Post-quantum Zero Knowledge in Constant Rounds[*]

Nir Bitansky[†] Omri Shmueli[‡]

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.

[*] This work was supported in part by ISF grants 18/484 and 19/2137, by Len Blavatnik and the Blavatnik Family Foundation, and by the European Union Horizon 2020 Research and Innovation Program via ERC Project REACT (Grant 756482).

[†] Tel Aviv University, nirbitan@tau.ac.il. Member of the Check Point Institute of Information Security. Supported also by the Alon Young Faculty Fellowship.

[‡] Tel Aviv University, omrishmueli@mail.tau.ac.il. Supported also by the Zevulun Hammer Scholarship from the Council for Higher Education in Israel.

Contents

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.

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 $\mathcal{L} \in$ **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 $\mathcal{L} \in$ **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

[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.

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 $\mathbf{CC}[f, s, z]$ is given by a function f (represented as a circuit), a target string s in its range, and a message z; it outputs z on every input x such that $f(x) = s$, and rejects all other inputs. A corresponding obfuscator compiles any such program into a program $\widetilde{\mathbf{CC}}$ with the same functionality. In terms of security, provided that the target s has high entropy conditioned on f and z, the obfuscated program is computationally indistinguishable from a simulated dummy program, independent of (f, s, 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 s and t,
   - a secret key sk for an FHE scheme,
   - an FHE encryption $\mathsf{ct}_t = \mathsf{FHE.Enc}_{\mathsf{sk}}(t)$ of t,

- an obfuscation $\widetilde{\mathbf{CC}}$ of $\mathbf{CC}[f, s, z]$, where $z = (m, \mathsf{sk})$ and $f = \mathsf{FHE.Dec}_{\mathsf{sk}}$ is the FHE decryption circuit.

It then sends $(\mathsf{ct}_t, \widetilde{\mathbf{CC}})$ to the receiver R.

2. The receiver Rec sends a guess t'.

3. Sen rewards a successful guess: if $t = t'$, it sends back s (and otherwise $\perp$).

The described commitment protocol comes close to our objective. First, it is binding — the obfuscation $\widetilde{\mathbf{CC}}$ 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 $t' = t$ at the second message, but with negligible probability. Indeed, given only the first sender message $(\mathsf{ct}_t, \widetilde{\mathbf{CC}})$, the receiver obtains no information about s. Hence, we can invoke the CCO security and replace the obfuscation $\widetilde{\mathbf{CC}}$ 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 $(\mathsf{ct}_t, \widetilde{\mathbf{CC}})$ hides t. It follows that the third sender message is $\perp$ (rather than the target s) 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 $(\mathsf{ct}_t, \widetilde{\mathbf{CC}})$. At this point, it can use the sender's circuit Sen* to continue the emulation of Sen* *homomorphically under the encryption* ct_t. The key point is that, under the encryption, we do have t. We can (homomorphically) feed t to the sender, and obtain an encryption ct_s of s. Now, the simulator feeds ct_s to the obfuscation $\widetilde{\mathbf{CC}}$, 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 $\widetilde{\mathbf{CC}}$ and thus also ct_s 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 $t' = t$, 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 s and t,
   - a secret key sk for an FHE scheme,
   - an FHE encryption $\mathsf{ct}_t = \mathsf{FHE.Enc}_{\mathsf{sk}}(t)$ of t,
   - an obfuscation $\widetilde{\mathbf{CC}}$ of $\mathbf{CC}[f, s, z]$, where $z = (m, \mathsf{sk})$ and $f = \mathsf{FHE.Dec}_{\mathsf{sk}}$ is the FHE decryption circuit.

   It then sends $(\mathsf{ct}_t, \widetilde{\mathbf{CC}})$ to the receiver Rec.
2. The receiver Rec sends $\mathsf{ct}'_{t'}$, a guess t' encrypted using SFE. (The honest receiver sets t' arbitrarily.)
3. Sen homomorphically evaluates the function that given input t, returns s (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 t and obtaining s directly, it now sends an SFE encryption of t and obtains back an SFE encryption of s, which it can then decrypt to obtain s. Thus, as before, the homomorphic computation results in an FHE encryption of s. Indistinguishability of the simulated sender view from the real sender view now follows since the SFE encryption $\mathsf{ct}'_{t'}$ hides t'. The SFE circuit privacy guarantees that the homomorphic SFE evaluation does not leak any information about the target s, as long as the receiver does not send an SFE encryption of t.

A Malleability Problem and its Resolution. While we could argue before that a malicious receiver cannot output t in the clear, arguing that it does not output an SFE encryption of t is more tricky. In particular, the receiver might be able to somehow maul the FHE encryption ct_t to get an SFE encryption ct'_t of the value t, without actually "knowing" the value t. 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 $\widetilde{\mathbf{CC}}$ 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 $\mathsf{ct}'_{t'}$ of an arbitrary string t', 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 $\mathsf{ct}'_{t'}$ 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 $\mathsf{Sim}_{\text{comb}}$ 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 $\mathsf{Sim}_{\text{comb}}$'s guess, and thus the simulator $\mathsf{Sim}_{\text{comb}}$ 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 $\mathsf{Sim}_{\text{comb}}$ 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 $\mathsf{Sim}_{\text{comb}}$ 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 $\mathsf{Sim}_{\text{comb}}$ 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 Sim_{na} (for never-aborting verifiers) crucially relies on this; in particular, the CCO $\widetilde{\mathbf{CC}}$ and the FHE ciphertext ct_t 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] 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]. Later on, Peikert and Shiehian constructed non-interactive post-quantum zero knowledge from QLWE in the common random string model [PS19]. 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 (in the plain model).

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

[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].

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. Broadbent and Grilo [BG19] construct quantum sigma protocols for QMA, that is, 3-message protocols that are zero-knowledge but have large soundness error. Relying on their protocol and our zero-knowledge protocol and extractable commitment, we obtain a conceptually simple constant-round zero-knowledge protocol for QMA with a negligible soundness error (in a previous version of our work, we constructed such a protocol based on earlier work of [BJSW16]). Coladangelo, Vidick, and Zhang [CVZ19] construct non-interactive zero-knowledge arguments with preprocessing for QMA in the common reference string model. The challenges tackled and corresponding techniques in our work are substantially different than those in both of the above mentioned works.

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 Preliminaries

We rely on standard notions of classical Turing machines and Boolean circuits:

- A PPT algorithm is a probabilistic polynomial-time Turing machine.
- We sometimes think about PPT algorithms as polynomial-size uniform families of circuits, these are equivalent models. A polynomial-size circuit family $\mathcal{C}$ is a sequence of circuits $\mathcal{C} = \{C_\lambda\}_{\lambda\in\mathbb{N}}$, such that each circuit C_λ is of polynomial size $\lambda^{O(1)}$. We say that the family is uniform if there exists a deterministic polynomial-time algorithm M that on input 1^λ outputs C_λ.
- For a PPT algorithm M, we denote by $M(x;r)$ the output of M on input x and random coins r. For such an algorithm and any input x, we write $m \in M(x)$ to denote the fact that m is in the support of $M(x;\cdot)$.

We follow standard notions from quantum computation.

- A QPT algorithm is a quantum polynomial-time Turing machine.

- We sometimes think about QPT algorithms as polynomial-size uniform families of quantum circuits, these are equivalent models. A polynomial-size quantum circuit family $\mathcal{C}$ is a sequence of quantum circuits $\mathcal{C} = \{C_\lambda\}_{\lambda \in \mathbb{N}}$, such that each circuit C_λ is of polynomial size $\lambda^{O(1)}$. We say that the family is uniform if there exists a deterministic polynomial-time algorithm M that on input 1^λ outputs C_λ.

- An interactive algorithm M, in a two-party setting, has input divided into two registers and output divided into two registers. For the input, one register I_m is for an input message from the other party, and a second register I_a is an auxiliary input that acts as an inner state of the party. For the output, one register O_m is for a message to be sent to the other party, and another register O_a is again for auxiliary output that acts again as an inner state. For a quantum interactive algorithm M, both input and output registers are quantum.

The Adversarial Model. Throughout, efficient adversaries are modeled as quantum circuits with non-uniform quantum advice (i.e. quantum auxiliary input). Formally, *a polynomial-size adversary* $\mathsf{A}^* = \{\mathsf{A}^*_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$, consists of a polynomial-size non-uniform sequence of quantum circuits $\{\mathsf{A}^*_\lambda\}_{\lambda \in \mathbb{N}}$, and a sequence of polynomial-size mixed quantum states $\{\rho_\lambda\}_{\lambda \in \mathbb{N}}$.

For an interactive quantum adversary in a classical protocol, it can be assumed without the loss of generality that its output message register (the register containing the message to be sent to the other side, not the register containing output quantum auxiliary information) is always measured in the computational basis at the end of computation. This assumption is indeed without the loss of generality, because whenever a quantum state is sent through a classical channel then qubits decohere and are effectively measured in the computational basis.

Indistinguishability in the Quantum Setting.

- Let $f : \mathbb{N} \to [0,1]$ be a function.
  - f is negligible if for every constant $c \in \mathbb{N}$ there exists $N \in \mathbb{N}$ such that for all $n > N$, $f(n) < n^{-c}$.
  - f is noticeable if there exists $c \in \mathbb{N}, N \in \mathbb{N}$ such that for every $n \geq N$, $f(n) \geq n^{-c}$.
  - f is overwhelming if it is in the form $1 - \mu(n)$, for a negligible function μ.

- We may consider random variables over bit strings or over quantum states. This will be clear from the context.

- For two random variables X and Y supported on quantum states, quantum distinguisher circuit D with, quantum auxiliary input ρ, and $\mu \in [0,1]$, we write $X \approx_{\mathsf{D},\rho,\mu} Y$ if

$$|\Pr[\mathsf{D}(X;\rho) = 1] - \Pr[\mathsf{D}(Y;\rho) = 1]| \leq \mu.$$

- Two ensembles of random variables $\mathcal{X} = \{X_i\}_{\lambda \in \mathbb{N}, i \in I_\lambda}$, $\mathcal{Y} = \{Y_i\}_{\lambda \in \mathbb{N}, i \in I_\lambda}$ over the same set of indices $I = \uplus_{\lambda \in \mathbb{N}} I_\lambda$ are said to be *computationally indistinguishable*, denoted by $\mathcal{X} \approx_c \mathcal{Y}$, if for every polynomial-size quantum distinguisher $\mathsf{D} = \{\mathsf{D}_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ there exists a negligible function $\mu(\cdot)$ such that for all $\lambda \in \mathbb{N}, i \in I_\lambda$,

$$X_i \approx_{\mathsf{D}_\lambda, \rho_\lambda, \mu(\lambda)} Y_i \ .$$

- The trace distance between two distributions X, Y supported over quantum states, denoted $\mathrm{TD}(X, Y)$, is a generalization of statistical distance to the quantum setting and represents the maximal distinguishing advantage between two distributions supported over quantum states, by unbounded quantum algorithms. We thus say that ensembles $\mathcal{X} = \{X_i\}_{\lambda\in\mathbb{N}, i\in I_\lambda}, \mathcal{Y} = \{Y_i\}_{\lambda\in\mathbb{N}, i\in I_\lambda}$, supported over quantum states, are statistically indistinguishable (and write $\mathcal{X} \approx_s \mathcal{Y}$), if there exists a negligible function $\mu(\cdot)$ such that for all $\lambda \in \mathbb{N}, i \in I_\lambda$,
$$\mathrm{TD}\left(X_i, Y_i\right) \leq \mu(\lambda) \enspace .$$

In what follows, we introduce the cryptographic tools used in this work. By default, all algorithms are classical and efficient unless stated otherwise, and security holds against polynomial-size non-uniform quantum adversaries with quantum advice.

2.1 Interactive Protocols, Witness Indistinguishability, and Zero Knowledge

We define proof and argument systems that are secure against quantum adversaries. We start with classical protocols and proceed to define quantum protocols. In what follows, we denote by (P, V) a protocol between two parties P and V. For common input x, we denote by $\mathsf{OUT}_\mathsf{V}\langle\mathsf{P}, \mathsf{V}\rangle(x)$ the output of V in the protocol. For honest verifiers, this output will be a single bit indicating acceptance or rejection of the proof. Malicious quantum verifiers may have arbitrary quantum output (which is formally captured by the verifier outputting its inner quantum state).

Definition 2.1 (Classical Proof and Argument Systems for NP)**.** *Let* (P, V) *be a protocol with an honest PPT prover* P *and an honest PPT verifier* V *for a language* $\mathcal{L} \in$ ***NP****, satisfying:*

1. **Perfect Completeness:** *For any* $\lambda \in \mathbb{N}, x \in \mathcal{L} \cap \{0,1\}^\lambda, w \in \mathcal{R}_\mathcal{L}(x)$*,*
$$\Pr[\mathsf{OUT}_\mathsf{V}\langle\mathsf{P}(w), \mathsf{V}\rangle(x) = 1] = 1 \enspace .$$
2. **Soundness:** *The protocol satisfies one of the following.*
   - **Computational Soundness:** *For any quantum polynomial-size prover* $\mathsf{P}^* = \{\mathsf{P}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$*, there exists a negligible function* $\mu(\cdot)$ *such that for any security parameter* $\lambda \in \mathbb{N}$ *and any* $x \in \{0,1\}^\lambda \setminus \mathcal{L}$*,*
$$\Pr\left[\mathsf{OUT}_\mathsf{V}\langle\mathsf{P}^*_\lambda(\rho_\lambda), \mathsf{V}\rangle(x) = 1\right] \leq \mu(\lambda) \enspace .$$
     A protocol with computational soundness is called an argument.
   - **Statistical Soundness:** *There exists a negligible function* $\mu(\cdot)$*, such that for any (unbounded) prover* P^**, any security parameter* $\lambda \in \mathbb{N}$*, and any* $x \in \{0,1\}^\lambda \setminus \mathcal{L}$*,*
$$\Pr\left[\mathsf{OUT}_\mathsf{V}\langle\mathsf{P}^*, \mathsf{V}\rangle(x) = 1\right] \leq \mu(\lambda) \enspace .$$
     A protocol with statistical soundness is called a proof.

Definition 2.2 (Quantum Proof and Argument Systems for QMA)**.** *Let* (P, V) *be a quantum protocol with an honest QPT prover* P *and an honest QPT verifier* V *for a language* $\mathcal{L} \in$ ***QMA****, satisfying:*

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]*,*
$$\Pr[\mathsf{OUT}_\mathsf{V}\langle\mathsf{P}(w^{\otimes k(\lambda)}), \mathsf{V}\rangle(x) = 1] \geq 1 - \mu(\lambda) \enspace .$$
2. **Soundness:** *As in Definition 2.1.*

[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 $1 - \mathrm{negl}(\lambda)$.

2.1.1 Witness Indistinguishability

We rely on classical constant-round (public-coin) proof systems for NP that are witness-indistinguishable; that is, proofs that use different witnesses (for the same statement) are computationally indistinguishable (for quantum attackers).

Definition 2.3 (WI Proof System for NP)**.** *A classical protocol proof system* (P,V) *for a language* $\mathcal{L} \in$ ***NP*** *(as in Definition 2.1) is witness-indistinguishable if it satisfies:*

Witness Indistinguishability: *For every quantum polynomial-size verifier* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda, \rho_\lambda\}_\lambda$*,*

$$\{\mathsf{OUT}_{\mathsf{V}^*_\lambda}\langle \mathsf{P}(w_0), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x)\}_{\lambda,x,w_0,w_1} \approx_c \{\mathsf{OUT}_{\mathsf{V}^*_\lambda}\langle \mathsf{P}(w_1), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x)\}_{\lambda,x,w_0,w_1} \ ,$$

where $\lambda \in \mathbb{N}, x \in \mathcal{L} \cap \{0,1\}^\lambda$*, and* $w_0, w_1 \in \mathcal{R}_\mathcal{L}(x)$ *are witnesses for* x*.*

Instantiations. 3-message, public-coin classical proof systems with WI follow from classical zero-knowledge proof systems such as the parallel repetition of the 3-coloring protocol [GMW91], which is in turn based on non-interactive perfectly-binding commitments. For the proof system to be WI against quantum attacks, we need the non-interactive commitments to be computationally hiding against quantum adversaries, which can be instantiated for example from QLWE.

2.1.2 Sigma Protocols

We use the abstraction of *Sigma Protocols*, which are public-coin three-message proof systems with a special zero knowledge guarantee. We define both classical and quantum Sigma Protocols.

Definition 2.4 (Classical Sigma Protocol for NP)**.** *A classical sigma protocol for* $\mathcal{L} \in$ ***NP*** *is a classical proof system* $(\Sigma.\mathsf{P}, \Sigma.\mathsf{V})$ *(as in Definition 2.1) with 3 messages and the following syntax.*

- $(\alpha, \tau) \leftarrow \Sigma.\mathsf{P}_1(x, w)$: *Given an instance* $x \in \mathcal{L}$ *and a witness* $w \in \mathcal{R}_\mathcal{L}(x)$*, the first prover execution outputs a public message* α *for* $\Sigma.\mathsf{V}$ *and a private inner state* τ*.*
- $\beta \leftarrow \Sigma.\mathsf{V}(x)$: *The verifier simply outputs a string of* $\mathrm{poly}(|x|)$ *random bits.*
- $\gamma \leftarrow \Sigma.\mathsf{P}_3(\beta, \tau)$: *Given the verifier's string* β *and the private state* τ*, the prover outputs a response* γ*.*

The protocol satisfies the following.

Special Zero-Knowledge: *There exists a PPT simulator* $\Sigma.\mathsf{S}$ *such that,*

$$\{(\alpha,\gamma) \mid (\alpha,\tau) \leftarrow \Sigma.\mathsf{P}_1(x,w), \gamma \leftarrow \Sigma.\mathsf{P}_3(\beta,\tau)\}_{\lambda,x,w,\beta} \approx_c \{(\alpha,\gamma) \mid (\alpha,\gamma) \leftarrow \Sigma.\mathsf{S}(x,\beta)\}_{\lambda,x,w,\beta} \ ,$$

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

The next claim follows directly from the special zero-knowledge requirement, and will be used throughout.

Claim 2.1 (First-Message Indistinguishability, [BKP18], Claim 8.1)**.** *In every* Σ *protocol:*

$$\{\alpha \mid (\alpha,\tau) \leftarrow \Sigma.\mathsf{P}_1(x,w)\}_{\lambda,x,w,\beta} \approx_c \left\{\alpha \mid (\alpha,\gamma) \leftarrow \Sigma.\mathsf{S}(x, 0^{|\beta|})\right\}_{\lambda,x,w,\beta} \ ,$$

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

Instantiations. Like witness-indistinguishable proofs, Sigma protocols are known to follow from the parallel repetition of the 3-coloring protocol [GMW91]. For the protocol to have special zero knowledge against quantum attacks, we need the non-interactive commitment α to be computationally hiding against quantum adversaries, which can be instantiated for example from QLWE.

Definition 2.5 (Quantum Sigma Protocol for QMA)**.** *A quantum sigma protocol for $\mathcal{L} \in$ **QMA** is a quantum proof system* $(\Xi.\mathsf{P}, \Xi.\mathsf{V})$ *(as in Definition 2.2) with 3 messages and the following syntax.*

- $(\alpha, \tau) \leftarrow \Xi.\mathsf{P}_1(x, w^{\otimes k(\lambda)})$: *Given an instance $x \in \mathcal{L} \cap \{0,1\}^\lambda$ and $k(\lambda)$ witnesses $w \in \mathcal{R}_\mathcal{L}(x)$ (for a polynomial $k(\cdot)$), the first prover execution outputs a public message α for $\Xi.\mathsf{V}$ and a private inner state τ.*

- $\beta \leftarrow \Xi.\mathsf{V}(x)$: *The verifier simply outputs a string of* $\mathrm{poly}(|x|)$ *random bits.*

- $\gamma \leftarrow \Xi.\mathsf{P}_3(\beta, \tau)$: *Given the verifier's string β and the private state τ, the prover outputs a response γ.*

The protocol satisfies the following.

Special Zero-Knowledge: *There exists a QPT simulator* $\Xi.\mathsf{Sim}$ *such that,*

$$\left\{(\alpha,\gamma) \mid (\alpha,\tau) \leftarrow \Xi.\mathsf{P}_1(x, w^{\otimes k(\lambda)}), \gamma \leftarrow \Xi.\mathsf{P}_3(\beta,\tau)\right\}_{\lambda,x,w,\beta} \approx_c \{(\alpha,\gamma) \mid (\alpha,\gamma) \leftarrow \Xi.\mathsf{Sim}(x,\beta)\}_{\lambda,x,w,\beta} \quad ,$$

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

Instantiations. Quantum sigma protocols follow from the parallel repetition of the 3-message quantum zero-knowledge protocols of [BG19] for QMA[4].

2.1.3 Quantum Zero-Knowledge Protocols

We next define post-quantum zero-knowledge classical protocols and zero-knowledge quantum protocols.

Definition 2.6 (Post-Quantum Zero-Knowledge Classical Protocol)**.** *Let* (P, V) *be a classical protocol (argument or proof) for a language $\mathcal{L} \in$ **NP** as in Definition 2.1. The protocol is quantum zero-knowledge if it satisfies:*

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

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

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.*

Definition 2.7 (Zero-Knowledge Quantum Protocol)**.** *Let* (P, V) *be a quantum protocol (argument or proof) for a language $\mathcal{L} \in$ **QMA** as in Definition 2.2, where the prover uses $k(\lambda)$ copies of a witness. The protocol is quantum zero-knowledge if it satisfies:*

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

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

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

[4] The authors in [BG19] use the name "sigma protocols" differently then in this work. Specifically, [BG19] call their 3-message protocols, that are zero-knowledge but have large soundness error, "sigma protocols". In this work we call the parallel repetition of such protocols (which have amplified soundness but weakened zero knowledge) "sigma protocols".

2.2 Additional Tools

2.2.1 Compute-and-Compare Obfuscation

We define compute-and-compare (CC) circuits and obfuscators for CC circuits.

Definition 2.8 (Compute-and-Compare Circuit)**.** *Let $f : \{0,1\}^n \to \{0,1\}^\lambda$ be a circuit, and let $u \in \{0,1\}^\lambda, z \in \{0,1\}^*$ be strings. Then $\mathbf{CC}[f,u,z](x)$ is a circuit that returns z if $f(x) = y$, and $\bot$ otherwise. $\mathbf{CC}[f,u,z]$ has a canonical description from which f, u, and z can be read.*

We now define compute-and-compare (CC) obfuscators (with perfect correctness). In what follows Obf is a PPT algorithm that takes as input a CC circuit $\mathbf{CC}[f,u,z]$ and outputs a new circuit $\widetilde{\mathbf{CC}}$.

Definition 2.9 (CC obfuscator)**.** *A PPT algorithm Obf is a compute-and-compare obfuscator if it satisfies:*

1. **Perfect Correctness:** *For any circuit $f : \{0,1\}^n \to \{0,1\}^\lambda$, $u \in \{0,1\}^\lambda$ and $z \in \{0,1\}^*$,*

$$\Pr\left[\forall x \in \{0,1\}^n : \widetilde{\mathbf{CC}}(x) = \mathbf{CC}[f,u,z](x) \;\middle|\; \widetilde{\mathbf{CC}} \leftarrow \mathsf{Obf}(\mathbf{CC}[f,u,z])\right] = 1 \enspace .$$

2. **Simulation:** *There exists a PPT simulator Sim such that for every two polynomials $\ell_1(\cdot)$, $\ell_2(\cdot)$,*

$$\{\widetilde{\mathbf{CC}} \mid u \leftarrow \{0,1\}^\lambda, \widetilde{\mathbf{CC}} \leftarrow \mathsf{Obf}(\mathbf{CC}[f,u,z])\}_{\lambda,f,z} \approx_c \{\mathsf{Sim}(1^{\ell_1(\lambda)}, 1^{\ell_2(\lambda)}, 1^\lambda)\}_{\lambda,f,z} \enspace ,$$

where $\lambda \in \mathbb{N}$, $f : \{0,1\}^n \to \{0,1\}^\lambda$ is a $\ell_1(\lambda)$-size circuit, $z \in \{0,1\}^{\ell_2(\lambda)}$.

Instantiations. Compute-and-compare obfuscators with almost-perfect correctness are constructed in [GKW17, WZ17] based on QLWE. CC obfuscators with perfect correctness are constructed [GKVW19] by Goyal, Koppula, Vusirikala and Waters, also based on QLWE.

2.2.2 Non-Interactive Commitments

We define non-interactive commitment schemes.

Definition 2.10 (Non-Interactive Commitment)**.** *A non-interactive commitment scheme is given by a PPT algorithm $\mathsf{Com}(\cdot)$ with the following syntax:*

- $\mathsf{cmt} \leftarrow \mathsf{Com}(1^\lambda, x)$: *A randomized algorithm that takes as input a security parameter 1^λ and input $x \in \{0,1\}^*$, and outputs a commitment* cmt.

The commitment algorithm satisfies:

1. **Perfect Binding:** *For any $\lambda_0, \lambda_1 \in \mathbb{N}, x_0, x_1, r_0, r_1 \in \{0,1\}^*$, $\mathsf{Com}(1^{\lambda_0}, x_0; r_0) = \mathsf{Com}(1^{\lambda_1}, x_1; r_1)$ implies $x_0 = x_1$.*

2. **Computational Hiding:** *For any polynomial $\ell(\cdot)$,*

$$\{\mathsf{Com}(1^\lambda, x_0)\}_{\lambda,x_0,x_1} \approx_c \{\mathsf{Com}(1^\lambda, x_1)\}_{\lambda,x_0,x_1} \enspace ,$$

where $\lambda \in \mathbb{N}$, $x_0, x_1 \in \{0,1\}^{\ell(\lambda)}$.

Instantiations. The above non-interactive commitments are known based on various standard assumptions, including QLWE [GHKW17, LS19].

2.2.3 Quantum Fully Homomorphic Encryption

We rely on quantum fully homomorphic encryption, specifically, a scheme where a classical input can be encrypted classically and a quantum input quantumly. The formal definition follows.

Definition 2.11 (Quantum Fully-Homomorphic Encryption)**.** *A quantum fully homomorphic encryption scheme is given by six algorithms* ($\mathsf{QHE.Keygen}$, $\mathsf{QHE.Enc}$, $\mathsf{QHE.QEnc}$, $\mathsf{QHE.Dec}$, $\mathsf{QHE.QDec}$, $\mathsf{QHE.Eval}$) *with the following syntax:*

- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{QHE.Keygen}(1^\lambda)$: *A PPT algorithm that given a security parameter 1^λ, samples a classical public key* pk *and a classical secret key* sk.
- $\mathsf{ct} \leftarrow \mathsf{QHE.Enc}_{\mathsf{pk}}(x)$: *A PPT algorithm that takes as input a classical string $x \in \{0,1\}^*$ and outputs a classical ciphertext* ct.
- $|\phi\rangle \leftarrow \mathsf{QHE.QEnc}_{\mathsf{pk}}(|\psi\rangle)$: *A QPT algorithm that takes as input a quantum state $|\psi\rangle$ and outputs a quantum ciphertext $|\phi\rangle$.*
- $x \leftarrow \mathsf{QHE.Dec}_{\mathsf{sk}}(\mathsf{ct})$: *A PPT algorithm that takes as input a classical ciphertext* ct *and outputs a string x.*
- $|\psi\rangle \leftarrow \mathsf{QHE.QDec}_{\mathsf{sk}}(|\phi\rangle)$: *A QPT algorithm that takes as input a quantum ciphertext $|\phi\rangle$ and outputs a quantum state $|\psi\rangle$.*
- $|\hat{\phi}\rangle \leftarrow \mathsf{QHE.Eval}_{\mathsf{pk}}(C, \mathsf{ct}, |\phi\rangle)$: *A QPT algorithm that takes as input a general quantum circuit C, a classical ciphertext* ct *and a quantm ciphertext $|\phi\rangle$ and outputs an evaluated quantum ciphertext $|\hat{\phi}\rangle$*

The scheme satisfies the following.

- **Quantum Semantic Security:** *For every polynomial $\ell(\cdot)$,*

$$\left\{ (\mathsf{ct}, |\phi\rangle) \;\middle|\; \begin{array}{l} (\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{QHE.Keygen}(1^\lambda), \\ \mathsf{ct} \leftarrow \mathsf{QHE.Enc}_{\mathsf{pk}}(x_0), \\ |\phi\rangle \leftarrow \mathsf{QHE.QEnc}_{\mathsf{pk}}(|\psi_0\rangle) \end{array} \right\}_{\lambda, x_0, |\psi_0\rangle, x_1, |\psi_1\rangle} \approx_c$$

$$\left\{ (\mathsf{ct}, |\phi\rangle) \;\middle|\; \begin{array}{l} (\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{QHE.Keygen}(1^\lambda), \\ \mathsf{ct} \leftarrow \mathsf{QHE.Enc}_{\mathsf{pk}}(x_1), \\ |\phi\rangle \leftarrow \mathsf{QHE.QEnc}_{\mathsf{pk}}(|\psi_1\rangle) \end{array} \right\}_{\lambda, x_0, |\psi_0\rangle, x_1, |\psi_1\rangle} ,$$

  where $\lambda \in \mathbb{N}$, $x_0, x_1 \in \{0,1\}^{\ell(\lambda)}$ and $|\psi_0\rangle$, $|\psi_1\rangle$ are $\ell(\lambda)$-qubit states.

- **Compactness:** *There exists a polynomial* $\mathrm{poly}(\cdot)$ *s.t. for every quantum circuit C with ℓ output qubits and an enryption of an input for C, the output size of the evaluation algorithm is $\ell \cdot \mathrm{poly}(\lambda)$, where λ is the security parameter of the scheme.*

- **Measurement-Preserving Homomorphism:** *For every polynomial $s(\cdot)$ there exists a neligible function* $\mathrm{negl}(\cdot)$ *such that for every $\lambda \in \mathbb{N}$, size-$s(\lambda)$ quantum circuit C, input $(x, |\psi\rangle)$ for C which is comprised of a classical string x and quantum state $|\psi\rangle$, subset M of the output qubits of C, public and secret key pair* $(\mathsf{pk}, \mathsf{sk}) \in \mathsf{QHE.Keygen}(1^\lambda)$ *and randomness strings $(r_x, r_{|\psi\rangle})$:*

$$\mathrm{TD}\left(D_0, D_1\right) \leq \mathrm{negl}(\lambda) \ ,$$

  where D_0, D_1 are the distributions which are defined as follows:

- D_0 : *Compute* $|\psi'\rangle \leftarrow C(x, |\psi\rangle)$*, measure the subset of qubits of* $|\psi'\rangle$ *which are in* M *and output the obtained state.*
- D_1 :
  - $*$ *Encrypt* $\mathsf{ct} = \mathsf{QHE.Enc}_{\mathsf{pk}}(x; r_x)$, $|\phi\rangle = \mathsf{QHE.QEnc}_{\mathsf{pk}}(|\psi\rangle; r_{|\psi\rangle})$.
  - $*$ *Evaluate* $|\hat{\phi}\rangle \leftarrow \mathsf{QHE.Eval}_{\mathsf{pk}}(C, \mathsf{ct}, |\phi\rangle)$.
  - $*$ *Measure the* $|M|$ *packets of qubits that correspond to the output qubits in* M *(by compactness, each packet is exactly of size* $\mathrm{poly}(\lambda)$*).*
  - $*$ *Decrypt the measured* $|M|$ *packets with* $\mathsf{QHE.Dec}_{\mathsf{sk}}(\cdot)$*, and decrypt the rest of the qubits with* $\mathsf{QHE.QDec}_{\mathsf{sk}}(\cdot)$*. Output the obtained state.*

Instantiations. Mahadev [Mah18a] shows how to build quantum FHE based on super-polynomial QLWE modulus and a circular security assumption with respect to a secret key and an additional trapdoor information. Brakerski [Bra18] subsequently shows how to construct quantum FHE based on polynomial QLWE modulus and a circular security assumption (analogous to the assumptions required for multi-key FHE in the classical setting). The above definition is more specific then the standard definition of QFHE. Specifically, *measurement-preservation* and (statistical) correctness for *every* triplet $(\mathsf{pk}, \mathsf{sk}, r)$ of public and secret keys and randomness r for the encryption algorithm, is not an explicit part of the standard definition. The construction of Brakerski satisfies this more general definition. This follows readily from the main Theorem (4.1) in [Bra18].

2.2.4 Function-Hiding Secure Function Evaluation

We define two-message function evaluation protocols with statistical circuit privacy and quantum input privacy.

Definition 2.12 (2-Message Function Hiding SFE)**.** *A two-message secure function evaluation protocol* $(\mathsf{SFE.Gen}, \mathsf{SFE.Enc}, \mathsf{SFE.Eval}, \mathsf{SFE.Dec})$ *has the following syntax:*

- $\mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda)$: *a probabilistic algorithm that takes a security parameter* 1^λ *and outputs a secret key* dk.

- $\mathsf{ct} \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(x)$: *a probabilistic algorithm that takes a string* $x \in \{0,1\}^*$*, and outputs a ciphertext* ct.

- $\hat{\mathsf{ct}} \leftarrow \mathsf{SFE.Eval}(C, \mathsf{ct})$: *a probabilistic algorithm that takes a (classical) circuit* C *and a ciphertext* ct *and outputs an evaluated ciphertext* $\hat{\mathsf{ct}}$.

- $\hat{x} = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$: *a deterministic algorithm that takes a ciphertext* $\hat{\mathsf{ct}}$ *and outputs a string* $\hat{x}$.

The scheme satisfies the following.

- **Perfect Correctness:** *For any polynomial* $s(\cdot)$*, for any* $\lambda \in \mathbb{N}$*, size-*$s(\lambda)$ *circuit* C *and input* x *for* C*,*
$$\Pr\left[\mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}}) = C(x) \;\middle|\; \begin{array}{l} \mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda), \\ \mathsf{ct} \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(x), \\ \hat{\mathsf{ct}} \leftarrow \mathsf{SFE.Eval}(C, \mathsf{ct}) \end{array}\right] = 1 \ .$$

- **Quantum Input Privacy:** *For every polynomial* $\ell(\cdot)$*,*
$$\left\{\mathsf{ct} \;\middle|\; \begin{array}{l} \mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda), \\ \mathsf{ct} \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(x_0) \end{array}\right\}_{\lambda, x_0, x_1} \approx_c \left\{\mathsf{ct} \;\middle|\; \begin{array}{l} \mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda), \\ \mathsf{ct} \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(x_1) \end{array}\right\}_{\lambda, x_0, x_1} ,$$
where $\lambda \in \mathbb{N}$ *and* $x_0, x_1 \in \{0,1\}^{\ell(\lambda)}$.

- **Statistical Circuit Privacy:** *There exist unbounded algorithms, probabilistic* Sim *and deterministic* Ext *such that:*
  - *For every* $x \in \{0,1\}^*$, $\mathsf{ct} \in \mathsf{SFE.Enc}(x)$, *the extractor outputs* $\mathsf{Ext}(\mathsf{ct}) = x$.
  - *For any polynomial* $s(\cdot)$,

$$\{\mathsf{SFE.Eval}(C, \mathsf{ct}^*)\}_{\lambda, C, \mathsf{ct}^*} \approx_s \{\mathsf{Sim}(\ 1^\lambda, C(\mathsf{Ext}(1^\lambda, \mathsf{ct}^*))\ )\}_{\lambda, C, \mathsf{ct}^*} \ ,$$

where $\lambda \in \mathbb{N}$, C *is a* $s(\lambda)$*-size circuit, and* $\mathsf{ct}^* \in \{0,1\}^*$.

The next claim follows directly from the circuit privacy property, and will be used throughout the analysis.

Claim 2.2 (Evaluations of Agreeing Circuits are Statistically Close)**.** *For any polynomial* $s(\cdot)$,

$$\{\mathsf{SFE.Eval}(C_0, \mathsf{ct}^*)\}_{\lambda, C_0, C_1, \mathsf{ct}} \approx_s \{\mathsf{SFE.Eval}(C_1, \mathsf{ct}^*)\}_{\lambda, C_0, C_1, \mathsf{ct}} \ ,$$

where $\lambda \in \mathbb{N}$, C_0, C_1 *are two* $s(\lambda)$*-size functionally-equivalent circuits, and* $\mathsf{ct}^* \in \{0,1\}^*$.

Instantiations. Such secure function evaluation schemes are known based on QLWE [OPCPC14, BD18].

2.2.5 Quantum Rewinding Lemma

We use Lemma 9 from [Wat09], which constructs a quantum algorithm for amplifying the success probability of quantum sampler circuits under some conditions.

Lemma 2.1 (Lemma 9, [Wat09])**.** *There is a quantum algorithm* R *that gets as input:*

- *A general quantum circuit* Q *with* n *input qubits that outputs a classical bit* b *and an additional* m *output qubits.*

- *An* n*-qubit state* $|\psi\rangle$.

- *A number* $t \in \mathbb{N}$.

R *executes in time* $t \cdot \mathrm{poly}(|\mathsf{Q}|)$ *and outputs a distribution over* m*-qubit states* $D_\psi := \mathsf{R}(\mathsf{Q}, |\psi\rangle, t)$ *with the following guarantees.*

For an n*-qubit state* $|\psi\rangle$*, denote by* Q_ψ *the conditional distribution of the output distribution* $\mathsf{Q}(|\psi\rangle)$*, conditioned on* $b = 0$*, and denote by* $p(\psi)$ *the probability that* $b = 0$*. If there exist* $p_0, q \in (0,1)$, $\varepsilon \in \left(0, \frac{1}{2}\right)$ *such that:*

- *Amplification executes for enough time:* $t \geq \frac{\log(1/\varepsilon)}{4 \cdot p_0(1-p_0)}$,

- *There is some minimal probability that* $b = 0$*: For every* n*-qubit state* $|\psi\rangle$, $p_0 \leq p(\psi)$,

- $p(\psi)$ *is input-independant, up to* ε *distance: For every* n*-qubit state* $|\psi\rangle$, $|p(\psi) - q| < \varepsilon$*, and*

- q *is closer to* $\frac{1}{2}$*:* $p_0(1-p_0) \leq q(1-q)$,

then for every n*-qubit state* $|\psi\rangle$,

$$\mathrm{TD}\Big(\mathsf{Q}_\psi, D_\psi\Big) \leq 4\sqrt{\varepsilon} \frac{\log(1/\varepsilon)}{p_0(1-p_0)} \ .$$

The exact wording in the Lemma from [Wat09] differs from the above in two manners. First, the original lemma states that for each circuit Q there exists an amplified circuit Q', but actually the proof of the Lemma proves that there is an algorithm R that on input Q, executes an amplified version of Q (and thus the circuit implementation of $\mathsf{R}(\mathsf{Q})$ can be thought of as Q'). Second, the original lemma deals with unitary quantum circuits i.e. Q contains no measurement gates. By standard quantum circuit purification, it follows that the above formulation is equivalent to the analogous statement that includes only unitary circuits.

3 Constant-Round Zero-Knowledge Arguments for NP

In this section we construct a classical argument system for an arbitrary NP language $\mathcal{L}$, with a constant number of rounds, quantum soundness and quantum zero-knowledge (according to Definition 2.6).

Ingredients and notation:

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

We describe the protocol in Figure 1.

3.1 Quantum Soundness

Proposition 3.1 (The Protocol is Sound)**.** *Let* V *be the verifier from Protocol 1. For any quantum polynomial-size prover* $\mathsf{P}^* = \{\mathsf{P}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$*, there exists a negligible function* $\mu(\cdot)$ *such that for any security parameter* $\lambda \in \mathbb{N}$ *and any* $x \in \{0,1\}^\lambda \setminus \mathcal{L}$*,*

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

Proof. Let $\mathsf{P}^* = \{\mathsf{P}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ a polynomial-size quantum prover and let $x = \{x_\lambda\}_{\lambda\in\mathbb{N}}$ be a sequence such that $\forall \lambda \in \mathbb{N} : x_\lambda \in \{0,1\}^\lambda \setminus \mathcal{L}$. We prove soundness by a hybrid argument. We consider a series of hybrid processes with output over $\{0,1\}$, starting from $\mathsf{OUT}_\mathsf{V}\langle \mathsf{P}^*(\rho), \mathsf{V}\rangle(x)$ the output distribution of V in the interaction with P^*. The proof will show that the probability to output 1 is negligible, which proves the soundness of the protocol.

We assume without the loss of generality that the first prover message is deterministic, and that the commitments $\mathsf{cmt}_1, \mathsf{cmt}_2$ it sends are both valid commitments and furthermore, there is some SFE secret key $\mathsf{dk} \in \mathsf{SFE.Gen}(1^\lambda)$ such that $\mathsf{cmt}_2 \in \mathsf{Com}(1^\lambda, \mathsf{dk})$. First note that if the above property is false, then the whole WI statement of the prover is false (because the first statement in P^*'s OR statement, that claims $x \in \mathcal{L}$, is always false in the case of a cheating prover).

This assumption is without the loss of generality because we can consider a new prover that chooses the first message (and quantum inner state at the end of this message) as the message that maximizes the probability that V outputs 1. If this message is such that $\mathsf{cmt}_1, \mathsf{cmt}_2$ are not consistent with the prover's WI statement, then by the soundness of the proof that P^* gives, with overwhelming probability V outputs 0 and soundness already holds.

As a final note, observe that because cmt_1, cmt_2 are consistent with the prover's WI statement, cmt_1 is necessarily a commitment to a non-witness $u \notin \mathcal{R}_\mathcal{L}(x)$, and denote by r_u a string s.t. $\mathsf{cmt}_1 = \mathsf{Com}(1^\lambda, u; r_u)$.

Define the following hybrid distributions.

- Hyb_0 : The output distribution of $\mathsf{OUT}_\mathsf{V}\langle \mathsf{P}^*(\rho), \mathsf{V}\rangle(x)$.

- Hyb_1 : This hybrid process is identical to Hyb_0, with the exception that in step 4 (verifier WI), V uses the information (u, r_u) as witness for its WI statement, instead of the witness that shows its transcript is explainable.

- Hyb_2 : This hybrid process is identical to Hyb_1, with the exception that V obtains dk, and when it gets the prover message ct_P in step 2c, it performs the following check: If $t = \mathsf{SFE.Dec}_\mathsf{dk}(\mathsf{ct}_\mathsf{P})$ then the process halts and outputs $\bot$, otherwise the interaction carries on regularly.

- Hyb_3 : This hybrid process is identical to Hyb_2, except that in step 2d when the verifier responds with an SFE evaluation, instead of performing an SFE evaluation of $\mathbf{CC}[\mathsf{Id}(\cdot), t, s]$, the verifier performs an SFE evaluation of $C_\bot$, a circuit that always outputs $\bot$.

- Hyb_4 : This hybrid process is identical to Hyb_3, except that the verifier does not perform the check at step 2c like described in Hyb_2.

- Hyb_5 : This hybrid process is identical to Hyb_4, except that in step 2b where V sends its first message, the reward value of the CC program $\widetilde{\mathbf{CC}}$ it uses is $(r, 0^{|\beta|})$ instead of (r, β).

We now explain why each consecutive pair of the distributions above are statistically indistinguishable (recall that for a pair of distributions over a single bit, they are statistically indistinguishable iff they are computationally indistinguishable). We will then use the last process Hyb_5 to show that soundness follows from the soundness of the sigma protocol $(\Sigma.\mathsf{P}, \Sigma.\mathsf{V})$.

- $\mathsf{Hyb}_0 \approx_s \mathsf{Hyb}_1$: Follows from the witness indistinguishability property of the WI proof that the verifier gives.

- $\mathsf{Hyb}_1 \approx_s \mathsf{Hyb}_2$: Follows from Claim 3.1, which says that the probability that ct_P is an encryption of (the correct) t with the secret key dk (that is inside cmt_2) is negligible, and thus the erasure of such cases can't be noticed by a distinguisher.

- $\mathsf{Hyb}_2 \approx_s \mathsf{Hyb}_3$: As a basic explanation, this indistinguishability follows from the combination of the circuit privacy property of the SFE and the soundness of the WI proof that P^* gives.

  As a fuller explanation, assume toward contradiction there's a distinguisher D^* that tells the difference between the two distributions, and by an averaging argument, consider the transcript (and inner quantum state of P^*) generated at the end of step 2c (where P^* sends ct_P), which maximizes D^*'s distinguisability adventage - other than the prover's ciphertext ct_P, this transcript fixes t, s, which in turn fix the circuit $\mathbf{CC}\big[\mathsf{Id}(\cdot), t, s\big]$. We now consider three cases, and explain why we get a contradiction in each of them.

  1. $\mathsf{ct}_\mathsf{P} \in \mathsf{SFE.Enc}_\mathsf{dk}(t)$: In this case, no matter what will be generated next in the transcript, the output will be $\bot$ (by the check described in Hyb_2), thus it is impossible to distinguish the outputs of the two processes and we get a contradiction.
  2. $\exists y \in \{0,1\}^\lambda \setminus \{t\}$ s.t. $\mathsf{ct}_\mathsf{P} \in \mathsf{SFE.Enc}_\mathsf{dk}(y)$: In this case, $\bot = \mathbf{CC}\big[\mathsf{Id}(\cdot), t, s\big](y)$ and thus we get a contradiction by using the circuit privacy property of the SFE.
  3. Else: In that case, either ct_P is a ciphertext encrypted with some other SFE key dk', or it is not a valid ciphertext at all and in any case, it is not a valid ciphertext encrypted with the secret key dk. In that case, the WI statement of the prover is necessarily false, and thus a 1 output happens with at most negligible probability in both cases (by the soundness of the WI proof of P^*), thus the statistical distance between them is at most negligible, in contradiction.

- $\mathsf{Hyb}_3 \approx_s \mathsf{Hyb}_4$: Follows from the same reasoning as in the indistinguishability $\mathsf{Hyb}_1 \approx_s \mathsf{Hyb}_2$.

- $\mathsf{Hyb}_4 \approx_s \mathsf{Hyb}_5$: Follows from the simulation property (obfuscation security) of the CC obfuscation scheme.

Now, assume toward contradiction that P^* succeeds in making the verifier accept with some noticeable probability $\varepsilon(\lambda)$, that is, the probability for the output 1 in Hyb_0 is noticeable. $\mathsf{Hyb}_0 \approx_s \mathsf{Hyb}_5$, and thus the probability for the output 1 in Hyb_5 is also noticeable. Finally, we get a contradiction to the soundness of the sigma protocol $(\Sigma.\mathsf{P}, \Sigma.\mathsf{V})$, by using the prover sigma protocol messages from steps 3a, 6 as messages to convince a sigma protocol verifier $\Sigma.\mathsf{V}$. Since the probability that the verifier V is convinced in Hyb_5 is noticeable, and such verifier is convinced if and only if the sigma protocol verifier is convinced, we get our contradiction.

□

Claim 3.1 (Producing an SFE Encryption of t with dk is Hard)**.** *Let* $\mathsf{P}^* = \{\mathsf{P}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ *be a quantum polynomial-size prover in Protocol 1, sending a deterministic first message* $\mathsf{cmt}_1, \mathsf{cmt}_2$ *where there exists* $\mathsf{dk} \in \mathsf{SFE.Gen}(1^\lambda)$ *s.t.* $\mathsf{cmt}_2 \in \mathsf{Com}(1^\lambda, \mathsf{dk})$*. Then there exists a negligible function* $\mu(\cdot)$ *such that the probability that* $t = \mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{ct}_\mathsf{P})$ *is bounded by* $\mu(\lambda)$*.*

Proof. The proof will be based on the security of the QFHE, and on the security of the CC obfuscation. We start with observing that the security of the QFHE implies that for every efficient quantum adversary $\mathsf{A}^* = \{\mathsf{A}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$, there's a negligible function $\mu(\cdot)$ s.t. the probability that A^* finds t given $\mathsf{pk}, \mathsf{ct} \leftarrow \mathsf{QHE.Enc}_{\mathsf{pk}}(t)$ for a uniformly random $t \leftarrow \{0,1\}^\lambda$, is bounded by $\mu(\lambda)$ - we will assume toward contradiction that our claim is false, that is, we assume that P^* sends ct_P s.t. $t = \mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{ct}_\mathsf{P})$ with noticeable probability (for infinitely many security parameters), and get a contradiction with the last claim about the hardness of finding a random encrypted t.

Using P^* and the fact that $t = \mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{ct}_\mathsf{P})$ with noticeable probability, we now describe a (non-uniform) algorithm A^* that finds t given $\mathsf{pk}, \mathsf{ct} \leftarrow \mathsf{QHE.Enc}_{\mathsf{pk}}(t)$ for $t \leftarrow \{0,1\}_\lambda$ and thus breaks the security of the QFHE. As part of the non-uniform advice of A^*, it will have the secret SFE key dk, which is fixed. Given $\mathsf{pk}, \mathsf{ct} \leftarrow \mathsf{QHE.Enc}_{\mathsf{pk}}(t)$, the algorithm A^* will use the simulator Sim^{CC} (from the simulation property of the CC obfuscation) and send to P^* the following, as the protocol message sent at step 2b,

$$\mathsf{pk},\ \mathsf{ct},\ \mathsf{Sim}^{\mathsf{CC}}(1^{|\mathsf{QHE.Dec}|}, 1^{\ell+|\beta|}, 1^\lambda)\ ,$$

where ℓ is the randomness complexity of the QFHE key generation algorithm $\mathsf{QHE.Keygen}$. P^* will respond with ct_P, and A^* uses dk to output $\mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{ct}_\mathsf{P})$.

We now use the simulation property guarantee of the CC obfuscation: Note that the probability that P^* outputs ct_P s.t. $\mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{ct}_\mathsf{P}) = t$ in the simulated setting, where A^* sends $\mathsf{Sim}^{\mathsf{CC}}(1^{|\mathsf{QHE.Dec}_{0^{|\mathsf{sk}|}}|}, 1^{|\mathsf{sk}|+|\beta|}, 1^\lambda)$ instead of $\widetilde{\mathbf{CC}}$, is negligibly close to the probability that it outputs ct_P s.t. $\mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{ct}_\mathsf{P}) = t$ in the regular setting where it gets $\widetilde{\mathbf{CC}}$ - this is due to the security of the CC obfuscator. Because we know that in the regular interaction, P^* sends ct_P s.t. $t = \mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{ct}_\mathsf{P})$ with a noticeable probability, this implies that so does A^*, in contradiction. □

3.2 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, $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}, \mathsf{V}^*(\rho)\rangle(x)$. Throughout this section, a malicious verifier V^* is modeled as a family of non-uniform quantum circuits with auxiliary quantum input, consistently with the rest of the paper.

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 (prover WI), 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 (prover WI), 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 Proof. 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 pk, $\mathsf{ct}_{\mathsf{V}^*}$, $\widetilde{\mathbf{CC}}$.
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 sigma protocol simulator $\alpha \leftarrow \Sigma.\mathsf{S}(x, 0^{|\beta|})$, 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, $\mathsf{Sim}_{\mathrm{a}}$ outputs the aborting verifier's output. Otherwise, $\mathsf{Sim}_{\mathrm{a}}$ outputs $\mathtt{Fail}$.

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

1. **Simulation of Initial Commitments and Verifier Message:**
   (a) $\mathsf{Sim}_{\mathrm{na}}$ 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 pk, $\mathsf{ct}_{\mathsf{V}^*}$, $\widetilde{\mathbf{CC}}$.
2. **Non-Black-Box Extraction Attempt:**

(a) $\mathsf{Sim}_{\mathrm{na}}$ computes

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

$\mathsf{Sim}_{\mathrm{na}}$ 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) $\mathsf{Sim}_{\mathrm{na}}$ 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) $\mathsf{Sim}_{\mathrm{na}}$ 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 $(r, \beta') = \widetilde{\mathbf{CC}}(\mathsf{ct}_s)$.

(d) $\mathsf{Sim}_{\mathrm{na}}$ checks validity: $(\mathsf{pk}', \mathsf{sk}) = \mathsf{QHE.Keygen}(1^\lambda; r)$, if $\mathsf{pk}' \neq \mathsf{pk}$ then it halts simulation and outputs `Fail`. Otherwise, $\mathsf{Sim}_{\mathrm{na}}$ obtains the inner state of V^* by decryption: $\rho^{(2)} \leftarrow \mathsf{QHE.QDec}_{\mathsf{sk}}(\mathsf{ct}_{\rho^{(2)}})$. Additionally, $\mathsf{Sim}_{\mathrm{na}}$ 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) $\mathsf{Sim}_{\mathrm{na}}$ executes the sigma protocol simulator $(\alpha, \gamma) \leftarrow \Sigma.\mathsf{S}(x, \beta')$ and sends α to V^*.

   (b) V^* returns β.

4. **WI Proof by the Malicious Verifier:** $\mathsf{Sim}_{\mathrm{na}}$ 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 `Fail`.

5. **Simulation of the Prover's WI Proof and Information Reveal:** $\mathsf{Sim}_{\mathrm{na}}$ gives V^* a WI proof using the witness that shows $\mathsf{cmt}_1, \mathsf{cmt}_2$ are both valid commitments (and that cmt_2 is a commitment to the SFE key dk used in step 2c). After the proof, $\mathsf{Sim}_{\mathrm{na}}$ sends γ to V^*.

6. **Simulation Verdict:** If V^* completed interaction without aborting and gave a convincing WI proof, $\mathsf{Sim}_{\mathrm{na}}$ outputs the verifier's output. Otherwise, $\mathsf{Sim}_{\mathrm{na}}$ outputs `Fail`.

$\mathsf{Sim}_{\mathrm{comb}}(x, \mathsf{V}^*, \rho)$: Sample $b \leftarrow \{0,1\}$ and execute $\mathsf{Sim}_b(x, \mathsf{V}^*, \rho)$.

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

1. Generate the circuit $\mathsf{Sim}_{\mathrm{comb},x,\mathsf{V}^*}$, which is the circuit implementation of $\mathsf{Sim}_{\mathrm{comb}}$, with hardwired input x, V^*, that is, the only input to $\mathsf{Sim}_{\mathrm{comb},x,\mathsf{V}^*}$ is the quantum state ρ.

2. Let R be the algorithm from Lemma 2.1. The output of the simulation is $\mathsf{R}(\mathsf{Sim}_{\mathrm{comb},x,\mathsf{V}^*}, \rho, \lambda)$.

Proof of Simulation Validity. We now turn to prove that the simulated output $\mathsf{Sim}(x, \mathsf{V}^*, \rho)$ is computationally indistinguishable from $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}, \mathsf{V}^*(\rho)\rangle(x)$. This is done in several steps:

1. **Simulating aborting interactions:** Let $\mathsf{V}_{\mathrm{a}}^*$ be the augmented verifier that is identical to V^*, with the exception that if V^* does not abort, $\mathsf{V}_{\mathrm{a}}^*$ outputs `Fail`. Then the output of $\mathsf{Sim}_{\mathrm{a}}$ is indistinguishable from the output of $\mathsf{V}_{\mathrm{a}}^*$ 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:

   - $\mathsf{Sim}_{\text{comb}} \neq \texttt{Fail}$ with probability negligibly close to $\frac{1}{2}$, for every verifier and auxiliary input ρ.
   - The output of V^* in a real interaction is indistinguishable from the output of $\mathsf{Sim}_{\text{comb}}$ conditioned on $\mathsf{Sim}_{\text{comb}} \neq \texttt{Fail}$.

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

Proposition 3.2 (Similarity of Aborting Part)**.** *Let* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ *a polynomial-size quantum verifier, and let* $\mathsf{OUT}_{\mathsf{V}^*_{\text{a}}}$ *be the verifier's output at the end of protocol such that if* V^* *does not abort, the output is* `Fail`*. Then,*

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

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

Proof. We prove the claim by a hybrid argument, specifically, we consider hybrid distributions, all of which will be computationally indistinguishable.

- Hyb_0 : The output distribution of Sim_{a}.
- Hyb_1 : This hybrid process is identical to Hyb_0, with the exception that when the simulator gives a WI proof in the simulation, it uses the witness w in the proof, that proves the first statement in the OR statement ($x \in \mathcal{L}$) rather then the second statement.
- Hyb_2 : This hybrid process is identical to Hyb_1, with the exception that cmt_1 is a commitment to w rather than to $0^{|w|}$.
- Hyb_3 : This hybrid process is identical to Hyb_2, with the exception that the message α that the simulator sends to V^* is generated by the actual sigma protocol $(\alpha, \tau) \leftarrow \Sigma.\mathsf{P}_1(x, w)$, and not by the sigma protocol simulator $\Sigma.\mathsf{S}(x, 0^{|\beta|})$. Note that this process is exactly $\mathsf{OUT}_{\mathsf{V}^*_{\text{a}}}\langle\mathsf{P}(w), \mathsf{V}^*(\rho)\rangle(x)$.

It is left to reason about the indistinguishability between each two subsequent hybrids.

- $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$: Follows from the witness-indistinguishability property of the WI proof that the simulator gives (as the prover) in step 5 of the protocol.
- $\mathsf{Hyb}_1 \approx_c \mathsf{Hyb}_2$: Follows from the hiding property of the commitment cmt_1.
- $\mathsf{Hyb}_2 \approx_c \mathsf{Hyb}_3$: Follows from Claim 2.1.

□

Proposition 3.3 (Similarity of Non-Aborting Part)**.** *Let* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ *a polynomial-size quantum verifier, and let* $\mathsf{OUT}_{\mathsf{V}^*_{\text{na}}}$ *be the verifier's output at the end of protocol such that if* V^* *aborts, the output is* `Fail`*. Then,*

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

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

Proof. We prove the claim by a hybrid argument, specifically, we consider hybrid distributions, all of which will be computationally indistinguishable.

- Hyb_0 : The output distribution of $\mathsf{Sim}_{\mathrm{na}}$.

- Hyb_1 : This hybrid process is identical to Hyb_0, with the exception that when the simulator gives a WI proof in the simulation, it uses the witness w in the proof, that proves the first statement in the OR statement ($x \in \mathcal{L}$) rather then the second statement.

- Hyb_2 : This hybrid process is identical to Hyb_1, with the exception that cmt_1 is a commitment to the witness w rather than to $0^{|w|}$, and cmt_2 is a commitment to $0^{|\mathsf{dk}|}$ rather than to the generated SFE key $\mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda)$.

- Hyb_3 : This hybrid process is identical to Hyb_2, with the exception that if the verifier's message β from part 3b of the simulation does not match the extracted β' from part 2c of the simulation, the process halts on the spot and outputs `Fail`.

- Hyb_4 : This hybrid process is identical to Hyb_3, with the exception that in parts 3a, 5 where the simulator sends α and γ, instead of computing α, γ using $\Sigma.\mathsf{S}$, it computes $(\alpha, \tau) \leftarrow \Sigma.\mathsf{P}_1(x, w)$ and $\gamma \leftarrow \Sigma.\mathsf{P}_3(\beta, \tau)$.

- Hyb_5 : This hybrid process is identical to Hyb_4, with the exception that it does not perform the check described in Hyb_3, that is, even if the extracted challenge and sent challenge are distinct $\beta' \neq \beta$, the process carries on regularly.

- Hyb_6 : At this point in our series of hybrid distributions, we do not use the extracted challenge β', and we would like to move to a process that does not perform extraction. The current hybrid will still perform extraction, but will not use the extracted information. This hybrid process is identical to Hyb_5, with the changes described next. If the first verifier message *is not* explainable then the process chooses to fail and outputs `Fail`. If the first verifier message *is* explainable, note that it fixes a public and secret key pair $(\mathsf{pk}, \mathsf{sk}) = \mathsf{QHE.Keygen}(1^\lambda; r)$, and a string $s \in \{0,1\}^\lambda$ hidden inside the CC program $\widetilde{\mathbf{CC}}$. In that case, the process acts like Hyb_5, except that at the end of step 2b of the simulation, the process inefficiently obtains sk and uses it to decrypt $\left(\mathsf{ct}_s^{\mathrm{SFE}}, \mathsf{ct}_{\rho^{(2)}}\right)$, instead of using the program $\widetilde{\mathbf{CC}}$ to get sk. The process also inefficiently obtains s and performs a check: if $s \neq \mathsf{SFE.Dec}_{\mathsf{dk}}(\mathsf{QHE.Dec}_{\mathsf{sk}}(\mathsf{ct}_s^{\mathrm{SFE}}))$ then the process fails and outputs `Fail`, and otherwise continues the simulation regularly as in the rest of Hyb_5.

- Hyb_7 : This process will get rid of extraction altogether and will not perform the homomorphic evaluation of the verifier's response. This distributions is the output of a process that acts like Hyb_6, with the exception that if the first verifier message is explainable (in particular, ct_{V} is a QFHE encryption of some $t \in \{0,1\}^\lambda$), then as the prover message from step 2c of the protocol, the process sends $\mathsf{SFE.Enc}_{\mathsf{dk}}(t)$. If at step 2d the verifier responds with $\hat{\mathsf{ct}}$ s.t. $s = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$ then the simulation continues regularly as in Hyb_6, and otherwise the process fails and outputs `Fail`.

- Hyb_8 : Like the previous two processes, this process is also inefficient. This hybrid process is identical to Hyb_7, with the exception that it does not perform the check on the verifier's response $\hat{\mathsf{ct}}$, and continues regularly either way, even when $s \neq \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$.

- Hyb_9 : We now go back to an efficient hybrid process. This hybrid process is identical to Hyb_8, with the exception that instead of performing the inefficient check on the verifier's first message from step 1b (and then either halting and outputting `Fail`, or sending $\mathsf{SFE.Enc}_{\mathsf{dk}}(t)$ to V^*), the process always sends $\mathsf{SFE.Enc}_{\mathsf{dk}}(0^\lambda)$ to V^*, and continues simuation regularly. Observe that this process is exactly $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{na}}}\langle \mathsf{P}(w), \mathsf{V}^*(\rho)\rangle(x)$.

We now prove that each pair of consecutive distributions are computationally indistinguishable, and our proof is finished.

- $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$: This indistinguishability follows from the witness-indistinguishability property of the WI proof that the simulator gives in step 5 of the simulation.
- $\mathsf{Hyb}_1 \approx_c \mathsf{Hyb}_2$: This indistinguishability follows from the hiding of the commitments $\mathsf{cmt}_1, \mathsf{cmt}_2$ that the simulator gives in step 1a of the simulation.
- $\mathsf{Hyb}_2 \approx_s \mathsf{Hyb}_3$: The indistinguishability follows from the perfect correctness of both the CC obfuscation and the SFE schemes, along with the soundness of the WI proof. Assume toward contradiction that the two distributions are distinguishable and fix, by an averaging argument, the partial transcript T' that is generated at the end of step 3b of the simulation, which maximizes distinguishability. We consider two cases for T':
  - T' is explainable. In that case it follows from the perfect correctness of the CC obfuscation and the perfect correctness of the SFE evaluation, that the extracted β' and the sent β are necessarily equal, and the processes are identical (and have statistical distance of 0), in contradiction.
  - T' is not explainable. In that case, recall that cmt_1 is a commitment to a witness and thus the statement in the verifier's WI proof is necessarily false. By the soundness of the WI proof, V^* will fail in proving the statement with overwhelming probability, which implies that with the same probability the output in the process Hyb_2 is $\mathtt{Fail}$. Because with *at least* the same probability, the output in Hyb_3 is also $\mathtt{Fail}$, the contradiction follows.
- $\mathsf{Hyb}_3 \approx_c \mathsf{Hyb}_4$: This indistinguishability follows from the special zero-knowledge property of the sigma protocol.
- $\mathsf{Hyb}_4 \approx_s \mathsf{Hyb}_5$: The statistical indistinguishability follows from the exact same reasoning that explains why distributions $\mathsf{Hyb}_2 \approx_s \mathsf{Hyb}_3$.
- $\mathsf{Hyb}_5 \approx_s \mathsf{Hyb}_6$: This indistinguishability will follow from the perfect correctness of the CC obfuscation, the statistical correctness of the QFHE and from the soundness of the WI proof that V^* gives. Formally, assume toward contradiction that the two distributions are distinguishable and fix, by an averaging argument, the partial transcript T' and inner quantum state $\rho^{(1)}$ of V^* generated at the end of step 1b of the simulation, that maximize the distinguishability. Denote by $\widetilde{\mathsf{Hyb}}_5$, $\widetilde{\mathsf{Hyb}}_6$ the distributions that carry on from the point that T', $\rho^{(1)}$ are fixed, according to Hyb_5, Hyb_6, respectively. Consider two cases for T'.
  - T' is not explainable. In that case, $\widetilde{\mathsf{Hyb}}_6$ outputs $\mathtt{Fail}$ with probability 1, and by the soundness of the WI proof of the verifier, the proof is going to fail with overwhelming probability (in the process $\widetilde{\mathsf{Hyb}}_5$) and with at least the same probability the output is going to be $\mathtt{Fail}$, and the two distributions will have at most negligible statistical distance, in contradiction.
  - T' is explainable, which means that the verifier's first message fixes t, s, sk, and also r_t the QFHE encryption randomness s.t. $\mathsf{ct}_{\mathsf{V}^*} = \mathsf{QHE.Enc}_{\mathsf{pk}}(t; r_t)$. Consider the quantum circuit C that for input $(t, \rho^{(1)})$, encrypts $\mathsf{ct}_t \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(t)$, executes $(\hat{\mathsf{ct}}, \rho^{(2)}) \leftarrow \mathsf{V}^*(\mathsf{ct}_t, \rho^{(1)})$, decrypts $s' = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$ and outputs $s', \hat{\mathsf{ct}}, \rho^{(2)}$. Now, observe the following about the distributions $\widetilde{\mathsf{Hyb}}_5$, $\widetilde{\mathsf{Hyb}}_6$.

  - $*$ $\widetilde{\mathsf{Hyb}}_5$ can be described by the following process: Encrypt $\mathsf{ct}_{\mathsf{V}^*} = \mathsf{QHE.Enc}_{\mathsf{pk}}(t; r_t)$, $\mathsf{ct}_{\rho^{(1)}} \leftarrow\leftarrow \mathsf{QHE.QEnc}_{\mathsf{pk}}(\rho^{(1)})$, perform homomorphic evaluation of the circuit C, and then decrypt with sk to get $\left(s', \hat{\mathsf{ct}}, \rho^{(2)}\right)$.
    If $s' \neq s$ then output `Fail`, otherwise carry on the simulation as in Hyb_5. The fact that $\widetilde{\mathsf{Hyb}}_5$ can be described by this process follows from the fact that by the perfect correctness of the CC obfuscation, if the first verifier message is explainable then $\widetilde{\mathbf{CC}}$ indeed executes the decryption circuit $\mathsf{QHE.Dec}_{\mathsf{sk}}(\cdot)$ s.t. if the result was s, it outputs the QFHE key-generation randomness r (which in turn yields sk), and if the result wasn't s, $\widetilde{\mathbf{CC}}$ necessarily yields $\perp$.
  - $*$ $\widetilde{\mathsf{Hyb}}_6$ can be described by the following process: the exact same homomorphic evaluation process as described above, except that after getting the output $\left(s', \hat{\mathsf{ct}}, \rho^{(2)}\right)$, the check is that $s = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$, and if the check fails the output is `Fail`, and if the check succeeds then the process continues simulation regularly in the exact same way as in $\widetilde{\mathsf{Hyb}}_5$.

  The above descriptions of $\widetilde{\mathsf{Hyb}}_5$, $\widetilde{\mathsf{Hyb}}_6$ imply that the statistical distance between them is bounded by the probability that the check in one process fails and in the other it succeeds, which in turn bounded by the probability that $s' \neq \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$. The point is, due to the fact that the SFE decryption algorithm is deterministic, it is always the case when evaluating the circuit C (out in the open, not under homomorphic evaluation) we have $s' = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$. It follows by the statistical correctness of the QFHE that the probability that the evaluated $s', \hat{\mathsf{ct}}$ are s.t. $s' \neq \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$, and thus the bound on the statistical distance between $\widetilde{\mathsf{Hyb}}_5$, $\widetilde{\mathsf{Hyb}}_6$, in contradiction.

- $\mathsf{Hyb}_6 \approx_s \mathsf{Hyb}_7$: This indistinguishability follows directly from the statistical correctness of the QFHE. Assume toward contradiction that the two distributions are distinguishable and fix, by an averaging argument, the partial transcript T' and inner quantum state $\rho^{(1)}$ of V^* generated at the end of step 1b of the simulation, that maximize the distinguishability. Denote by $\widetilde{\mathsf{Hyb}}_6$, $\widetilde{\mathsf{Hyb}}_7$ the distributions that carry on from the point that $T', \rho^{(1)}$ are fixed, according to Hyb_6, Hyb_7, respectively. Consider two cases for T'.

  - T' is not explainable. In that case both processes act the same and output `Fail`, and are indistinguishable.
  - T' is explainable, which means that the verifier's first message fixes t, s, sk, and also r_t the QFHE encryption randomness s.t. $\mathsf{ct}_{\mathsf{V}^*} = \mathsf{QHE.Enc}_{\mathsf{pk}}(t; r_t)$. In that case, recall the circuit C from the above proof of the indistinguishability $\mathsf{Hyb}_5 \approx_s \mathsf{Hyb}_6$, and observe the following about the distributions $\widetilde{\mathsf{Hyb}}_6$, $\widetilde{\mathsf{Hyb}}_7$.
    - $*$ The distribution $\widetilde{\mathsf{Hyb}}_6$ can be described by the following process: Encrypt $\mathsf{ct}_{\mathsf{V}^*} = \mathsf{QHE.Enc}_{\mathsf{pk}}(t; r_t)$, $\mathsf{ct}_{\rho^{(1)}} \leftarrow\leftarrow \mathsf{QHE.QEnc}_{\mathsf{pk}}(\rho^{(1)})$, perform homomorphic evaluation of the circuit C, and then decrypt with sk to get $\left(s', \hat{\mathsf{ct}}, \rho^{(2)}\right)$. If $s = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$ then process continues simulation regularly as in Hyb_6, and otherwise fails and outputs `Fail`.
    - $*$ The distribution $\widetilde{\mathsf{Hyb}}_7$ can be described by the following process: Instead of encrypting $t, \rho^{(1)}$ and computing C under homomorphic evaluation (and then decrypting), we simply execute $\left(\hat{\mathsf{ct}}, \rho^{(2)}\right) \leftarrow C(t, \rho^{(1)})$ in the clear. The process continues in the exact same way as described after the homomorphic evaluation in $\widetilde{\mathsf{Hyb}}_6$; If $s = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$ then process continues simulation regularly, and otherwise fails and outputs `Fail`.

  The above implies that the only difference between the two processes is the fact that in $\widetilde{\mathsf{Hyb}}_6$ we execute C under homomorphic evaluation, and in $\widetilde{\mathsf{Hyb}}_7$ we execute C in the clear. By the statistical correctness of the QFHE, it follows that the two processes are statistically indistinguishable, in contradiction.

- $\mathsf{Hyb}_7 \approx_s \mathsf{Hyb}_8$: This indistinguishability follows from the perfect correctness of the SFE encryption and the soundness of the WI proof that V^* gives. Assume toward contradiction that the distributions are distinguishable and fix, by an averaging argument, the partial transcript T' (and inner verifier state $\rho^{(2)}$) generated after the verifier's second message $\hat{\mathsf{ct}}$. If the first verifier message was explainable and also $s = \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$ then then processes are identical, as they carry on simulation in the exact same way. If the first verifier message was not explainable then again, both processes fail and output `Fail` and are identical, and if the first verifier message is explainable but $s \neq \mathsf{SFE.Dec}_{\mathsf{dk}}(\hat{\mathsf{ct}})$, it follows Hyb_7 outputs `Fail`, and in Hyb_8, by the perfect correctness of the SFE evaluation, the transcript cannot be explainable, and thus the WI proof by the verifier fails with overwhelming probability, and with the same probability the output of Hyb_8 is `Fail`, and the processes are indistinguishable.

- $\mathsf{Hyb}_8 \approx_c \mathsf{Hyb}_9$: This indistinguishability follows from the input privacy property of the SFE encryption. More precisely, as usual, we assume toward contradiction that the distributions are distinguishable and we fix the transcript until the end of step 1b of the simulation. If the transcript is not explainable then Hyb_8 outputs `Fail`, and Hyb_9 outputs `Fail` with overwhelming probability, because the WI proof of the verifier will fail with overwhelming probability. If the transcript is explainable, we can get either an SFE encryption of t or of 0, as t is fixed by the averaging argument. By continuing the simulation regularly, as identically performed in both processes, we get the reduction from breaking the security of the SFE encryption to distinguishing between Hyb_8 and Hyb_9.

□

Corollary 3.1 (Probabilities to Abort are Negligibly Close Over Different Cases)**.** *For a quantum auxiliary input ρ, instance in the language $x \in \{0,1\}^\lambda \cap \mathcal{L}$ and witness $w \in \mathcal{R}_{\mathcal{L}}(x)$, define the following probabilities.*

- $a(x, \rho)$: *The probability that in the simulation $\mathsf{Sim}_{\mathrm{a}}(x, \mathsf{V}^*, \rho)$, the verifier V^* aborted before the end of step 5 where the simulator simulates the Prover's WI proof, or failed to prove its WI statement in step 4 (i.e. the simulation of $\mathsf{Sim}_{\mathrm{a}}(x, \mathsf{V}^*, \rho)$ was aboting).*

- $b(x, \rho)$: *The probability that in the simulation $\mathsf{Sim}_{\mathrm{na}}(x, \mathsf{V}^*, \rho)$, the verifier V^* aborted before the end of step 5 where the simulator simulates the Prover's WI proof, or failed to prove its WI statement in step 4 (i.e. the simulation of $\mathsf{Sim}_{\mathrm{na}}(x, \mathsf{V}^*, \rho)$ was aboting).*

- $c(x, \rho, w)$: *The probability that the interaction $\langle \mathsf{P}(w), \mathsf{V}^*(\rho)\rangle(x)$ was aborting.*

There exists a negligible function $\mathrm{negl}(\cdot)$ *s.t. for every sequences $\rho = \{\rho_\lambda\}_{\lambda\in\mathbb{N}}$, $x = \{x_\lambda\}_{\lambda\in\mathbb{N}}$, $w = \{w_\lambda\}_{\lambda\in\mathbb{N}}$ where,*

- $\forall \lambda \in \mathbb{N} : \rho_\lambda$ *is a λ^c-size quantum state (for some constant $c \in \mathbb{N}$),*
- $\forall \lambda \in \mathbb{N} : x_\lambda \in \{0,1\}^\lambda \cap \mathcal{L}$,
- $\forall \lambda \in \mathbb{N} : w_\lambda \in \mathcal{R}_{\mathcal{L}}(x_\lambda)$,

we have

$$\forall \lambda \in \mathbb{N} : |a(x_\lambda, \rho_\lambda) - b(x_\lambda, \rho_\lambda)|\,, |b(x_\lambda, \rho_\lambda) - c(x_\lambda, \rho_\lambda, w_\lambda)|\,, |c(x_\lambda, \rho_\lambda, w_\lambda) - a(x_\lambda, \rho_\lambda)| \leq \mathrm{negl}(\lambda)\ .$$

Proof. It immediately follows from Proposition 3.2 that the distance between $a(x,\rho)$ and $c(x,\rho,w)$ is negligible. By the same reasoning it follows from Proposition 3.3 that $b(x,\rho)$ and $c(x,\rho,w)$ are negligibly close. By triangle inequality it follows that also $a(x,\rho)$ and $b(x,\rho)$ are negligibly close. □

From the above it follows that the success probability of $\mathsf{Sim}_{\mathrm{comb}}(x,\mathsf{V}^*,\rho)$ is negligibly close to $\frac{1}{2}$, regardless of the quantum state ρ.

Corollary 3.2 (Success Probability of $\mathsf{Sim}_{\mathrm{comb}}$ is Input-Oblivious)**.** *For every quantum verifier* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda\}_{\lambda\in\mathbb{N}}$ *there exists a negligible function* $\mathrm{negl}(\cdot)$ *s.t. for every instance in the language* $x = \{x_\lambda\}_{\lambda\in\mathbb{N}}$ *and quantum auxiliary input* $\rho = \{\rho_\lambda\}_{\lambda\in\mathbb{N}}$ *for the verifier, we have*

$$\forall\lambda\in\mathbb{N}: \left|\Pr\left[\textit{The simulation } \mathsf{Sim}_{\mathrm{comb}}(x_\lambda,\mathsf{V}^*_\lambda,\rho_\lambda) \textit{ succeeds}\right] - \frac{1}{2}\right| \le \mathrm{negl}(\lambda) \enspace .$$

Proof.

$$\begin{aligned}
\forall\lambda\in\mathbb{N}: &\left|\Pr\left[\text{The simulation } \mathsf{Sim}_{\mathrm{comb}}(x_\lambda,\mathsf{V}^*_\lambda,\rho_\lambda) \text{ succeeds}\right] - \frac{1}{2}\right| \\
&= \left|\frac{1}{2}\cdot\Pr\left[\text{The simulation } \mathsf{Sim}_{\mathrm{a}}(x_\lambda,\mathsf{V}^*_\lambda,\rho_\lambda) \text{ succeeds}\right]\right. \\
&\qquad \left. + \frac{1}{2}\cdot\Pr\left[\text{The simulation } \mathsf{Sim}_{\mathrm{na}}(x_\lambda,\mathsf{V}^*_\lambda,\rho_\lambda) \text{ succeeds}\right] - \frac{1}{2}\right| \\
&= \left|\frac{1}{2}\cdot a(x_\lambda,\rho_\lambda) + \frac{1}{2}\cdot\big(1 - b(x_\lambda,\rho_\lambda)\big) - \frac{1}{2}\right| \\
&= \frac{1}{2}\cdot\left|a(x_\lambda,\rho_\lambda) - b(x_\lambda,\rho_\lambda)\right| \le \mathrm{negl}(\lambda) \enspace ,
\end{aligned}$$

where the last inequality is due to Corollary 3.1. □

We next prove that conditioned on succeeding, the output distribution of the simulator $\mathsf{Sim}_{\mathrm{comb}}$ is indistinguishable from the real interaction.

Proposition 3.4 (The Output of a Successful $\mathsf{Sim}_{\mathrm{comb}}$ is Indistinguishable from Real Interaction)**.** *Let* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda,\rho_\lambda\}_{\lambda\in\mathbb{N}}$ *be a polynomial-size quantum verifier. For* $x\in\mathcal{L}$*, let* $\widetilde{\mathsf{Sim}}_{\mathrm{comb}}(x,\mathsf{V}^*,\rho)$ *denote the conditional distribution of* $\mathsf{Sim}_{\mathrm{comb}}(x,\mathsf{V}^*,\rho)$*, conditioned on the simulation being successful. Then,*

$$\{\mathsf{OUT}_{\mathsf{V}^*}\langle\mathsf{P}(w),\mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x)\}_{\substack{\lambda\in\mathbb{N},\\ x\in\mathcal{L}\cap\{0,1\}^\lambda,\\ w\in\mathcal{R}_{\mathcal{L}}(x)}} \approx_c \{\widetilde{\mathsf{Sim}}_{\mathrm{comb}}(x,\mathsf{V}^*_\lambda,\rho_\lambda)\}_{\substack{\lambda\in\mathbb{N},\\ x\in\mathcal{L}\cap\{0,1\}^\lambda,\\ w\in\mathcal{R}_{\mathcal{L}}(x)}} \enspace .$$

Proof. Denote the following conditional distributions.

- $A_{\mathsf{Sim}} = \{A_{\mathsf{Sim},\lambda}\}_{\lambda\in\mathbb{N}}$: A conditional distribution of $\mathsf{Sim}_{\mathrm{a}}(x,\mathsf{V}^*,\rho)$, conditioned on that the output is not `Fail` (might be an empty distribution, if $a(x,\rho)=0$).
- $S_{\mathsf{Sim}} = \{S_{\mathsf{Sim},\lambda}\}_{\lambda\in\mathbb{N}}$: A conditional distribution of $\mathsf{Sim}_{\mathrm{na}}(x,\mathsf{V}^*,\rho)$, conditioned on that the output is not `Fail` (might be an empty distribution, if $b(x,\rho)=1$).
- $A_{\langle\mathsf{P},\mathsf{V}^*\rangle} = \{A_{\langle\mathsf{P},\mathsf{V}^*\rangle,\lambda}\}_{\lambda\in\mathbb{N}}$: A conditional distribution of $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{a}}}\langle\mathsf{P},\mathsf{V}^*\rangle$ (from 3.2), conditioned on that the output is not `Fail` (might be an empty distribution, if $c(x,\rho,w)=0$).
- $S_{\langle\mathsf{P},\mathsf{V}^*\rangle} = \{S_{\langle\mathsf{P},\mathsf{V}^*\rangle,\lambda}\}_{\lambda\in\mathbb{N}}$: A conditional distribution of $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{nm}}}\langle\mathsf{P},\mathsf{V}^*\rangle$ (from 3.3), conditioned on that the output is not `Fail` (might be an empty distribution, if $c(x,\rho,w)=1$).

Observe that the distribution $\widetilde{\mathsf{Sim}}_{\mathrm{comb}}(x, \mathsf{V}^*, \rho)$ is the distribution generated by outputting a sample from A_{Sim} with probability $\frac{a(x,\rho)}{1+a(x,\rho)-b(x,\rho)}$, and a sample from S_{Sim} with probability $\frac{1-b(x,\rho)}{1+a(x,\rho)-b(x,\rho)}$. Additionally, observe that the distribution $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}(w), \mathsf{V}^*(\rho)\rangle(x)$ is the distribution generated by outputting a sample from $A_{\langle \mathsf{P},\mathsf{V}^*\rangle}$ with probability $c(x, \rho, w)$ and from $S_{\langle \mathsf{P},\mathsf{V}^*\rangle}$ with probability $1 - c(x, \rho, w)$. We will show that the two distributions are computationally indistinguishable by a hybrid argument. Consider the following distributions.

- Hyb_0 : The distribution $\widetilde{\mathsf{Sim}}_{\mathrm{comb}}(x, \mathsf{V}^*, \rho)$.
- Hyb_1 : Same as in Hyb_0, with the exception that instead of sampling from A_{Sim} with probability $\frac{a(x,\rho)}{1+a(x,\rho)-b(x,\rho)}$ (and from S_{Sim} with probability $\frac{1-b(x,\rho)}{1+a(x,\rho)-b(x,\rho)}$), it samples from A_{Sim} with probability $a(x, \rho)$ (and from S_{Sim} with probability $1 - a(x, \rho)$).
- Hyb_2 : Same as in Hyb_1, but the probability $a(x, \rho)$ is changed to $c(x, \rho, w)$.
- Hyb_3 : Same as in Hyb_2, with the exception that with probability $c(x, \rho, w)$, the process outputs a sample from $A_{\langle \mathsf{P},\mathsf{V}^*\rangle}$ rather than from A_{Sim}.
- Hyb_4 : Same as in Hyb_3, with the exception that with probability $1-c(x, \rho, w)$, the process outputs a sample from $S_{\langle \mathsf{P},\mathsf{V}^*\rangle}$ rather than from S_{Sim}. This process is exactly $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}(w), \mathsf{V}^*(\rho)\rangle(x)$.

It is left to explain why each consecutive pair of distributions are computationally indistinguishable, and our proof is finished. For the following, define $a'(\lambda) := a(x_\lambda, \rho_\lambda)$, $b'(\lambda) := b(x_\lambda, \rho_\lambda)$, $c'(\lambda) := c(x_\lambda, \rho_\lambda, w_\lambda)$.

- $\mathsf{Hyb}_0 \approx_s \mathsf{Hyb}_1$: Due to the fact that $a'(\lambda)$ and $b'(\lambda)$ are negligibly close (Corollary 3.1), it follows that $a'(\lambda)$ and $\frac{a(x,\rho)}{1+a(x,\rho)-b(x,\rho)}$ are also negligibly close, and thus follows the statistical indistinguishability.
- $\mathsf{Hyb}_1 \approx_s \mathsf{Hyb}_2$: The probabilities $a'(\lambda)$ and $c'(\lambda)$ are negligibly close due to Corollary 3.1, and the statistical indistinguishability follows.
- $\mathsf{Hyb}_2 \approx_c \mathsf{Hyb}_3$: Assume toward contradiction that the indistinguishbility does not hold, this means there is a distinguisher D^*, an infinite subset $Q \subseteq \mathbb{N}$ and a polynomial $p : \mathbb{N} \to \mathbb{N}$, s.t. for all $\lambda \in Q$, D^* distinguishes with advantage at least $1/p(\lambda)$ between $\mathsf{Hyb}_{2,\lambda}$ and $\mathsf{Hyb}_{3,\lambda}$. We consider two cases for the function c', and show that in both of them the contradiction follows from 3.2.
  - **Case 1:** for every polynomial function $q : \mathbb{N} \to \mathbb{N}$, there are only finitely-many $\lambda \in Q$ s.t. $1 - c'(\lambda) > 1/q(\lambda)$. This means that there is a negligible function μ s.t. $\forall \lambda \in \mathbb{N}$: $1 - c'(\lambda) \leq \mu(\lambda)$. In that case, the contradiction follows directly from Proposition 3.2, because for indices $\lambda \in Q$, a sample from $\mathsf{Sim}_{\mathsf{a}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ is statistically indistinguishable from a sample from $\mathsf{Hyb}_{2,\lambda}$, and a sample from $\mathsf{OUT}_{\mathsf{V}^*_{\mathsf{a}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$ is statistically indistinguishable from a sample from $\mathsf{Hyb}_{3,\lambda}$.
  - **Case 2:** there is a polynomial function $q' : \mathbb{N} \to \mathbb{N}$, s.t. there are infinitely-many $\lambda \in Q$ s.t. $1 - c'(\lambda) > 1/q'(\lambda)$, we denote this infinite set of indices by Q'. For these indices we can violate the indistinguishbility from 3.2. More specifically, for $\lambda \in Q'$ we can sample in polynomial time (say, $q'(\lambda)^2$) and using polynomial-size quantum advice, from a distribution that is statistically indistinguishable from $S_{\mathsf{Sim},\lambda}$, and reduce distinguishing between $\mathsf{Sim}_{\mathsf{a}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ and $\mathsf{OUT}_{\mathsf{V}^*_{\mathsf{a}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$, to distinguishing between $\mathsf{Hyb}_{2,\lambda}$ and $\mathsf{Hyb}_{3,\lambda}$ in the following way.

    When getting a sample from either $\mathsf{Sim}_{\mathsf{a}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ or $\mathsf{OUT}_{\mathsf{V}^*_{\mathsf{a}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$, if the sample's value was `Fail`, approximately sample (as mentioned above, in time $q(\lambda)^2$) from

$S_{\mathsf{Sim},\lambda}$. This can be done, for example, by using a polynomial amount of copies (i.e. $q'(\lambda)^2$) of the quantum advice ρ of the verifier. This output of the reduction (whether it was `Fail` that was swapped to a sample that is close to $S_{\mathsf{Sim},\lambda}$, or whether it was a non-`Fail` and was not swapped) is sent to the distinguisher D^*. Due to the fact that for the cases we got `Fail`, the generated sample is statistically indistinguishable from $S_{\mathsf{Sim},\lambda}$, it follows that when we get a sample from $\mathsf{Sim}_{\mathrm{a}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ then the output sample of our reduction is statistically close to $\mathsf{Hyb}_{2,\lambda}$, and when we get a sample from $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{a}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$ then the output sample of our reduction is statistically close to $\mathsf{Hyb}_{3,\lambda}$, and we get a contradiction.

- $\mathsf{Hyb}_3 \approx_c \mathsf{Hyb}_4$: Assume toward contradiction that the indistinguishbility does not hold, this means there is a distinguisher D^*, an infinite subset $Q \subseteq \mathbb{N}$ and a polynomial $p : \mathbb{N} \to \mathbb{N}$, s.t. for all $\lambda \in Q$, D^* distinguishes with advantage at least $1/p(\lambda)$ between $\mathsf{Hyb}_{3,\lambda}$ and $\mathsf{Hyb}_{4,\lambda}$. We consider two cases for the function c', and show that in both of them the contradiction follows from 3.3.
  - **Case 1:** for every polynomial function $q : \mathbb{N} \to \mathbb{N}$, there are only finitely-many $\lambda \in Q$ s.t. $c'(\lambda) > 1/q(\lambda)$. This means that there is a negligible function μ s.t. $\forall \lambda \in \mathbb{N} : c'(\lambda) \le \mu(\lambda)$. In that case, the contradiction follows directly from Proposition 3.3, because for indices $\lambda \in Q$, a sample from $\mathsf{Sim}_{\mathrm{na}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ is statistically indistinguishable from a sample from $\mathsf{Hyb}_{3,\lambda}$, and a sample from $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{na}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$ is statistically indistinguishable from a sample from $\mathsf{Hyb}_{4,\lambda}$.
  - **Case 2:** there is a polynomial function $q' : \mathbb{N} \to \mathbb{N}$, s.t. there are infinitely-many $\lambda \in Q$ s.t. $c'(\lambda) > 1/q'(\lambda)$, we denote this infinite set of indices by Q'. For these indices we can violate the indistinguishbility from 3.3. More specifically, for $\lambda \in Q'$ we can sample, in polynomial time (say, $q'(\lambda)^2$) and using polynomial-size quantum advice, from a distribution that is statistically indistinguishable from $A_{\mathsf{Sim},\lambda}$, and reduce distinguishing between $\mathsf{Sim}_{\mathrm{na}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ and $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{na}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$, to distinguishing between $\mathsf{Hyb}_{3,\lambda}$ and $\mathsf{Hyb}_{4,\lambda}$ in the following way.

    When getting a sample from either $\mathsf{Sim}_{\mathrm{na}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ or $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{na}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$, if the sample's value was `Fail`, approximately sample (as mentioned above, in time $q(\lambda)^2$) from $A_{\mathsf{Sim},\lambda}$. This can be done, for example, by using a polynomial amount of copies (i.e. $q'(\lambda)^2$) of the quantum advice ρ of the verifier. This output of the reduction (whether it was `Fail` that was swapped to a sample that is close to $A_{\mathsf{Sim},\lambda}$, or whether it was a non-`Fail` and was not swapped) is sent to the distinguisher D^*. Due to the fact that for the cases we got `Fail`, the generated sample is statistically indistinguishable from $A_{\mathsf{Sim},\lambda}$, it follows that when we get a sample from $\mathsf{Sim}_{\mathrm{na}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda)$ then the output sample of our reduction is statistically close to $\mathsf{Hyb}_{3,\lambda}$, and when we get a sample from $\mathsf{OUT}_{\mathsf{V}^*_{\mathrm{na}}}\langle \mathsf{P}(w_\lambda), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x_\lambda)$ then the output sample of our reduction is statistically close to $\mathsf{Hyb}_{4,\lambda}$, and we get a contradiction.

□

We conclude with proving that the output of the simulation $\mathsf{Sim}(x, \mathsf{V}^*, \rho)$ is indeed computationally indistinguishable from the output of the real interaction $\mathsf{OUT}_{\mathsf{V}}\langle \mathsf{P}, \mathsf{V}^*(\rho)\rangle(x)$.

Proposition 3.5 (Simulation Output is Indistinguishable from Interaction)**.** *For any quantum polynomial-size verifier* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$,

$$\{\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}(w), \mathsf{V}^*_\lambda(\rho_\lambda)\rangle(x)\}_{\substack{\lambda\in\mathbb{N},\\ x\in\mathcal{L}\cap\{0,1\}^\lambda,\\ w\in\mathcal{R}_{\mathcal{L}}(x)}} \approx_c \{\mathsf{Sim}(x, \mathsf{V}^*_\lambda, \rho_\lambda)\}_{\substack{\lambda\in\mathbb{N},\\ x\in\mathcal{L}\cap\{0,1\}^\lambda,\\ w\in\mathcal{R}_{\mathcal{L}}(x)}} .$$

Proof. Let $\mathsf{V}^* = \{\mathsf{V}^*_\lambda\}_{\lambda\in\mathbb{N}}$ be a quantum polynomial-size verifier. According to Corollary 3.2, there is a negligible function $\mathrm{negl}(\cdot)$ s.t. for every instance in the language $x = \{x_\lambda\}_{\lambda\in\mathbb{N}}$ and auxiliary input quantum state $\rho = \{\rho_\lambda\}_{\lambda \in \mathbb{N}}$ for the verifier, we have

$$\forall \lambda \in \mathbb{N} : \left| \Pr\left[\text{The simulation } \mathsf{Sim}_{\text{comb}}(x_\lambda, \mathsf{V}^*_\lambda, \rho_\lambda) \text{ succeeds}\right] - \frac{1}{2} \right| \leq \text{negl}(\lambda) \enspace .$$

Consider the quantum circuit $\mathsf{Sim}_{\text{comb},x,\mathsf{V}^*}$, which is the circuit implementation of $\mathsf{Sim}_{\text{comb}}$ with hard-wired inputs x and V^*, that gets as input only the quantum state ρ. As mentioned above, the success probability of $\mathsf{Sim}_{\text{comb},x,\mathsf{V}^*}$ is negligibly close $\frac{1}{2}$, for *any* quantum state ρ. If we denote the success probability for input ρ by $p(\rho)$ and denote $\varepsilon := \text{negl}(\lambda) + 2^{-\lambda \cdot \frac{3}{4}}$, $p_0 := \frac{1}{4}$ and $q := \frac{1}{2}$, we can see that the 4 conditions for the Quantum Rewinding Lemma 2.1 are satisfied:

- $\lambda \geq \frac{\log(1/\varepsilon)}{4 \cdot p_0(1-p_0)}$.
- For every state ρ, $p_0 \leq p(\rho)$.
- For every state ρ, $|p(\rho) - q| < \varepsilon$.
- $p_0(1-p_0) \leq q(1-q)$.

This implies that $\mathsf{R}(\mathsf{Sim}_{\text{comb},x,\mathsf{V}^*}, \rho, \lambda)$ has trace distance bounded by $4\sqrt{\varepsilon}\frac{\log(1/\varepsilon)}{p_0(1-p_0)}$ from the success-conditioned output distribution of $\mathsf{Sim}_{\text{comb}}(x, \mathsf{V}^*, \rho)$. Since ε is a negligible function of λ, so is $4\sqrt{\varepsilon}\frac{\log(1/\varepsilon)}{p_0(1-p_0)}$.

Finally, recall that $\mathsf{Sim}(x, \mathsf{V}^*, \rho) = \mathsf{R}(\mathsf{Sim}_{\text{comb},x,\mathsf{V}^*}, \rho, \lambda)$, and that proposition 3.4 says that the success-conditioned distribution of $\mathsf{Sim}_{\text{comb}}(x, \mathsf{V}^*, \rho)$ is computationally indistinguishable from $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}(w), \mathsf{V}^*(\rho) \rangle(x)$, and our proof is concluded. □

Remark 3.1 (Classical Universal Simulator for Classical Verifiers). As a side note, we observe that the protocol preserves the trait of classical ZK, that is, classical verifiers learn nothing from the protocol (formally, for classical verifiers there is a classical simulator). A classical simulator showing this will simply execute $\mathsf{Sim}_{\text{comb}}(x, \mathsf{V}^*)$ repeatedly some polynomial number of times (either until it succeeds in one of the tries, or fails in all and then the output is `Fail`), specifically, λ tries will do. Since the probability for $\mathsf{Sim}_{\text{comb}}$ to succeed is $\approx \frac{1}{2}$, the probability to successfully sample from the success-conditioned distribution is overwhelming, and thus the output of the simulator is indistinguishable from the output of the verifier in the real interaction.

4 Quantumly-Extractable Classical Commitments

In this section we show how to use any constant-round post-quantum zero-knowledge argument for NP (and standard cryptographic assumptions) in order to construct a constant-round, quantumly-extractable classical commitment scheme. We start with the definition, and proceed to the construction.

Definition 4.1 (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 $m_T := \bot$ 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.

Remark 4.1. In the standard definition of extraction and more broadly, of simulation, the simulator does not output the interaction transcript (in classical-interaction protocols). It is noted however that it can be assumed without the loss of generality that the simulator also outputs the simulated transcript whenever needed. This is because, given a classical (or quantum) interactive circuit, it can be compiled in polynomial time (in the circuit size) to a circuit with identical functionality, that records the interaction transcript into its private inner state. Since the simulator simulates the inner state of the adversary at the end of interaction it in particular simulates the transcript.

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 $(\mathsf{P_{NP}}, \mathsf{V_{NP}})$ 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.

Binding and Hiding. The perfect binding property of the scheme follows readily from the perfect binding of the non-interactive commitment scheme Com. We next show hiding.

Proposition 4.1 (The Commitment Scheme is Computationally 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 $\lambda \in \mathbb{N}$*,* $m_0, m_1 \in \{0,1\}^{\ell(\lambda)}$*.*

Proof. We prove the claim by a hybrid argument. Define the following hybrid distributions on transcripts.

- Hyb_0 : This is the output distribution $\mathsf{VIEW}_{\mathsf{Rec}^*}\langle\mathsf{Sen}(m_0), \mathsf{Rec}^*(\rho)\rangle$.
- Hyb_1 : The output distribution of a process that acts like Hyb_0, with the exception that in step 5, instead of Sen communicating with Rec^* to give a ZK argument, we take the ZK simulator Sim of the argument system $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ and use it to simulate the argument by Sen, by executing $\mathsf{Sim}(T', \mathsf{Rec}^*, \rho')$, where T' (resp. ρ') is the transcript (resp. inner quantum state of Rec^*) generated at the end of step 4 of the interaction.
- Hyb_2 : The output distribution of a process that acts like Hyb_1, with the exception that in step 4b, instead of actually performing an SFE evaluation of $C_{1\to m_0}$, the process performs an SFE evaluation of the circuit $C_\perp$ that always outputs $\perp$.
- Hyb_3 : The output distribution of a process that acts like Hyb_2, with the exception that in step 1, instead of committing to m_0, the sender commits to m_1.
- Hyb_4 : The output distribution of a process that acts like Hyb_3, with the exception that in step 4b, the process performs an SFE evaluation of the circuit $C_{1\to m_1}$, and not of the circuit $C_\perp$.
- Hyb_5 : The output distribution of a process that acts like Hyb_4, with the exception that in step 5, instead of using the ZK simulator for the sender's argument, the process uses the ZK argument regularly, that is, the sender proves that the transcript so far is consistent. Observe that this is exactly the output distribution $\mathsf{VIEW}_{\mathsf{Rec}^*}\langle\mathsf{Sen}(m_1), \mathsf{Rec}^*(\rho)\rangle$.

We now explain why each consecutive pair of distributions are computationally indistinguishable, and our proof is finished.

- $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$: Follows from the post-quantum zero-knowledge property of the protocol $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$.
- $\mathsf{Hyb}_1 \approx_s \mathsf{Hyb}_2$: Assume toward contradiction that the two distributions are distinguishable, and fix, by an averaging argument, the partial transcript T' (and inner state of Rec^*) that is generated at the end of step 2 of the protocol and maximizes the distinguishing advantage between the two distributions.

  We consider two cases for the commitment $\mathsf{cmt}_{\mathsf{Rec}}$ in the transcript T': The simpler case is if $\mathsf{cmt}_{\mathsf{Rec}}$ is not a commitment to 0 (i.e. there is no $r_0 \in \{0,1\}^*$ s.t. $\mathsf{cmt}_{\mathsf{Rec}} = \mathsf{Com}(1^\lambda, 0; r_0)$), in that case, by the soundness of the argument that Rec^* gives in step 3, with overwhelming probability Sen is going to reject the proof and end communication, and only with a negligible probability the process continues to a point where the two processes $\mathsf{Hyb}_1, \mathsf{Hyb}_2$ differ, in contradiction.

  In the second case $\mathsf{cmt}_{\mathsf{Rec}}$ is a valid commitment to 0. In that case, the contradiction follows from (an implication of) the circuit privacy property of the SFE encryption, specifically, it follows from

Claim 2.2. From the perfect binding of the non-interactive commitment scheme Com, there is no string r_1 s.t. $\mathsf{cmt}_{\mathsf{Rec}} = \mathsf{Com}(1^\lambda, 1; r_1)$, which in turn implies that the circuit $C_{1\to m_0}$ is identical in functionality to the circuit $C_\perp$ that outputs $\perp$ on any input. By Claim 2.2 it follows that the responses from Sen in step 4b are statistically indistinguishable, and thus also the distributions Hyb_1 and Hyb_2, again in contradiction.

- $\mathsf{Hyb}_2 \approx_c \mathsf{Hyb}_3$: Follows from the hiding of the commitment $\mathsf{cmt}_{\mathsf{Sen}}$ that Sen gives in step 1, that is, the hiding property of the commitment scheme Com.
- $\mathsf{Hyb}_3 \approx_s \mathsf{Hyb}_4$: This indistinguishability follows from the exact same reasoning as in the explanation for the indistinguishability $\mathsf{Hyb}_1 \approx_s \mathsf{Hyb}_2$, by swapping m_0 with m_1 in the explanation.
- $\mathsf{Hyb}_4 \approx_c \mathsf{Hyb}_5$: Like the indistinguishability $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$, this indistinguishability follows again from the zero-knowledge property of the argument system $(\mathsf{P}_{\mathsf{NP}}, \mathsf{V}_{\mathsf{NP}})$.

□

4.1 Extractability

We show a quantum polynomial-time extractor Ext s.t. for any polynomial-size quantum sender $\mathsf{Sen}^* = \{\mathsf{Sen}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ extracts the sender's committed message (if it exists) and also simulates its quantum state at the end of the protocol.

$\mathsf{Ext}(1^\lambda, \mathsf{Sen}^*, \rho)$:

1. **Simulation of Commitments:** Sen^* outputs $\mathsf{cmt}_{\mathsf{Sen}}$. Ext then sends to Sen^* a commitment to 1: $\mathsf{cmt}_{\mathsf{Ext}} = \mathsf{Com}(1^\lambda, 1; r_1)$, where $r_1 \in \{0,1\}^{\mathrm{poly}(\lambda,1)}$ is the random string used as the randomness of the commitment algorithm.
2. **Simulation of ZK Argument by Rec:** Ext uses the zero-knowledge simulator Sim of the argument system $(\mathsf{P}_{\mathsf{NP}}, \mathsf{V}_{\mathsf{NP}})$. Ext executes $\mathsf{Sim}(\mathsf{cmt}_{\mathsf{Ext}}, \mathsf{Sen}^*, \rho^{(1)})$ to simulate the argument that Rec gives to Sen^* at step 3 of the protocol ($\rho^{(1)}$ is the inner state of Sen^* after step 1 of the extraction). At the end of the zero-knowledge simulation, we have a simulated argument transcript and a quantum state ρ' for Sen^* to carry on to the next step of extraction.
3. **Extraction of Message from Sen^*:**
   - Ext computes $\mathsf{dk} \leftarrow \mathsf{SFE.Gen}(1^\lambda)$ and sends $\mathsf{ct}_{\mathsf{Ext}} \leftarrow \mathsf{SFE.Enc}_{\mathsf{dk}}(r_1)$.
   - Sen^* outputs a response $\hat{\mathsf{ct}}$.

   Ext then decrypts the evaluated ciphertext to get a message m'.
4. **ZK Argument by Sen^*:** Ext takes the role of the honest receiver Rec in the ZK argument Sen^* gives.
5. **Extraction Procedure Output:** The output $(\sigma_{\mathsf{Ext}}, m_{\mathsf{Ext}})$ of the extraction procedure is as follows.
   - The simulated inner state σ_{Ext} for the sender is set to be the inner state of Sen^* at the time of halting of the procedure.
   - If the argument that Sen^* gave in step 4 of the procedure is convincing then $m_{\mathsf{Ext}} = m'$, otherwise $m_{\mathsf{Ext}} = \perp$.

It remains to explain why the extraction process yields an output that is computationally indistinguishable from a tuple (T, σ, m_T) generated by the real interaction between $\mathsf{Sen}^*(\rho)$ and Rec, and also that the extracted message m_{Ext} is indeed the message that T_{Ext} can be decommitted to.

Proposition 4.2. *Let* $\mathsf{Sen}^* = \{\mathsf{Sen}^*_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ *be a polynomial-size quantum sender, then,*

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

Proof. We prove the claim by a hybrid argument. Define the following hybrid processes:

- Hyb_0 : This distribution is the output distribution (σ, m_T) of the real interaction $\langle \mathsf{Sen}^*(\rho), \mathsf{Rec}\rangle$.

- Hyb_1 : The output distribution of a process that acts like Hyb_0, with the exception that in step 3, instead of Rec communicating with Sen^* to give a ZK argument, we take the ZK simulator Sim of the argument system $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ and use it to simulate the argument by Rec by executing $\mathsf{Sim}(T', \mathsf{Sen}^*, \rho')$, where T' (resp. ρ') is the transcript (resp. inner quantum state of Rec^*) generated at the end of step 2 of the interaction.

- Hyb_2 : The output distribution of a process that acts like Hyb_1, with the exception that when Rec sends $\mathsf{cmt}_{\mathsf{Rec}}$, it commits to 1 instead of to 0.

- Hyb_3 : The output distribution of a process that acts like Hyb_2, with the exception that in step 4a, Rec sends an SFE encryption $\mathsf{ct}_{\mathsf{Rec}}$ of the randomness r_1 that it used in step 2 when it committed for 1. Note that this output distribution is identical to the extraction's output $\mathsf{Ext}(1^\lambda, \mathsf{Sen}^*, \rho)$, with the only change being that m_T is generated as in $\langle \mathsf{Sen}^*(\rho), \mathsf{Rec}\rangle$.

- Hyb_4 : The output distribution of a process that acts like Hyb_3, with the exception that the output message m_T is generated differently, specifically, m_T is m_{Ext} that is generated as in step 5 of the extraction procedure. Note that this process is exactly the output distribution $(\sigma_{\mathsf{Ext}}, m_{\mathsf{Ext}}) \leftarrow \mathsf{Ext}(1^\lambda, \mathsf{Sen}^*, \rho)$.

We now explain why each pair of consecutive distributions are computationally indistinguishable, and our proof is finished.

- $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$: Assume toward contradiction that the two distributions are distinguishable, and fix, by an averaging argument, the partial transcript T' and inner quantum state σ' of Sen^* generated at the end of step 1 of the simulation, that maximizes the distinguishing advantage of the two distributions. Inside such transcript T' we consider the sender commitment $\mathsf{cmt}_{\mathsf{Sen}}$, and the (unique, by the perfect binding of the commitment scheme Com) message $m_{T'}$ that is inside this commitment (if the commitment cannot be opened to any message, $m_{T'} := \perp$).

  From our assumption that $\mathsf{Hyb}_0, \mathsf{Hyb}_1$ are distinguishable, follows the existence of a distinguisher that breaks the zero-knowledge property of $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$. Specifically, the distinguisher uses as non-uniform advice the partial transcript T' and the message $m_{T'}$, gets either a real interaction transcript or a simulation of the argument that Rec gives in step 3 of the protocol, then executes the rest of the commitment protocol, and uses the knowledge $m_{T'}$ at the end of protocol execution to output m_T. It follows that such distinguisher breaks the zero knowledge property of $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$, in contradiction.

  In the following explanations for the indistinguishabilities we will use the same averaging argument and non-uniform advice that includes the message $m_{T'}$, and refer to it simply as the "averaging argument with non-uniform advice message".

- $\mathsf{Hyb}_1 \approx_c \mathsf{Hyb}_2$: Follows from the same averaging argument and non-uniform advice message reasoning from the proof of $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$, along with the hiding of the commitment scheme Com.

- $\mathsf{Hyb}_2 \approx_c \mathsf{Hyb}_3$: Follows from the same averaging argument and non-uniform advice message reasoning from the proof of $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$, along with the input privacy (encryption security) property of the SFE encryption.

- $\mathsf{Hyb}_3 \approx_s \mathsf{Hyb}_4$: Recall that both processes Hyb_3, Hyb_4 generate the output state σ_{Ext} as in the extraction procedure, but differ only in the way they generate the output message. Assume toward contradiction that Hyb_3, Hyb_4 are distinguishable and fix, by an averaging argument, the partial transcript T' (and inner state σ' of Sen^*) generated at the end of step 3 of the extraction.

  Consider two cases for the partial transcript T'.

  - T' is consistent. In that case, by the (perfect) correctness of the SFE evaluation it follows that the generated messages m_T (from Hyb_3) and m_{Ext} (from Hyb_4) are identical. The rest of the protocol execution, which includes only the argument by Sen^*, is also identical between the two distributions. The distinguisher between Hyb_3, Hyb_4 (we assumed toward contradiction exists) cannot distinguish between these two distributions as they are identical, in contradiction.
  - T' is not consistent. In that case, by the soundness of the argument system $(\mathsf{P}_{\mathsf{NP}}, \mathsf{V}_{\mathsf{NP}})$, the argument by Sen^* fails with overwhelming probability, and with the same probability the values of both m_T (from Hyb_3) and m_{Ext} (from Hyb_4) are $\perp$. It follows that the statistical distance between the distributions is negligible, in contradiction.

□

5 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 2.7 where honest parties are polynomial-time and quantum (prover is efficient given a quantum witness) and communication is quantum.

The construction uses constant-round (post-quantum) zero-knowledge arguments for NP, quantumly-extractable commitments and a quantum sigma protocol for QMA[5].

We now proceed to the construction and proof.

Ingredients and notation:

- A constant-round quantumly-extractable commitment scheme $(\mathsf{Sen}, \mathsf{Rec})$.

- A constant-round post-quantum zero-knowledge argument system $(\mathsf{P}_{\mathsf{NP}}, \mathsf{V}_{\mathsf{NP}})$ for NP.

- A quantum sigma protocol for QMA $(\Xi.\mathsf{P}, \Xi.\mathsf{V})$.

We describe the protocol in Figure 3.

[5] In a previous version of this work we used the QMA zero-knowledge (with large soundness error) protocol of [BJSW16] instead of sigma protocols. Using sigma protocols yields a simplified protocol.

5.1 Computational Soundness

We prove that Protocol Protocol 3 has quantum computational soundness.

Proposition 5.1. *For any quantum polynomial-size prover* $\mathsf{P}^* = \{\mathsf{P}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$*, there exists a negligible function* $\mu(\cdot)$ *such that for any security parameter* $\lambda \in \mathbb{N}$ *and any* $x \in \{0,1\}^\lambda \setminus \mathcal{L}$*,*

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

Proof. Let $\mathsf{P}^* = \{\mathsf{P}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ a polynomial-size quantum prover and let $x = \{x_\lambda\}_{\lambda\in\mathbb{N}}$ be a sequence such that $\forall\lambda \in \mathbb{N} : x_\lambda \in \{0,1\}^\lambda \setminus \mathcal{L}$. We prove soundness by a hybrid argument. We consider a series of hybrid processes with output over $\{0,1\}$, starting from $\mathsf{OUT}_\mathsf{V}\langle \mathsf{P}^*(\rho), \mathsf{V}\rangle(x)$ the output distribution of V in the interaction with P^*.

- Hyb_0 : The output distribution of $\mathsf{OUT}_\mathsf{V}\langle \mathsf{P}^*_\lambda(\rho_\lambda), \mathsf{V}\rangle(x_\lambda)$.
- Hyb_1 : Identical to the process Hyb_0, with the exception that in step 3b when the verifier gives a ZK argument, the process instead uses the ZK simulator Sim of the argument system $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$. To simulate the prover's view, the process executes $\mathsf{Sim}\left((T_{\mathsf{Sen}}, \beta), \mathsf{P}^*, \rho'\right)$, where ρ' is the inner quantum state of P^* at the end of step 3a where the verifier sends β.
- Hyb_2 : Identical to the process Hyb_1, with the exception that in step 1 when the verifier commits to β, the process instead commits to $0^{|\beta|}$.

We next explain why each consecutive pair of distributions are indistinguishable.

- $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_1$: Follows from the quantum zero knowledge property of the protocol $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$.
- $\mathsf{Hyb}_1 \approx_c \mathsf{Hyb}_2$: Follows from the computational hiding of the commitment scheme $(\mathsf{Sen}, \mathsf{Rec})$.

Now, assume toward contradiction that P^* succeeds in making the verifier accept with some noticeable probability $\varepsilon(\lambda)$, that is, the probability for the output 1 in Hyb_0 is noticeable. $\mathsf{Hyb}_0 \approx_c \mathsf{Hyb}_2$, and thus the probability for the output 1 in Hyb_2 is also noticeable. Finally, we get a contradiction to the soundness of the sigma protocol $(\Xi.\mathsf{P}, \Xi.\mathsf{V})$, by using the prover sigma protocol messages from steps 2, 4 as messages to convince a quantum sigma protocol verifier $\Xi.\mathsf{V}$. Since the probability that the verifier V is convinced in Hyb_2 is noticeable, and such verifier is convinced if and only if the sigma protocol verifier is convinced, we get our contradiction. □

5.2 Computational Zero Knowledge

We prove that Protocol Protocol 3 is quantum computational zero knowledge.

We describe a universal simulator Sim for the protocol. We denote by $\mathsf{V}^* = \{\mathsf{V}^*_\lambda, \rho_\lambda\}_{\lambda\in\mathbb{N}}$ a polynomial-size quantum verifier. The simulator takes as input an instance in the language $x \in \{0,1\}^\lambda \cap \mathcal{L}$, a verifier circuit V^*_λ and quantum auxiliary input ρ_λ for V^*_λ. Subscripts are dropped when are clear from the context.

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

1. **Extraction of Message from Verifirer:** Sim executes the extractor Ext of the extractable commitment scheme $(\mathsf{Sen}, \mathsf{Rec})$. Sim computes a simulation of the commitment interaction transcript, inner state at the end of interaction and extracted message $(T_{\mathsf{Ext}}, \sigma_{\mathsf{Ext}}, \beta_{\mathsf{Ext}}) \leftarrow \mathsf{Ext}(1^\lambda, \mathsf{V}^*, \rho)$ and uses the simulated state σ_{Ext} as inner state for V^* in order to continue the protocol simulation[6].

[6] By the standard definition, the extractor Ext simulates only the state and extracted message $(\sigma_{\mathsf{Ext}}, m_{\mathsf{Ext}})$, but recall we can assume without the loss of generality that it also simulates the commitment transcript T_{Ext} (see Remark 4.1), and the triplet is indistinguishable from $(T, \sigma, m_T) \leftarrow \langle \mathsf{Sen}^*(\rho), \mathsf{Rec}\rangle(1^\lambda)$.

2. **Sigma Protocol First Part Simulation:** Sim executes $(\alpha_{\mathsf{Sim}}, \gamma_{\mathsf{Sim}}) \leftarrow \Xi.\mathsf{Sim}(x, \beta_{\mathsf{Ext}})$ and sends α_{Sim}.

3. **Malicious Verifier Challenge and ZK Argument:** Sim takes the role of the honest prover P when the verifier sends β and gives a ZK argument that $\exists r \in \{0,1\}^* : 1 = \mathsf{VDcom}(T_{\mathsf{Ext}}, \beta, r)$. If the argument was not convincing the simulator halts and concludes simulation.

4. **Sigma Protocol Second Part Simulation:** Sim sends γ_{Sim} and concludes simulation.

It remains to prove that the simulator's output is computationally indistinguishable from the verifier's output in the real interaction.

Proposition 5.2. *For any polynomial-size quantum verifier* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$,

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

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

Proof. We prove the claim by a hybrid argument, specifically, we consider hybrid distributions, all of which will be computationally indistinguishable.

- Hyb_0 : The output distribution of the simulator $\mathsf{Sim}(x, \mathsf{V}^*, \rho)$.

- Hyb_1 : Identical to the process Hyb_0, except that we erase some extreme cases from the output distribution, by making a check. Specifically, in step 3 when the verifier sends β and a ZK argument, if $\beta_{\mathsf{Ext}} \neq \beta$ and also the argument by V^* was convincing, the output of the process is $\perp$.

- Hyb_2 : Identical to the process Hyb_1, except that in steps 2, 4 where the simulator sends α_{Sim} and γ_{Sim}, the process instead uses the real sigma protocol prover to generate the messages, $(\alpha, \tau) \leftarrow \Xi.\mathsf{P}_1(x, w^{\otimes k(\lambda)})$, $\gamma \leftarrow \Xi.\mathsf{P}_3(\beta_{\mathsf{Ext}}, \tau)$.

- Hyb_3 : Identical to the process Hyb_2, except that when computing the the last sigma protocol message $\gamma \leftarrow \Xi.\mathsf{P}_3(\beta_{\mathsf{Ext}}, \tau)$, the process uses the β that V^* sent instead of the extracted β_{Ext}, that is, $\gamma \leftarrow \Xi.\mathsf{P}_3(\beta, \tau)$.

- Hyb_4 : Identical to the process Hyb_3, except that the check described in Hyb_1 is not performed, that is, even if the extracted challenge β_{Ext} and the challenge β sent by V^* are distinct and the ZK argument by V^* succeeds, the process carries on to the last step 4 and does not outputs $\perp$.

- Hyb_5 : At this point in our series of hybrid distributions we do not use the extracted challenge β_{Ext}, and we would like to move to a final process that does not use extraction at all. This process is identical to Hyb_4, with the exception that in step 1 of the simulation, where the simulator executes Ext to simulate the transcript and inner state of V^*, the process simply executes the real interaction between V^* and Rec, $(T, \sigma) \leftarrow \langle \mathsf{V}^*(\rho), \mathsf{Rec}\rangle(1^\lambda)$. Observe that Hyb_5 is exactly the real interaction output $\mathsf{OUT}_{\mathsf{V}^*}\langle \mathsf{P}(w^{\otimes k}), \mathsf{V}^*(\rho)\rangle(x)$.

Before proving that each consecutive pair of hybrids is indistnguishable,

We prove why each consecutive pair of distributions are computationally indistinguishable, and our proof is finished.

- $\mathsf{Hyb}_0 \approx_s \mathsf{Hyb}_1$: To show the indistinguishability we need to prove that the probabilistic event that exists in Hyb_0 but is erased in Hyb_1 happens with a negligible probability. This is exactly the statement proven in Claim 5.1.

- $\mathsf{Hyb}_1 \approx_c \mathsf{Hyb}_2$: This indistinguishability follows from the special zero knowledge property of the quantum sigma protocol.

- $\mathsf{Hyb}_2 \equiv \mathsf{Hyb}_3$: Due to the fact that in both hybrid processes, whenever $\beta_{\mathsf{Ext}} \neq \beta$ the process halts and outputs $\perp$, it is always the case that the first prover sigma protocol message γ is computed with respect to the sent β.

- $\mathsf{Hyb}_3 \approx_s \mathsf{Hyb}_4$: The reasoning for this indistinguishability is identical to the reasoning for the indistinguishability $\mathsf{Hyb}_0 \approx_s \mathsf{Hyb}_1$, and follows from Claim 5.1.

- $\mathsf{Hyb}_4 \approx_c \mathsf{Hyb}_5$: This indistinguishability follows from the extractability property (in Definition 4.1) of the commitment scheme $(\mathsf{Sen}, \mathsf{Rec})$.

□

Claim 5.1 (Extracted Information is Correct Under an Argument). *Let* $\mathsf{V}^* = \{\mathsf{V}^*_\lambda, \rho_\lambda\}_{\lambda \in \mathbb{N}}$ *a polynomial-size quantum verifier. Consider the process of interaction between* $\mathsf{V}^*(\rho)$ *and* P *in the original protocol, with one change: when* V^* *gives an extractable commitment, instead of executing the interaction* $(T, \sigma) \leftarrow \langle \mathsf{V}^*(\rho), \mathsf{Rec}\rangle(1^\lambda)$*, the process executes the extractor* $(T_{\mathsf{Ext}}, \sigma_{\mathsf{Ext}}, \beta_{\mathsf{Ext}}) \leftarrow \mathsf{Ext}(1^\lambda, \mathsf{V}^*, \rho)$*. Then, there is some negligible function* negl *such that,*

$$\Pr\left[(\beta \neq \beta_{\mathsf{Ext}}) \wedge (\mathsf{V}^* \textit{ gives a convincing argument})\right] \leq \mathrm{negl}(\lambda) \enspace .$$

Proof. Let T_{Sen} be the transcript generated at the end of the extractable commitment protocol, in the original interaction between V^* and P. By the perfect binding of the commitment scheme $(\mathsf{Sen}, \mathsf{Rec})$ it follows that if the statement from V^*'s ZK argument is correct, that is, there is some $r \in \{0,1\}^*$ s.t. $1 = \mathsf{VDcom}(T_{\mathsf{Sen}}, \beta, r)$, then β is necessarily the committed message, in symbols (denoted in the binding property in Definition 4.1) $\beta = m_{T_{\mathsf{Sen}}}$. It follows from the soundness of the argument that V^* gives, that only with a negligible probability $\mathrm{negl}'(\lambda)$ it happens that both, $\beta \neq m_{T_{\mathsf{Sen}}}$, and V^* gives a convincing argument.

Recall that by the extractability property of the commitment scheme (Extractability property in Definition 4.1), the following two distributions are indistinguishable,

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

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

This means that when considering the process described in this claim's statement, where extraction takes place (instead of executing the commitment procedure), only with some negligible probability $\mathrm{negl}(\lambda)$ it can happen that both, $\beta \neq \beta_{\mathsf{Ext}}$, and V^* gives a convincing argument, this is because if this probability wasn't negligible we would be able to break the extractability property of the commitment scheme $(\mathsf{Sen}, \mathsf{Rec})$. □

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.

[BD18] Zvika Brakerski and Nico Döttling. Two-message statistically sender-private ot from lwe. In *Theory of Cryptography Conference*, pages 370–390. Springer, 2018.

[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.

[BKP18] Nir Bitansky, Yael Tauman Kalai, and Omer Paneth. Multi-collision resistance: a paradigm for keyless hash functions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 671–684, 2018.

[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.

[CVZ19] Andrea Coladangelo, Thomas Vidick, and Tina Zhang. Non-interactive zero-knowledge arguments for qma, with preprocessing. *arXiv preprint arXiv:1911.07546*, 2019.

[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.

[GHKW17] Rishab Goyal, Susan Hohenberger, Venkata Koppula, and Brent Waters. A generic approach to constructing and proving verifiable random functions. In *Theory of Cryptography - 15th International Conference, TCC 2017, Baltimore, MD, USA, November 12-15, 2017, Proceedings, Part II*, pages 537–566, 2017.

[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.

[GMW91] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. *Journal of the ACM (JACM)*, 38(3):690–728, 1991.

[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.

[LS19] Alex Lombardi and Luke Schaeffer. A note on key agreement and non-interactive commitments. *IACR Cryptology ePrint Archive*, 2019:279, 2019.

[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.

[OPCPC14] Rafail Ostrovsky, Anat Paskin-Cherniavsky, and Beni Paskin-Cherniavsky. Maliciously circuit-private fhe. In *Annual Cryptology Conference*, pages 536–553. Springer, 2014.

[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.

[PS19] Chris Peikert and Sina Shiehian. Noninteractive zero knowledge for np from (plain) learning with errors. In *Annual International Cryptology Conference*, pages 89–114. Springer, 2019.

[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.

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
   $$\mathsf{pk},\ \ \mathsf{ct}_\mathsf{V} \leftarrow \mathsf{QHE.Enc}_\mathsf{pk}(t),\ \ \widetilde{\mathbf{CC}} \leftarrow \mathsf{Obf}\Big(\mathbf{CC}\big[\mathsf{QHE.Dec}_\mathsf{sk}(\cdot), s, (r,\beta)\big]\Big)\ .$$
   (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 $\hat{\mathsf{ct}} \leftarrow \mathsf{SFE.Eval}\Big(\mathbf{CC}\big[\mathsf{Id}(\cdot), t, s\big],\ \mathsf{ct}_\mathsf{P}\Big)$, where $\mathsf{Id}(\cdot)$ is the identity function.
3. **Sigma Protocol Execution:**
   (a) P computes $(\alpha, \tau) \leftarrow \Sigma.\mathsf{P}_1(x, w)$ 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 $\gamma = \Sigma.\mathsf{P}_3(\beta, \tau)$.
7. **Acceptance:** V accepts if $\Sigma.\mathsf{V}(\alpha, \beta, \gamma) = 1$.
8. **Reactions to Aborts:** During the protocol, if either party sends a message of an incorrect form or provides a non-convincing WI proof, the other party terminates the interaction.

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

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 interacts with Sen through $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ to give a ZK argument that $\mathsf{cmt}_{\mathsf{Rec}}$ is indeed a commitment to 0, that is, there exists randomness $r_0 \in \{0,1\}^{\mathrm{poly}(\lambda,1)}$ string[a] s.t. $\mathsf{cmt}_{\mathsf{Rec}} = \mathsf{Com}(1^\lambda, 0; r_0)$.
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 $r_1 \in \{0,1\}^{\mathrm{poly}(\lambda,1)}$ 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 interacts with Rec through $(\mathsf{P}_{\mathrm{NP}}, \mathsf{V}_{\mathrm{NP}})$ to give a ZK argument for the statement that its transcript until the end of step 4b is consistent, that is, there exists a message and randomness for the honest sender algorithm Sen that generates the transcript.

[a] Let $\mathrm{poly}(\lambda, \ell)$ denote the polynomial that represents the amount of randomness the commitment algorithm $\mathsf{Com}(\cdot)$ needs for security parameter λ and message length ℓ.

Figure 2: A Quantumly-Extractable Classical Commitment Scheme.

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 $\mathcal{L} \in \mathbf{QMA}$.