, $m_0, m_1 \in \{0,1\}^{\ell(\lambda)}$.

- **Extractability:** *There exists a quantum polynomial-time algorithm Ext s.t. for every polynomial-size quantum sender $\mathsf{Sen}^* = \{\mathsf{Sen}^*_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ outputs a quantum state σ_{Ext} and message m_{Ext}, with the following guarantee.*

$$\left\{(\sigma, m_T) \mid (T, \sigma, m_T) \leftarrow \langle \mathsf{Sen}^*_\lambda(\rho_\lambda), \mathsf{Rec}\rangle(1^\lambda)\right\}_{\lambda \in \mathbb{N}}$$

$$\approx_c \left\{(\sigma_{\mathsf{Ext}}, m_{\mathsf{Ext}}) \mid (\sigma_{\mathsf{Ext}}, m_{\mathsf{Ext}}) \leftarrow \mathsf{Ext}(1^\lambda, \mathsf{Sen}^*_\lambda, \rho_\lambda)\right\}_{\lambda \in \mathbb{N}} \ ,$$

where σ is the inner state of Sen^* after executing the interaction with Rec.

We describe the protocol between Sen and Rec in Figure 2.

Ingredients and notation:

- A non-interactive commitment scheme Com.
- A 2-message function-hiding secure function evaluation scheme (SFE.Gen, SFE.Enc, SFE.Eval, SFE.Dec).
- A constant-round post-quantum zero-knowledge argument system $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ for NP.

Decommitment Verification. On input (T, m, r) the decommitment verification algorithm VDcom deduces the security parameter λ (the security parameter is public and can be assumed to be part of the transript). It then checks two things:

- The argument that Sen gave at the last step of the transcript T is convincing (this is possible as the argument is publicly verifiable).
- The commitment $\mathsf{cmt}_{\mathsf{Sen}}$ from step 1 in the transcript T indeed decommits to m, r (i.e. $\mathsf{Com}(1^\lambda, m; r) = \mathsf{cmt}_{\mathsf{Sen}}$).

The output is 1 iff the check succeeds.

Protocol 2

Common Input: A security parameter $\lambda \in \mathbb{N}$.
Private Input of Sen: A message $m \in \{0,1\}^*$ to commit to.

1. **Commitment by** Sen**:** Sen sends a commitment to m, $\mathsf{cmt}_{\mathsf{Sen}} \leftarrow \mathsf{Com}(1^\lambda, m)$.
2. **Commitment by** Rec**:** Rec sends a commitment to 0, $\mathsf{cmt}_{\mathsf{Rec}} \leftarrow \mathsf{Com}(1^\lambda, 0)$.
3. **ZK Argument by** Rec**:** Rec uses $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ and gives a zero-knowledge argument for the statement that $\mathsf{cmt}_{\mathsf{Rec}}$ is a valid commitment to 0.
4. Sen **Challenges** Rec**:** The parties interact so that Sen can offer to send m if Rec managed to trick Sen in the ZK argument.
   (a) Rec computes $\mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda)$ and sends $\mathsf{ct}_{\mathsf{Rec}} \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(0^{\mathrm{poly}(\lambda,1)})$.
   (b) Sen sends $\hat{\mathsf{ct}} \leftarrow \mathsf{SFE.Eval}\Big(C_{1\to m}, \mathsf{ct}_{\mathsf{Rec}}\Big)$, where $C_{1\to m}$ is the (canonical) circuit that for input $r_1 \in \{0,1\}^*$ s.t. $\mathsf{cmt}_{\mathsf{Rec}} = \mathsf{Com}(1^\lambda, 1; r_1)$, outputs m, and for any other input outputs $\perp$.
5. **ZK Argument by** Sen**:** Sen uses $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ and gives a zero-knowledge argument for the statement that its transcript so far is explainable.

Figure 2: A Quantumly-Extractable Classical Commitment Scheme.

4 CONSTANT-ROUND ZERO-KNOWLEDGE QUANTUM ARGUMENTS FOR QMA

In this section we explain how the tools from previous sections imply a constant-round zero-knowledge quantum argument for QMA, that is, according to Definition 4.1. We only describe the protocol, the proofs of soundness and zero knowledge are in the full version.

Definition 4.1 (Zero-Knowledge Quantum Argument for QMA). *A quantum protocol (P, V) with an honest QPT prover P and an honest QPT verifier V for a language $\mathcal{L} \in$ **QMA** is a quantum zero-knowledge argument if it satisfies the following:*

(1) **Statistical Completeness:** *There is a polynomial $k(\cdot)$ and a negligible function $\mu(\cdot)$ s.t. for any $\lambda \in \mathbb{N}, x \in \mathcal{L} \cap \{0,1\}^\lambda$, $w \in \mathcal{R}_{\mathcal{L}}(x)$*[3]*,*

$$\Pr[\mathsf{OUT}_{\mathsf{V}}\langle \mathsf{P}(w^{\otimes k(\lambda)}), \mathsf{V}\rangle(x) = 1] \geq 1 - \mu(\lambda) \ .$$

(2) **Quantum Soundness:** *For any quantum polynomial-size prover $\mathsf{P}^* = \left\{\mathsf{P}^*_\lambda, \rho_\lambda\right\}_{\lambda \in \mathbb{N}}$, there exists a negligible function $\mu(\cdot)$ such that for any security parameter $\lambda \in \mathbb{N}$ and any $x \in \{0,1\}^\lambda \setminus \mathcal{L}$,*

$$\Pr\left[\mathsf{OUT}_{\mathsf{V}}\langle \mathsf{P}^*_\lambda(\rho_\lambda), \mathsf{V}\rangle(x) = 1\right] \leq \mu(\lambda) \ .$$

(3) **Quantum Zero Knowledge:** *There exists a quantum polynomial-time simulator Sim, such that for any quantum polynomial-size verifier $\mathsf{V}^* = \left\{\mathsf{V}^*_\lambda, \rho_\lambda\right\}_{\lambda \in \mathbb{N}}$,*

$$\{\mathsf{OUT}_{\mathsf{V}^*_\lambda}\langle \mathsf{P}(w^{\otimes k(\lambda)}), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x)\}_{\lambda, x, w} \approx_c \{\mathsf{Sim}(x, \mathsf{V}^*_\lambda, \rho_\lambda)\}_{\lambda, x, w} \ ,$$

where $\lambda \in \mathbb{N}$, $x \in \mathcal{L} \cap \{0,1\}^\lambda$, $w \in \mathcal{R}_{\mathcal{L}}(x)$.

The construction uses (post-quantum) zero-knowledge arguments for NP, quantumly-extractable commitments and a sigma protocol for QMA (defined in the full version).
We conclude with the protocol, which is described in Figure 3.

[3] For a language $\mathcal{L}$ in QMA, for an instance $x \in \mathcal{L}$ in the language, the set $\mathcal{R}_{\mathcal{L}}(x)$ is the (possily infinite) set of quantum witnesses that make the BQP verification machine accept with some overwhelming probability $1 - \mathrm{negl}(\lambda)$.

Ingredients and notation:

- A constant-round quantumly-extractable commitment scheme $(\mathsf{Sen}, \mathsf{Rec})$.
- A constant-round post-quantum zero-knowledge argument system $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ for NP.
- A quantum sigma protocol for QMA $(\Xi.\mathsf{P}, \Xi.\mathsf{V})$.

Protocol 3

Common Input: An instance $x \in \mathcal{L} \cap \{0,1\}^{\lambda}$, for security parameter $\lambda \in \mathbb{N}$.

P's private input: Polynomially many identical witnesses for x: $w^{\otimes k(\lambda)}$ s.t. $w \in \mathcal{R}_{\mathcal{L}}(x)$.

(1) **Verifier Extractable Commitment to Challenge:** V computes $\beta \leftarrow \Xi.\mathsf{V}$ and commits to it using the extractable commitment $(\mathsf{Sen}, \mathsf{Rec})$. V executes $\mathsf{Sen}(1^{\lambda}, \beta)$ and P executes $\mathsf{Rec}(1^{\lambda})$, and commitment transcript T_{Sen} is generated.

(2) **Prover Commitment:** P computes $(\alpha, \tau) \leftarrow \Xi.\mathsf{P}_1(x, w^{\otimes k(\lambda)})$ and sends α to V.

(3) **Verifier Challenge and ZK Argument:**

(a) V sends β.

(b) V proves in ZK (using the argument system $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$) that the sent β is the value inside the extractable commitment, that is, $\exists r \in \{0,1\}^*$ such that $1 = \mathsf{VDcom}(T_{\mathsf{Sen}}, \beta, r)$. If the argument was not convincing P terminates communication.

(4) **Sigma Protocol Completion:** If the proof by V was convincing then P computes $\gamma \leftarrow \Xi.\mathsf{P}_3(\beta, \tau)$ and sends γ.

(5) **Acceptance:** The verifier accepts iff $1 = \Xi.\mathsf{V}(\alpha, \beta, \gamma)$.

Figure 3: A quantum constant-round zero-knowledge argument for $\mathcal{L} \in$ QMA.

ACKNOWLEDGMENTS

We thank Zvika Brakerski for insightful discussions about Quantum Fully-Homomorphic Encryption. We also thank Venkata Koppula for advice regarding the state of the art of Compute-and-Compare Obfuscation.

REFERENCES

[AP] Prabhanjan Ananth and Rolando La Placa. Personal communication.

[ARU14] Andris Ambainis, Ansis Rosmanis, and Dominique Unruh. Quantum attacks on classical proof systems: The hardness of quantum rewinding. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 474–483. IEEE, 2014.

[Bar01] Boaz Barak. How to go beyond the black-box simulation barrier. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, pages 106–115, 2001.

[BG19] Anne Broadbent and Alex B Grilo. Zero-knowledge for qma from locally simulatable proofs. *arXiv preprint arXiv:1911.07782*, 2019.

[BGJ+13] Elette Boyle, Sanjam Garg, Abhishek Jain, Yael Tauman Kalai, and Amit Sahai. Secure computation against adaptive auxiliary information. In *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, pages 316–334, 2013.

[BJSW16] Anne Broadbent, Zhengfeng Ji, Fang Song, and John Watrous. Zero-knowledge proof systems for qma. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 31–40. IEEE, 2016.

[BKP19] Nir Bitansky, Dakshita Khurana, and Omer Paneth. Weak zero-knowledge beyond the black-box barrier. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1091–1102. ACM, 2019.

[BM84] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. Comput.*, 13(4):850–864, 1984.

[BOCG+06] Michael Ben-Or, Claude Crépeau, Daniel Gottesman, Avinatan Hassidim, and Adam Smith. Secure multiparty quantum computation with (only) a strict honest majority. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 249–260. IEEE, 2006.

[BP15] Nir Bitansky and Omer Paneth. On non-black-box simulation and the impossibility of approximate obfuscation. *SIAM J. Comput.*, 44(5):1325–1383, 2015.

[Bra18] Zvika Brakerski. Quantum fhe (almost) as secure as classical. In *Annual International Cryptology Conference*, pages 67–95. Springer, 2018.

[CFGS18] Alessandro Chiesa, Michael A. Forbes, Tom Gur, and Nicholas Spooner. Spatial isolation implies zero knowledge even in a quantum world. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 755–765, 2018.

[CLP13] Kai-Min Chung, Huijia Lin, and Rafael Pass. Constant-round concurrent zero knowledge from p-certificates. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 50–59, 2013.

[CPS16] Kai-Min Chung, Rafael Pass, and Karn Seth. Non-black-box simulation from one-way functions and applications to resettable security. *SIAM J. Comput.*, 45(2):415–458, 2016.

[DFS04] Ivan Damgård, Serge Fehr, and Louis Salvail. Zero-knowledge proofs and string commitments withstanding quantum attacks. In *Annual International Cryptology Conference*, pages 254–272. Springer, 2004.

[DGS09] Yi Deng, Vipul Goyal, and Amit Sahai. Resolving the simultaneous resettability conjecture and a new non-black-box simulation strategy. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25-27, 2009, Atlanta, Georgia, USA*, pages 251–260, 2009.

[DNS10] Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail. Secure two-party quantum evaluation of unitaries against specious adversaries. In *Annual Cryptology Conference*, pages 685–706. Springer, 2010.

[DNS12] Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail. Actively secure two-party evaluation of any quantum operation. In *Annual Cryptology Conference*, pages 794–811. Springer, 2012.

[FP96] Christopher A. Fuchs and Asher Peres. Quantum-state disturbance versus information gain: Uncertainty relations for quantum information. *Phys. Rev. A*, 53:2038–2045, Apr 1996.

[GK96a] Oded Goldreich and Ariel Kahan. How to construct constant-round zero-knowledge proof systems for np. *Journal of Cryptology*, 9(3):167–189, 1996.

[GK96b] Oded Goldreich and Hugo Krawczyk. On the composition of zero-knowledge proof systems. *SIAM J. Comput.*, 25(1):169–192, 1996.

[GKVW19] Rishab Goyal, Venkata Koppula, Satyanarayana Vusirikala, and Brent Waters. On perfect correctness in (lockable) obfuscation. 2019.

[GKW17] Rishab Goyal, Venkata Koppula, and Brent Waters. Lockable obfuscation. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 612–621. IEEE, 2017.

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

[GMW86] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to prove all np statements in zero-knowledge and a methodology of cryptographic protocol design. In *Conference on the Theory and Application of Cryptographic Techniques*, pages 171–185. Springer, 1986.

[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*, pages 218–229, 1987.

[Goy13] Vipul Goyal. Non-black-box simulation in the fully concurrent setting. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 221–230, 2013.

[GSY19] Alex Bredariol Grilo, William Slofstra, and Henry Yuen. Perfect zero knowledge for quantum multiprover interactive proofs. *Electronic Colloquium on Computational Complexity (ECCC)*, 26:86, 2019.

[HIK+11] Iftach Haitner, Yuval Ishai, Eyal Kushilevitz, Yehuda Lindell, and Erez Petrank. Black-box constructions of protocols for secure computation. *SIAM J. Comput.*, 40(2):225–266, 2011.

[HSS11] Sean Hallgren, Adam Smith, and Fang Song. Classical cryptographic protocols in a quantum world. In *Annual Cryptology Conference*, pages 411–428. Springer, 2011.

[Kob03] Hirotada Kobayashi. Non-interactive quantum perfect and statistical zero-knowledge. In *Algorithms and Computation, 14th International Symposium, ISAAC 2003, Kyoto, Japan, December 15-17, 2003, Proceedings*, pages 178–188, 2003.

[Liu06] Yi-Kai Liu. Consistency of local density matrices is qma-complete. In *Approximation, randomization, and combinatorial optimization. algorithms and techniques*, pages 438–449. Springer, 2006.

[LN11] Carolin Lunemann and Jesper Buus Nielsen. Fully simulatable quantum-secure coin-flipping and applications. In *International Conference on Cryptology in Africa*, pages 21–40. Springer, 2011.

[Mah18a] Urmila Mahadev. Classical homomorphic encryption for quantum circuits. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 332–338. IEEE, 2018.

[Mah18b] Urmila Mahadev. Classical verification of quantum computations. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 259–267, 2018.

[MHNF15] Tomoyuki Morimae, Masahito Hayashi, Harumichi Nishimura, and Keisuke Fujii. Quantum merlin-arthur with clifford arthur. *arXiv preprint arXiv:1506.06447*, 2015.

[PRS02] Manoj Prabhakaran, Alon Rosen, and Amit Sahai. Concurrent zero knowledge with logarithmic round-complexity. In *43rd Symposium on Foundations of Computer Science (FOCS 2002), 16-19 November 2002, Vancouver, BC, Canada, Proceedings*, pages 366–375, 2002.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):34:1–34:40, 2009.

[Unr12] Dominique Unruh. Quantum proofs of knowledge. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 135–152. Springer, 2012.

[Unr16a] Dominique Unruh. Collapse-binding quantum commitments without random oracles. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 166–195. Springer, 2016.

[Unr16b] Dominique Unruh. Computationally binding quantum commitments. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 497–527. Springer, 2016.

[VDGC97] Jeroen Van De Graaf and C Crepeau. *Towards a formal definition of security for quantum protocols.* Université de Montréal, 1997.

[Wat02] John Watrous. Limits on the power of quantum statistical zero-knowledge. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, pages 459–468. IEEE, 2002.

[Wat09] John Watrous. Zero-knowledge against quantum attacks. *SIAM Journal on Computing*, 39(1):25–58, 2009.

[WZ82] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. *Nature*, 299:802–803, 1982.

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