A New Approach to Arguments of Quantum Knowledge

James Bartusek [*] Ruta Jawale[†] Justin Raizes[‡] Kabir Tomer[§]

Abstract

We construct a publicly-verifiable non-interactive zero-knowledge argument system for QMA with the following properties of interest.

- **Transparent setup.** Our protocol only requires a uniformly random string (URS) setup. The only prior publicly-verifiable NIZK for QMA (Bartusek and Malavolta, ITCS 2022) requires an *entire obfuscated program* as the common reference string.
- **Extractability.** Valid QMA witnesses can be extracted directly from our accepting proofs. That is, we obtain a publicly-verifiable non-interactive argument of *quantum knowledge*, which was previously only known in a privately-verifiable setting (Coladangelo, Vidick, and Zhang, CRYPTO 2020).

Our construction introduces a novel type of ZX QMA verifier with "strong completeness" and builds upon the coset state authentication scheme from (Bartusek, Brakerski, and Vaikuntanathan, STOC 2024) within the context of QMA verification. Along the way, we establish new properties of the authentication scheme.

The security of our construction rests on the heuristic use of a post-quantum indistinguishability obfuscator. Rather than rely on the full-fledged classical oracle model (i.e. ideal obfuscation), we isolate a particular game-based property of the obfuscator that suffices for our proof, which we dub the *evasive composability* heuristic.

As an additional contribution, we study a general method for replacing heuristic use of obfuscation with heuristic use of hash functions in the post-quantum setting. In particular, we establish security of the ideal obfuscation scheme of Jain, Lin, Luo, and Wichs (CRYPTO 2023) in the *quantum* pseudorandom oracle model (QPrO), which can be heuristically instantiated with a hash function. This gives us NIZK arguments of quantum knowledge for QMA in the QPrO, and additionally allows us to translate several quantum-cryptographic results that were only known in the classical oracle model to results in the QPrO.

[*]Columbia University.
[†]UIUC.
[‡]NTT Research.
[§]UIUC. Part of this work was done while the authors were visiting the Simons Institute for the Theory of Computing

Contents

1 Introduction **3**
- 1.1 Results . . . 4

2 Technical Overview **7**
- 2.1 Approach . . . 7
- 2.2 The protocol and analysis . . . 9
- 2.3 The quantum pseudorandom oracle model . . . 12

3 Preliminaries **13**
- 3.1 Statistics . . . 13
- 3.2 Quantum Information . . . 14
- 3.3 Estimating Quantum Acceptance Probabilities . . . 15
- 3.4 Quantum Complexity . . . 17
- 3.5 Encryption . . . 18
- 3.6 NIZK for NP . . . 20
- 3.7 Obfuscation . . . 21
- 3.8 The (Q)PrO model . . . 22

4 QMA Verification with Strong Completeness **26**
- 4.1 Permuting ZX Verifier for QMA . . . 29

5 Coset State Authentication **30**
- 5.1 Construction . . . 31
- 5.2 Properties . . . 31

6 Post-Quantum NIZK Arguments of Knowledge for NP **38**
- 6.1 Knowledge Soundness Definition . . . 38
- 6.2 Proofs of Knowledge for NP with CRS . . . 38
- 6.3 Adaptively Sound Arguments for NP with URS . . . 42
- 6.4 Arguments of Knowledge for NP with URS . . . 42

7 Provably-Correct Obfuscation **43**
- 7.1 Definition . . . 43
- 7.2 Construction in QPrO Model . . . 45

8 Security of the JLLW Obfuscator in the QPrO Model **52**
- 8.1 QPrO Key Reprogramming . . . 53
- 8.2 Proof of Security . . . 54

9 NIZK Arguments of Knowledge for QMA **61**
- 9.1 Definition . . . 61
- 9.2 Protocol . . . 63
- 9.3 Analysis . . . 65

References **75**

1 Introduction

Inspecting the proof of a mathematical statement generally reveals significantly more information than the fact that the statement is true. Remarkably, [BFM88] (building on an earlier *interactive* system [GMR89]) demonstrated that this need not always be the case, using cryptography to produce convincing proofs of NP statements that reveal nothing beyond the validity of the statement. This was the first example of a *non-interactive zero-knowledge* (NIZK) argument, which is now considered one of the most basic and natural cryptographic primitives with numerous applications throughout cryptography.

While it was established by [GO94] that any non-interactive argument for a language outside of BPP requires some form of setup, the widespread utility of NIZKs rests on the *public verifiability* of proofs with respect to the setup. That is, the setup algorithm must output some *public* information crs that can be used by any user in order to verify a purported proof π. Hence, a NIZK argument for some language $\mathcal{L}$ has the following syntax.

- $\mathsf{Setup}(1^\lambda) \to \mathsf{crs}$. The Setup algorithm produces some public parameters crs.
- $\mathsf{Prove}(\mathsf{crs}, x) \to \pi$. The Prove algorithm takes crs and an instance $x \in \mathcal{L}$, and produces a proof π of this fact.
- $\mathsf{Verify}(\mathsf{crs}, x, \pi)$. The Verify algorithm takes crs, an instance x, and a proof π, and either accepts or rejects.

Following [BFM88], the community has developed an impressive array of approaches (e.g. [GO94, FLS99, Fis05, GOS06, SW14, PS19, JJ21, Wat24, WWW25]) to obtaining NIZK arguments for NP with security based upon various cryptographic assumptions.

Quantumly-provable statements. Unfortunately, the situation changes dramatically if we expand the goal to proving mathematical statements in zero-knowledge whose proofs involve quantum information. That is, does there exist NIZK arguments for all of QMA? Perhaps surprisingly, given how well-understood NIZKs are in the classical setting, the existence of NIZKs for QMA from standard assumptions has remained wide open.

In fact, there is currently only one result in the literature [BM22] that proposes a candidate NIZK for QMA. This protocol builds on techniques from the area of *classical verification of quantum computation* (CVQC). In particular, the public parameters crs contain an *obfuscated program* that runs the CVQC verifier. In order to argue security, [BM22] model this obfuscation as a classical oracle, an approach that can also be used to establish the existence of publicly-verifiable CVQC. However, the existence of both NIZKs and QMA and publicly-verifiable CVQC from standard assumptions is still unknown, and it has remained unclear whether these applications are fundamentally linked.

Instead, the community has developed several protocols for a significantly weaker notion of NIZK for QMA that requires some form of *private* setup. For example, there exist several protocols from standard assumptions in the "secret parameters model", where we assume a trusted third-party that samples (potentially quantum) private randomness r_P for the prover and r_V for the verifier [ACGH20, CVZ20, Shm21, BCKM21, MY22] (we note that some only require private randomness for the verifier, but not the prover). The community has also considered the "shared EPR pair model", where we assume that the prover and verifer begin the protocol with several shared EPR

pairs [MY22, BKS23]. However, this model has the same drawbacks as the secret parameters model, as security relies on the assumption that these EPR pairs are set up correctly by an honest dealer, and they can only be used by the parties who receive them from the dealer.

Transparent setup. Even among the standard notion of publicly-verifiable NIZKs, there is an important distinction to make between *transparent* and *private-coin* setup. While a private-coin setup requires the Setup algorithm to sample crs from some structured distribution, a transparent setup only requires that crs is a uniformly random string. Thus, transparent setups are by far the easiest to realize in practice.

Examples of transparent setups include the *uniform reference string*, or URS, model,[1] and the *random oracle model* (ROM). By now, we have several approaches for realizing NIZKs for all of NP with transparent setup (e.g. [FLS99, Fis05, PS19]). However, as mentioned above, the only previously proposed NIZK for QMA [BM22] has a highly structured crs that includes an obfuscated program. Thus, the following question has remain unaddressed.

Do there exist non-interactive zero-knowledge arguments for QMA with transparent setup*?*

Knowledge soundness. In addition to minimizing the complexity of Setup, another important goal in the design of argument systems is to strengthen the soundness property. Traditionally, soundness guarantees that the prover cannot convince the verifier to accept a proof relative to any "no" instance. However, we often want to capture the idea that in order for a prover to produce a convincing proof (even of a "yes" instance), then they must *possess* a valid witness. This is formalized by requiring that an accepting witness can be *extracted* from any prover that manages to convince the verifier to accept its proof. Again, this property gives meaningful guarantees even when the statement to be proven is true, and has been broadly useful in the classical setting, for example in the area of anonymous credentials (e.g. [CvH91, CL01]).

While knowledge-soundness is again quite well-understood in the classical setting, we have far less convincing results in the quantum setting. The only non-interactive protocol that has been shown to have knowledge soundness is that of [CVZ20], which is in the secret parameters model. Thus, the following question has also been left unresolved.

Do there exist publicly-verifiable non-interactive zero-knowledge arguments of knowledge *for QMA?*

1.1 Results

In this work, we address both questions simultaneously by presenting a NIZK argument of quantum knowledge for QMA with transparent setup.

Informal Theorem 1.1. *Assuming any (post-quantum) NIZK argument of knowledge for NP with transparent setup, there exists a NIZK argument of quantum knowledge for QMA with transparent setup making heuristic use of a post-quantum obfuscator for classical computation.*

We observe that there exist post-quantum NIZK arguments of knowledge for NP in the URS model from LWE, by building on [PS19]. Thus, we obtain NIZK arguments of quantum knowledge for

[1] Sometime this is referred to as a *common random string*, or CRS, but this is often confused with the notion of a common *reference* string, which may be structured.

QMA in the URS model from standard cryptographic assumptions plus the heuristic use of a post-quantum obfuscator.

In fact, our protocol satisfies a very natural *straightline extraction* property, where if the extractor programs the URS, it is then able to directly recover a witness from the prover's proof. Moreover, our protocol has no knowledge error, and we are able to extract a QMA witness of quality $1-\gamma$ for some arbitrarily small inverse polynomial γ (see Section 2.2 for more discussion).

Evasive composability. Next, we provide more detail about the heuristic manner in which we use the classical obfuscator.

As discussed below in the technical overview, for much of the proof (of zero-knowledge) we use the standard indistinguishability property of the obfuscator. However, in one key step we resort to what we call the *evasive composability heuristic*:

Definition 1.2 (Evasive Composability Heuristic, simplified and informal)**.** Let Obf be an obfuscator, and $\mathcal{S}$ be any "non-contrived" sampler that outputs two classical circuits C_0, C_1 along with some (potentially quantum) side information $|\psi\rangle$.

IF for any QPT adversary $\mathcal{A}$ and each $b \in \{0,1\}$, it holds that

$$\left| \Pr_{|\psi\rangle, C_0, C_1 \leftarrow \mathcal{S}} [\mathcal{A}(|\psi\rangle, \mathsf{Obf}(C_b)) = 1] - \Pr_{|\psi\rangle, C_0, C_1 \leftarrow \mathcal{S}} [\mathcal{A}(|\psi\rangle, \mathsf{Obf}(\mathsf{NULL})) = 1] \right| = \mathsf{negl}(\lambda),$$

THEN it holds that

$$\left| \Pr_{|\psi\rangle, C_0, C_1 \leftarrow \mathcal{S}} [\mathcal{A}(|\psi\rangle, \mathsf{Obf}(C_0 \| C_1)) = 1] - \Pr_{|\psi\rangle, C_0, C_1 \leftarrow \mathcal{S}} [\mathcal{A}(|\psi\rangle, \mathsf{Obf}(\mathsf{NULL})) = 1] \right| = \mathsf{negl}(\lambda),$$

where NULL is the always-rejecting circuit, and $C_0 \| C_1$ is the "composed" circuit that maps $(b, x) \to C_b(x)$.

Informally, this heuristic asserts that obfuscating a circuit composed of sub-circuits whose obfuscations are indistinguishable from null, is itself indistinguishable from null. While this appears to be quite reasonable for natural choice of samplers $\mathcal{S}$, we remark that there do exist contrived samplers (involving "self-eating" circuits) that violate the statement [BGI+12]. However, in our case we only need evasive composability to hold for a specific class of samplers, which do not seem to possess the self-referential properties used in the [BGI+12] impossibility result. Moreover, it is easy to see that this heuristic holds for *any* choice of $\mathcal{S}$ in the classical oracle model, where Obf is modeled as a black-box.

Comparison with [BM22]. As mentioned earlier, there is only one other candidate publicly-verifiable NIZK for QMA [BM22], and it is worth comparing our results a little more closely. We consider our approach to have three key benefits over [BM22].

- Our protocol only requires a *uniformly random string* setup, while [BM22] requires a highly structured obfuscated program as the shared string.
- Our protocol satisfies a natural argument of quantum knowledge property with straightline extraction, which is not satisfied by [BM22].

- While both approaches make heuristic use of classical obfuscation, we isolate our heuristic use to the evasive composability heuristic, while it is unclear how to do so with [BM22]. We thus hope that our approach will yield more progress towards the long-standing goal of obtaining NIZKs for QMA from standard assumptions. In particular, our protocol does not rely on techniques that also yield publicly-verifiable CVQC (to the best of our knowledge), giving hope that NIZKs for QMA might be "easier" to construct than publicly-verifiable CVQC.

On the other hand, we remark that the [BM22] arguments are *classical* and *succinct*, while ours are *quantum*, and grow with the size of the witness, which allows us to establish the straightline knowledge extraction property. Thus, the results are strictly incomparable. Indeed, as we will see in the technical overview, our technical approach is completely different from that of [BM22].

The quantum pseudorandom oracle model. As it stands, both known approaches (ours and [BM22]) to NIZKs for QMA, as well as several other results in quantum cryptography, rely on the heuristic use of a classical obfuscator. However, if we are to rely on heuristic arguments, it is certainly in our interest to reduce the complexity of the object that we treat heuristically. For instance, it is a much more common practice in cryptography to make heuristic use of a simple *hash function*, which can instantiated with cryptographic hashes such as SHA2 or SHA3.

As a contribution of independent interest, we show that heuristic use of obfuscation in the *post-quantum* setting can often be replaced with heuristic use of a hash function. To do so, we adapt the recently-introduced *pseudorandom oracle*, or PrO, model [JLLW23] to the post-quantum setting.

The pseudorandom oracle model is defined with respect to some pseudorandom function $\{f_k\}_k$. It internally samples a uniformly random permutation π and presents the following interfaces:

- $\mathsf{PrO}(\mathsf{Gen}, k) \to \pi(k)$
- $\mathsf{PrO}(\mathsf{Eval}, h, x) \to f_{\pi^{-1}(h)}(x)$

That is, one can generate *handles* $\pi(k)$ corresponding to PRF keys k, which can be used to evaluate the function but reveal nothing about the key itself. As argued in [JLLW23], one can plausibly instantiate the PrO using a cryptographic hash function such as SHA3, where $\mathsf{PrO}(\mathsf{Gen}, k) \to k$ and $\mathsf{PrO}(\mathsf{Eval}, k, x) \to \mathsf{SHA3}(k, x)$. Although there is clearly a mismatch with the idealized model in that the permutation π is instantiated with the *identity*, [JLLW23] argue that this is justified based on the heuristic understanding that SHA3 behaves likes a "self-obfuscated" PRF. Thus, just like the *random* oracle model, we consider the pseudorandom oracle model to be a transparent setup assumption, as it can be plausibly instantiated using the public description of a cryptographic hash function.

Now, while [JLLW23]'s main result was to show how to construct ideal obfuscation for classical circuits in the PrO model from functional encryption, they did not address the *post-quantum* setting, where the PrO may be accessed in *quantum superposition*. In this work, we fill that gap, and prove the following result of independent interest.

Informal Theorem 1.3. *Assuming sub-exponentially secure functional encryption, there exists (post-quantum) ideal obfuscation in the quantum pseudorandom oracle (QPrO) model.*

As corollaries, we obtain several results in the QPrO that were previously only known in the full-fledged classical oracle model (e.g. witness encryption for QMA and publicly-verifiable CVQC [BM22], copy-protection for all unlearnable functionalities [ALL+21], obfuscation for various classes of quantum circuits [BKNY23, BBV24, HT25], and quantum fire [ÇGS25]).

Due to a non-black-box use of obfuscation in our NIZK argument, the above theorem doesn't immediately imply NIZKs of knowledge for QMA in the QPrO. However as we explain further in the technical overview, we do manage to show this result, encapsulated in the following theorem.

Informal Theorem 1.4. *Assuming (post-quantum) NIZK arguments of knowledge for NP with transparent setup and (post-quantum) sub-exponentially secure functional encryption, there exists NIZK arguments of knowledge for QMA in the QPrO model with transparent setup.*

2 Technical Overview

2.1 Approach

From a bird's eye view, our approach is fairly natural: Given a QMA instance x, witness $|\psi\rangle$, and a QMA verification measurement $\mathcal{M}$, the proof consists of an appropriate "encoding" of $|\psi\rangle$, an appropriate "obfuscation" of the measurement $\mathcal{M}$, and a NIZK (for NP) argument that the obfuscation and encoding have been prepared honestly.

However, it is not clear how to instantiate this approach. First of all, we don't currently have candidates for obfuscating arbitrary quantum measurements (let alone doing so in a provably-correct manner). Moreover, it is also in general unclear how to use a proof for NP (or even QMA!) to prove facts about quantum states, e.g. that the witness was encoded honestly.

Despite these obstacles, we show that a careful choice of the QMA verifier enables us to leverage techniques from obfuscation of quantum computation [BBV24] to achieve a publicly-verifiable NIZK (of knowledge) for QMA. In particular, we use the [BBV24] "coset-state authentication" scheme in order to encode the witness $|\psi\rangle$, which encodes each qubit according to the map

$$\mathsf{CSA.Enc}: \quad |0\rangle \to X^x Z^z |S\rangle\,, \quad |1\rangle \to X^x Z^z |S+\Delta\rangle\,,$$

where S is a random subspace of $\mathbb{F}_2^\lambda$, and x, z, Δ are random vectors. This encoding scheme admits a classical circuit $\mathsf{CSA.Ver}$ that can be used to *verify* membership in the codespace, as well as classical circuits $\mathsf{CSA.Dec}_0$ and $\mathsf{CSA.Dec}_1$ that can be used to *measure* the encoded state in the standard and Hadamard basis respectively.[2]

Before going further into the details, we sketch the high-level overview of our NIZK for QMA. Let $\mathcal{M}$ be a quantum verifier for a QMA promise problem $(\mathcal{L}_{yes}, \mathcal{L}_{no})$. We construct a protocol $(\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ with the following form.

- Setup outputs the commom random string crs for a NIZK for NP.
- P takes input the crs, an instance $x \in \mathcal{L}_{yes}$, and a corresponding quantum witness $|\psi\rangle$. It outputs an encoding of the witness $|\widetilde{\psi}\rangle = \mathsf{CSA.Enc}(|\psi\rangle)$, a classical obfuscation $\widetilde{\mathcal{V}}$ of the

[2] Strictly speaking, these classical circuits must be applied in *quantum superposition* in order to realize these verification and measurement functionalities.

codespace membership tester $\mathsf{CSA.Ver}$, a classical obfuscation of the quantum verifier $\widetilde{\mathcal{M}}$ (which will be derived from the $\mathsf{CSA.Dec}_0$ and $\mathsf{CSA.Dec}_1$ circuits in a manner described below), and a NIZK for NP proof π that the obfuscations were prepared honestly.

- V takes input the crs, the instance x, the encoded witness $|\widetilde{\psi}\rangle$, the obfuscations $\widetilde{\mathcal{V}}, \widetilde{\mathcal{M}}$, and proof π. It accepts iff (1) the NIZK for NP verifier accepts (crs, π), (2) the tester $\widetilde{\mathcal{V}}$ accepts $|\widetilde{\psi}\rangle$, and (3) the quantum verifier $\widetilde{\mathcal{M}}$ accepts $|\widetilde{\psi}\rangle$.

Before going further, we discuss some necessary background.

Background: ZX verifiers. We recall [BL08, CM16, MNS16] that any QMA language can be verified using just standard and Hadamard basis measurements, followed by some classical post-processing. It will be convenient for us to describe such a verifier's behavior as a collection of *coherent* ZX measurements on n qubits, where each ZX measurement is specified by a sequence of bases $\theta \in \{0,1\}^n$ and a function $f : \{0,1\}^n \to \{0,1\}$. That is, a ZX measurement $M[\theta, f]$ is defined by the projector

$$M[\theta, f] \coloneqq H^\theta \left(\sum_{x : f(x)=1} |x\rangle\langle x| \right) H^\theta,$$

and the QMA verifier is specified by some collection $\{M[\theta_i, f_i]\}_{i \in [N]}$ of ZX measurements. To verify a proof $|\psi\rangle$, it:

- Samples $i \leftarrow [N]$.
- Applies $\{\Pi[\theta_i, f_i], I - \Pi[\theta_i, f_i]\}$ to $|\psi\rangle$, and accepts if the measurement accepts.

Specifying the verifier in this manner is quite promising for instantiating our template above, as the CSA scheme supports ZX measurements on encoded states. However, note that the verifier doesn't only apply a ZX measurement – it first must *sample* the choice of measurement $i \leftarrow [N]$ that it will perform. This seemingly innocuous sampling step actually introduces a subtle issue in finalizing the instantiation of our template.

Who samples the randomness? One could imagine two ways to handle the sampling of $i \leftarrow [N]$ in our NIZK for QMA. One strategy would have the *prover* sample i, and then only send over an obfuscation of the CSA decoder for the particular choice of measurement $M[\theta_i, f_i]$. Unfortunately, this completely breaks soundness, as it may be possible for the prover to find some state $|\psi\rangle$ that is accepted by some *fixed* measurement $M[\theta_i, f_i]$ (even if there is no state $|\psi\rangle$ that is accepted with high probability over the random choice of measurement).

Another strategy would be to have the prover obfuscate CSA decoders for the *entire set* of ZX measurements, and then have the verifier choose which one to apply. Unfortunately, for traditional QMA verifiers (say, parallel repetition of XX/ZZ Hamiltonians), this completely breaks zero-knowledge, as there may exist two accepting witnesses $|\psi_0\rangle, |\psi_1\rangle$ and some choice of measurement $M[\theta_i, f_i]$ such that $M[\theta_i, f_i]$ accepts $|\psi_0\rangle$ and $|\psi_1\rangle$ with very different probabilities.

QMA verification with strong completeness. We resolve this tension by sticking with the second choice, but explicitly designing a ZX verifier that does not have this issue. In particular, we say that

a ZX verifier has *strong completeness* if for every yes instance, there exists a witness $|\psi\rangle$ such that for *all* choices of $i \in [N]$, it holds that

$$\| M[\theta_i, f_i] \, |\psi\rangle \, \|^2 = 1 - 2^{-n},$$

where n is the instance size. We also require negligible soundness, where for any no instance and state $|\psi\rangle$,

$$\mathop{\mathbb{E}}_{i \leftarrow [N]} \left[\| M[\theta_i, f_i] \, |\psi\rangle \, \|^2 \right] = \mathsf{negl}(n).$$

That is, we boost the completeness guarantee to $1 - 2^{-n}$ for every choice of measurement, rather than just on average over the choice of measurement. Formally, we show that every promise problem in QMA has a ZX verifier with strong completeness, which may be of independent interest. We accomplish this with what we call a "permuting QMA verifier".

Permuting Verifier for QMA. The permuting verifier is a modification of the standard parallel repetition amplification for QMA. Imagine that the verifier was given a large register which (allegedly) contained many copies of the same witness. Instead of sampling a measurement independently for each copy, the verifier starts with a fixed list of measurements containing each $M[\theta_i, f_i]$ many times, then permutes it randomly. Then, it applies the permuted list of measurements to the witness register and accepts if the majority of them accept.[3]

If the verifier really were given many copies of the same witness, then the permutation does not matter; the verifier is just applying each $M[\theta_i, f_i]$ many times to the same state. With a slight change in perspective, the verifier is simply estimating the outcome distribution of each measurement on the witness by performing it many times. This has high accuracy, so the verifier is convinced with overwhelming probability.

The case of soundness is more complicated. If each measurement were sampled independently at random, we could argue that each index constitutes its own QMA verifier, and so should reject with a fixed probability. Unfortunately, the measurements are highly correlated because there are a fixed number of each $M[\theta_i, f_i]$.

The saving grace is that the independent distribution is *heavily* concentrated around its mean. We can view the independent distribution as first sampling the number of times to apply each $M[\theta_i, f_i]$, then permuting the resulting list. The number of times each measurement appears in the independent list versus the fixed list is very close with high probability. If we were to replace each difference by a measurement that always accepts in the independent list, the number of modifications is very small relative to the overall size of the list. Finally, we can show that the number of accepting measurements can only increase by the number of replacements, which is not enough to bridge the gap between a NO instance and a YES instance. More details can be found in Section 4.

2.2 The protocol and analysis

We are now ready to present our protocol in some more detail, and then give high-level, informal, overviews of our proofs of knowledge-soundness and zero-knowledge.

[3] Technically speaking, there is some weighting of the measurements involved, which we ignore here for the sake of exposition.

Recall that we not only want to obfuscate the CSA algorithms, we also want to *prove* correctness of these obfuscations. For reasons that we expand on below in Section 2.3, we define an abstraction called *provably-correct obfuscation* which has all of the properties that we require.

Let Obf be a provably-correct obfuscation with respect to public parameters crs (concretely, think of crs as the public parameters for some NIZK of knowledge for NP). Let x be an instance, $\{M[\theta_i, f_i]\}_{i\in[N]}$ be the corresponding ZX verifier with strong completeness, and let $|\psi\rangle$ be a witness for x. Then our protocol operates as follows.

- The prover P takes input the crs and the witness $|\psi\rangle$. It computes $|\widetilde{\psi}\rangle = \mathsf{CSA.Enc}(|\psi\rangle)$, sets $\widetilde{\mathcal{V}} = \mathsf{Obf}(\mathsf{CSA.Ver})$, and sets

$$\widetilde{\mathcal{V}}, \{\widetilde{\mathcal{M}_i}\}_{i\in[N]} = \mathsf{Obf}\left(\mathsf{crs}, \mathsf{CSA.Ver} \| \{\mathsf{CSA.Dec}_i\}_{i\in[N]}\right),$$

  where we are obfuscating the *concatenation* of all $N+1$ programs, and parsing the resulting obfuscation as the part $\widetilde{\mathcal{V}}$ that can be used to evaluate CSA.Ver and the parts $\widetilde{\mathcal{M}_i}$ that can be used to evaluate each $\mathsf{CSA.Dec}_i$. Here, $\mathsf{CSA.Dec}_i$ refers to the CSA algorithm that measures the encoded state according to $M[\theta_i, f_i]$.

- V takes input the crs, the instance x, the encoded witness $|\widetilde{\psi}\rangle$, and the obfuscation $(\widetilde{\mathcal{V}}, \{\widetilde{\mathcal{M}_i}\}_i)$. It first checks that the obfuscation is well-formed using crs. Then, it checks that applying $\widetilde{\mathcal{V}}$ to $|\widetilde{\psi}\rangle$ accepts. Finally, it accepts if $\widetilde{\mathcal{M}_i}$ accepts $|\widetilde{\psi}\rangle$ in expectation over the choice of $i \leftarrow [N]$.

Argument of quantum knowledge. We first show that our protocol satisfies a straightline (i.e. non-rewinding) argument of quantum knowledge property. To do so, we require that the provably-correct obfuscation satisfies an extraction property, namely that there exists an extractor that, given an obfuscated program, can extract the description of the plaintext program. Given this ability, our QMA extractor is quite natural, and consists of two parts $(\mathsf{Ext}_0, \mathsf{Ext}_1)$.

- Ext_0 outputs the public parameters crs along with a trapdoor td for the provably-correct obfuscation extractor.
- Ext_1 takes input $\left(\mathsf{td}, \pi = (\,|\widetilde{\psi}\rangle\,, \widetilde{\mathcal{V}}, \{\widetilde{\mathcal{M}_i}\}_i)\right)$. It uses the provably-correct obfuscation extractor to extract the description of CSA.Ver, which contains a description of the CSA *authentication key* k. Given k, it can undo the encoding on the state $|\widetilde{\psi}\rangle$ in order to obtain an underlying state $|\psi\rangle$.

It remains to analyze the extracted state $|\psi\rangle$. In particular, in the case that the verifier would have accepted, must the extracted state be a QMA witness of high quality?

In the description of the protocol above, we left the final line of the verifier's action somewhat unclear. For example, suppose that the verifier simply samples a uniform $i \leftarrow [N]$, applies $\widetilde{\mathcal{M}_i}$ to $|\widetilde{\psi}\rangle$, and accepts if the measurement accepts. By the negligible soundness of the ZX verifier, it holds that if the verifier accepts with non-negligible probability, then the instance must have been a yes instance. Thus, this argument would be good enough to establish standard soundness of the protocol.

However, accepting with non-negligible probability only guarantees that the witness encoded by the prover would be accepted by the QMA verifier with some non-negligible probability. That is,

the encoded state $|\psi\rangle$ might be a witness of some arbitrarily small inverse polynomial "quality", meaning that the QMA verifier accepts $|\psi\rangle$ with some inverse polynomial probability $\gamma = 1/\mathsf{poly}$. Thus, a natural question is whether it is possible to guarantee extraction of *high* quality witnesses, namely states $|\psi\rangle$ that cause the QMA verifier to accept with probability close to 1.

To do so, the verifier will need to implement a procedure that essentially distinguishes between witnesses of varying quality. Fortunately, [Zha20] derived a method for approximating the probability that a given state is accepted by any *mixture of projective measurements*, which we can take to be the set $\{\widetilde{\mathcal{M}}_i\}_i$. Given any desired cutoff $1-\gamma$ for inverse polynomial γ, the verifier can run [Zha20]'s "approximate threshold implementation" (ATI) procedure, and only accept if ATI accepts. In this case, the state has been (approximately) projected into a space of states that cause the QMA verifier to accept with probability at least $1-\gamma$.

This allows us to establish a strong extraction property: Our extractor will succeed with probability negligibly close to the verifier (meaning we have no knowledge error), and in the case of success, can output a QMA witness with arbitrarily high quality. This improves upon the prior extraction result of [CVZ20], whose protocol had some non-negligible knowledge error and lower witness quality (in addition to being in the secret parameters model).

Zero-knowledge. Our proof of zero-knowledge is somewhat involved, and motivates our use of the evasive composability heuristic on the obfuscator, discussed earlier in Section 1.

Consider an encoded witness $|\widetilde{\psi}\rangle$ along with its obfuscated codespace membership tester $\widetilde{\mathcal{V}}$ and set of obfuscated ZX measurements $\{\widetilde{\mathcal{M}}_i\}_i$. Roughly, our goal will be to replace $|\widetilde{\psi}\rangle$ with an encoded zero state $|\widetilde{0}\rangle$, which clearly contains no information about the witness.

Encouragingly, [BBV24] has established that we can do this as long as the *only* side information is the obfuscated codespace membership tester $\widetilde{\mathcal{V}}$. While technically they show this when the obfuscation is modeled as a black-box, it is easy to see that the only property they use for this is "subspace-hiding" [Zha19], which is implied by indistinguishability obfuscation.

We take this one step further. In Section 5, we show that for any state $|\psi\rangle$ and ZX measurement $\mathcal{M}$ such that $\mathcal{M}$ accepts $|\psi\rangle$ with probability $1-\mathsf{negl}(\lambda)$, it holds that

$$\left(|\widetilde{\psi}\rangle, \widetilde{\mathcal{M}}\right) \approx \left(|\widetilde{0}\rangle, \widetilde{\mathcal{V}}\right),$$

where we still only make use of indistinguishability obfuscation (iO).

Unfortunately, this claim is still not enough to argue zero-knowledge due to the presence of *many* obfuscated ZX measurements. In fact, we run into trouble when trying to argue about an ensemble of the form

$$\left(|\widetilde{\psi}\rangle, \widetilde{\mathcal{M}}_0, \widetilde{\mathcal{M}}_1\right),$$

where $\widetilde{\mathcal{M}}_0$ is an obfuscated ZX measurement with respect to bases θ, $\widetilde{\mathcal{M}}_1$ is an obfuscated ZX measurement with respect to bases θ', and $\theta \neq \theta'$. The reason is that a crucial step in the iO-based proof *decomposes* the state $|\widetilde{\psi}\rangle$ in the θ-basis, and argues separately about each component. However, while $|\widetilde{\psi}\rangle$ itself may be accepted by $\widetilde{\mathcal{M}}_1$, it's components in bases θ may *not* be, meaning that $|\widetilde{\psi}\rangle$ and $|\widetilde{\psi}\rangle$ measured in the θ-basis are no longer indistinguishable in the presence of $\widetilde{\mathcal{M}}_1$!

Now, if we model the obfuscations as *oracles*, then it is possible to "paste" all these arguments together with respect to the separate $\widetilde{\mathcal{M}}_i$ and prove that they are all indistinguishable from $\widetilde{\mathcal{V}}$ in one fell swoop. Indeed, we consider the difficulty above to merely be a difficulty with the particular proof techniques that we (and [BBV24]) utilize, and leave it as a fascinating open question to identify a new proof technique that relies on only indistinguishability obfuscation.

In this work, rather than resorting to the full-fledged oracle model, we extract out a simple game-based property that we need from the obfuscator, which we refer to as the evasive composability heuristic. We consider this a step towards eventually removing heuristics entirely, and relying on only indistinguishability obfuscation or other concrete assumptions. In particular, this highlights a "core" property that we need here (and in other contexts such as obfuscation [BBV24]) from the obfuscated CSA circuits, which current techniques suffice to prove in the oracle model but not in the plain model. We hope that this will lead to a crisper understanding of the current gap in obtaining results such as NIZKs for QMA and obfuscation for quantum programs in the plain model.

2.3 The quantum pseudorandom oracle model

Finally, we discuss our results on the quantum pseudorandom oracle model. We begin by taking a look at our specific NIZK for QMA protocol, attempting to argue security in some idealized model in which the evasive composability heuristic holds.

First consider an attempt to instantiate our protocol with an *ideal obfuscator*. In isolation, it is easy to see that ideal obfuscation satisfies the evasive composability heuristic. However, recall that we also need to prove *correctness* of the obfuscated program. If the prover's obfuscation is modeled completely as a black-box, it is unclear how to do this. Even if the (otherwise plain model) obfuscator makes use of a *random oracle*, our proposed protocol would require NIZKs for oracle-aided NP languages, which are not known.

Drawing inspiration from real-world heuristic use of hash functions as random oracles, [JLLW23] recently defined the *pseudorandom oracle* (PrO) model. In the PrO model, query access to the ideal functionality is indistinguishable from a truly random function, yet there exist "handles" which can be used to uniquely specify a key for the PrO. These "handles" can in turn be used to prove properties regarding the PrO. In their paper, [JLLW23] show that, assuming functional encryption, ideal obfuscation exists in the PrO model, thus showing that heuristic use of obfuscation can often be replaced with the heuristic use of a hash function.

For our purposes, we need to extend the JLLW analysis in two ways: (1) we need a *provably-correct* ideal obfuscator in the PrO, and (2) we need to argue *post-quantum* security, meaning the adversary gets quantum superposition access to the PrO, which we call the quantum PrO, or QPrO. It turns out that to address the first challenge, we must make use of a "cut-and-choose" trick wherein we obfuscate multiple programs that each make use of different PrO keys, and require that the prover reveal a random subset of these keys. This enables the verifier to check that the prover is being (mostly) honest about its key to handle mapping (more details can be found in Section 7.2). Next, we discuss the second challenge below.

Post-Quantum Security of JLLW. Our main contribution in this section is to lift the JLLW results to the post-quantum setting, showing that heuristic use of *post-quantum* obfuscation can often be replaced with the heuristic use of post-quantum hash functions.

At a high level, we carefully follow their construction and analysis in the classical setting in order to show that it is indeed secure in the post-quantum setting, but with some small caveats. The main caveat is that the quantum setting seems to require *subexponential* security from the underlying primitives, as required by the works [BV15, AJ15] that inspired JLLW's construction. JLLW's insight to simulate the obfuscated program for an exponential number of potential inputs is to adaptively reprogram the PrO only on the (polynomial number of) inputs which the adversary queries. Unfortunately, adaptively programming a quantum-accessible random oracle is out of reach of current techniques, at least in the context of simulation security. Instead, we are forced to individually address each input individually like in prior works, which causes an exponential security loss.

The second difference is more minor. Technically, we need to rely on security of the underlying primitives *with access to the QPrO*. The QPrO is implemented using a PRF and a random permutation. To say that the underlying primitives are secure in the QPrO, we need to be able to implement the random permutation. Unfortunately, efficiently statistically simulating a random permutation oracle is an open problem. Instead, we can simulate the QPrO in the plain model using pseudorandom permutations (PRPs). This resolves the issue at the cost of additionally assuming PRPs. Fortunately, post-quantum PRPs are known to be implied by post-quantum PRFs [Zha25].

Roadmap. In Section 4, we define and construct a ZX verifier with strong completeness. We then prove new properties of coset state authentication in Section 5. In Section 6, we construct post-quantum NIZK for NP with knowledge soundness, by putting together results from prior work. Next, we formalize our definition of provably-correct obfuscation, and prove its existence using an obfuscation scheme in Section 7. In Section 8, we establish ideal obfuscation in the quantum pseudorandom oracle model. Finally, we construct and prove our main NIZK for QMA result in Section 9.

3 Preliminaries

We say that two distributions are δ-indistinguishable if no polynomial time adversary can distinguish them with probability better than δ. Frequently, δ will be an arbitrary negligible function, in which case we simply say that the two distributions are computationally distinguishable. In the case where $\delta = 2^{-\lambda^c}$ for some constant c, we say that the distributions are *subexponentially* indistinguishable. If a primitive's security is based on the indistinguishability of two distributions, then we say it is δ-secure if those distributions are δ-indistinguishable. Additionally, for notational purposes, we use use $A[i]$ to denote indexing into a list or string A with i.

3.1 Statistics

We denote the spectral norm, which is the largest singular value of a matrix, by $\|\cdot\|_{\mathsf{spec}}$.

Theorem 3.1 (Rectangular Matrix Bernstein Inequality[Tro15])**.** *Consider a finite sequence $\{\mathbf{Z}_k\}$ of independent, random matrices with dimensions $d_1 \times d_2$. Assume that each matrix satisfies*

$$\mathbb{E}[\mathbf{Z}_k] = \mathbf{0} \quad \textit{and} \quad \|\mathbf{Z}_k\|_{\mathsf{spec}} \leq R$$

Define

$$\sigma^2 := \max\left\{\left\|\sum_k \mathbb{E}[\mathbf{Z}_k\mathbf{Z}_k^*]\right\|_{\mathsf{spec}}, \left\|\sum_k \mathbb{E}[\mathbf{Z}_k^*\mathbf{Z}_k]\right\|_{\mathsf{spec}}\right\}$$

Then for all $t \geq 0$,

$$\Pr\left[\left\|\sum_k \mathbf{Z}_k\right\|_{\mathsf{spec}} \geq t\right] \leq (d_1+d_2)\exp\left(\frac{-t^2/2}{\sigma^2 + Rt/3}\right)$$

We can use this inequality to get a concentration inequality on the sum of independent random vectors.

Lemma 3.2. Let $\{\mathbf{V}_k\}_{k\in[n]}$ be a finite sequence of independent random vectors with dimension d. If $\|\mathbf{V}_k\|_2 \leq R$ for some $R \in \mathbb{R}$ for all k, then

$$\Pr\left[\left\|\sum_k \mathbf{V}_k - \mathbb{E}\left[\sum_k \mathbf{V}_k\right]\right\|_2 \geq t\right] \leq d\exp\left(\frac{-t^2}{2R(2n+t/3)}\right)$$

This also holds for the L-1 norm, since $\|\mathbf{w}\|_2 \leq \|\mathbf{w}\|_1$ for all vectors $\mathbf{w}$. For $t = cn$, the bound becomes $d\exp\big(-c^2k/(4R+2c/3)\big)$.

Proof. Define $\mathbf{Z}_k := \mathbf{V}_k - \mathbb{E}[\mathbf{V}_k]$. This random variable has mean $\mathbf{0}$ and satisfies $\|\mathbf{Z}_k\|_2 \leq 2R$. Since the spectral norm is equivalent the L2 norm on vectors, we may apply the matrix Bernstein inequality to bound $\sum_k \mathbf{Z}_k = \sum_k \mathbf{V}_k - \mathbb{E}[\sum_k \mathbf{V}_k]$ in terms of

$$\sigma^2 = \max\left\{\left\|\sum_k \mathbb{E}[\mathbf{Z}_k\mathbf{Z}_k^*]\right\|_{\mathsf{spec}}, \left\|\sum_k \mathbb{E}[\mathbf{Z}_k^*\mathbf{Z}_k]\right\|_{\mathsf{spec}}\right\}$$

We can bound this by

$$\begin{aligned}\left\|\sum_k \mathbb{E}[\mathbf{Z}_k\mathbf{Z}_k^*]\right\|_{\mathsf{spec}} &\leq \sum_k \|\mathbb{E}[\mathbf{Z}_k\mathbf{Z}_k^*]\|_{\mathsf{spec}} \\ &\leq \sum_k \mathbb{E}[\|\mathbf{Z}_k\mathbf{Z}_k^*]\|_{\mathsf{spec}}] \\ &= \sum_k \mathbb{E}[\|\mathbf{Z}_k\|_2] \\ &\leq 2nR\end{aligned}$$

using a combination of the triangle inequality, Jensen's inequality, and the a-priori bound on $\|\mathbf{V}_k\|_2$. A similar bound applies to $\|\sum_k \mathbb{E}[\mathbf{Z}_k^*\mathbf{Z}_k]\|_{\mathsf{spec}}$, giving us an overall bound on σ^2. □

3.2 Quantum Information

First, we define the notion of a (binary-outcome) ZX measurement.

Definition 3.3 (Binary-outcome ZX measurement)**.** An n-qubit binary-outcome ZX measurement is parameterized by a string $\theta \in \{0,1\}^n$ of basis choices (where each 0 corresponds to standard basis and each 1 corresponds to Hadamard basis), and a function $f : \{0,1\}^n \to \{0,1\}$. It is defined as

$$\{M[\theta, f], \mathcal{I} - M[\theta, f]\}, \quad \text{where} \quad M[\theta, f] := H^\theta \left(\sum_{x : f(x)=1} |x\rangle\langle x| \right) H^\theta.$$

We say that the ZX measurement is efficient if f is computable by a uniform circuit of size polynomial in n. We consider only efficient ZX measurements in this work.

Lemma 3.4 (Gentle measurement [Win99])**.** Let ρ be a quantum state and let $(\Pi, I - \Pi)$ be a projective measurement such that $\mathrm{Tr}(\Pi\rho) \geq 1 - \delta$. Let

$$\rho' = \frac{\Pi\rho\Pi}{\mathrm{Tr}(\Pi\rho)}$$

be the state after applying $(\Pi, I - \Pi)$ to ρ and post-selecting on obtaining the first outcome. Then, $\mathsf{TD}(\rho, \rho') \leq 2\sqrt{\delta}$.

The following two lemmas are taken mostly verbatim from [BBV24].

Lemma 3.5 (Oracle indistinguishability)**.** For each $\lambda \in \mathbb{N}$, let $\mathcal{K}_\lambda$ be a set of keys, and $\{z_k, O_k^0, O_k^1, S_k\}_{k \in \mathcal{K}_\lambda}$ be a set of strings z_k, classical functions O_k^0, O_k^1, and sets S_k. Suppose that the following properties hold.

1. The oracles O_k^0 and O_k^1 are identical on inputs outside of S_k.
2. For any oracle-aided unitary U with $q = q(\lambda)$ queries, there is some $\epsilon = \epsilon(\lambda)$ such that

$$\mathop{\mathbb{E}}_{k \leftarrow \mathcal{K}} \left[\|\Pi[S_k] U^{O_k^0}(z_k)\|^2 \right] \leq \epsilon.$$

Then, for any oracle-aided unitary U with $q(\lambda)$ queries and distinguisher D,

$$\left| \Pr_{k \leftarrow \mathcal{K}} \left[D\left(k, U^{O_k^0}(z_k)\right) = 0 \right] - \Pr_{k \leftarrow \mathcal{K}} \left[D\left(k, U^{O_k^1}(z_k)\right) = 0 \right] \right| \leq 4q\sqrt{\epsilon}.$$

Lemma 3.6 (State decomposition)**.** Let $\mathcal{K}$ be a set of keys, N an integer, and $\{|\psi_k\rangle, \{\Pi_{k,i}\}_{i \in [N]}\}_{k \in \mathcal{K}}$ be a set of states $|\psi_k\rangle$ and projective submeasurements $\{\Pi_{k,i}\}_{i \in [N]}$ such that $|\psi_k\rangle \in \mathsf{Im}(\sum_i \Pi_{k,i})$ for each k. Then for any binary-outcome projector D, it holds that

$$\mathop{\mathbb{E}}_{k \leftarrow \mathcal{K}} \left[\| D |\psi_k\rangle \|^2 \right] - \sum_i \mathop{\mathbb{E}}_{k \leftarrow \mathcal{K}} \| D\Pi_{k,i} |\psi_k\rangle \|^2 \leq N \cdot \sqrt{\sum_{i \neq j} \mathop{\mathbb{E}}_{k \leftarrow \mathcal{K}} \| \Pi_{k,j} D \Pi_{k,i} |\psi_k\rangle \|^2}.$$

3.3 Estimating Quantum Acceptance Probabilities

[Zha20] gave a method of approximating the probability that a state is accepted by a POVM $(\mathcal{P} = \sum_i p_i P_i, \mathcal{Q} = \sum_i p_i (I - P_i))$ which is a mixture of binary-outcome projective measurements $\{P_i, I - P_i\}$. Crucially, the method is almost-projective. In other words, if run twice, it will almost

certainly give the same result both times. Later, [ALL+21] observed that the technique can be applied to test if a state's acceptance probability is greater than some threshold.

Although the technique is quite general, we only need a few very specific properties that arise from plugging in specific parameters to the general technique. We refer the reader to [Zha20] for a fully detailed description of the general technique.

Lemma 3.7. Let $(\mathcal{P} = \sum_i p_i P_i, \mathcal{Q} = \sum_i p_i(I - P_i))$ be a mixture of projective measurements such that each is implementable in time $\mathsf{poly}(\lambda)$ and one can sample from the distribution defined by $\Pr[i] = p_i$ in time $\mathsf{poly}(\lambda)$. For any $\gamma(\lambda) \in [1/\mathsf{poly}(\lambda), 1 - 1/\mathsf{poly}(\lambda)]$, there exists an algorithm ATI_γ outputting Accept or Reject such that the following hold.

- **Efficient.** The expected running time of ATI_γ is $\mathsf{poly}(\lambda)$.
- **Approximately Projective.** ATI_γ is approximately projective. In other words, for all states ρ,

$$\Pr\left[b_1 = b_2 : \begin{array}{c}(b_1, \rho') \leftarrow \mathsf{ATI}_\gamma(\rho) \\ b_2 \leftarrow \mathsf{ATI}_\gamma(\rho')\end{array}\right] = 1 - \mathsf{negl}(\lambda).$$

- **Soundness.** For every state ρ

$$\Pr\left[\begin{array}{c}b = \mathsf{Accept} \wedge \\ \mathrm{Tr}[\mathcal{P}\rho'] \leq 1 - \gamma(\lambda)\end{array} : (\rho', b) \leftarrow \mathsf{ATI}_\gamma(\rho)\right] = \mathsf{negl}(\lambda).$$

- For every state $|\psi\rangle$ such that $\mathrm{Tr}[\mathcal{P}|\psi\rangle\langle\psi|] \geq 1 - \mathsf{negl}(\lambda)$,

$$\Pr[\mathsf{Accept} \leftarrow \mathsf{ATI}_\gamma(|\psi\rangle)] \geq 1 - \mathsf{negl}(\lambda).$$

- If $\mathrm{Tr}[\mathcal{P}\rho] < 1 - \gamma(\lambda)$ for every state ρ, then for any state ρ,

$$\Pr[\mathsf{Accept} \leftarrow \mathsf{ATI}_\gamma(\rho)] = \mathsf{negl}(\lambda).$$

Proof Sketch. This follows by plugging in explicit parameters to Corollary 1 in [ALL+21] and Theorem 2 in [Zha20]. Specifically, set the approximation precision $\epsilon = \gamma/2$, set the approximation accuracy $\delta = 2^\lambda$, and set the threshold to 1.

Their algorithm runs in time $\mathsf{poly}(2/\gamma(\lambda), \log(2^\lambda)) = \mathsf{poly}(\lambda)$ and is δ-approximately projective. It δ-approximates the threshold projective implementation $(\Pi_{\geq 1-\gamma/2}, I - \Pi_{\geq 1-\gamma/2})$ of $(\mathcal{P}, \mathcal{Q})$. Specifically, if $|\psi\rangle$ is in the image of $\Pi_{\geq 1-\gamma/2}$, then ATI_γ accepts $|\psi\rangle$ with probability $1 - \delta$ and otherwise it rejects it with probability $1 - \delta$. Here, $\Pi_{\geq 1-\gamma/2}$ projects onto eigenstates of the projective implementation of $(\mathcal{P}, \mathcal{Q})$ with eigenvalues $\geq 1 - \gamma/2$. Any such eigenstate $|\psi\rangle$ with eigenvalue ζ has $\mathrm{Tr}[\mathcal{P}|\psi\rangle] = \zeta$.

For any state ρ where $\Pr[\mathsf{Accept} \leftarrow \mathsf{ATI}_\gamma(\rho)] = \mathsf{negl}(\lambda)$, it is clearly the case that

$$\Pr\left[\begin{array}{c}b = \mathsf{Accept} \wedge \\ \mathrm{Tr}[\mathcal{P}\rho'] \leq 1 - \gamma\end{array} : (\rho', b) \leftarrow \mathsf{ATI}_\gamma(\rho)\right] = \mathsf{negl}(\lambda).$$

On the other hand, if ATI_γ accepts ρ with noticeable probability, then by approximate projectivity the probability that ATI_γ accepts ρ but then rejects the residual state ρ' is negligible. Suppose we are in the case where $\Pr[\mathsf{Accept} \leftarrow \mathsf{ATI}_\gamma(\rho')] = 1 - \mathsf{negl}(\lambda)$. Since ATI_γ $(2^{-\lambda})$-approximates

$(\Pi_{\geq 1-\gamma/2}, I - \Pi_{\geq 1-\gamma/2})$, ρ' must have negligible projection onto the eigenspace of the projective implementation of $\mathcal{P}$ with eigenvalues $< 1-\gamma/2$. In other words, $\mathrm{Tr}[\mathcal{P}\rho'] \geq 1 - \gamma/2 - \mathsf{negl} > 1 - \gamma$.

If $\mathrm{Tr}[\mathcal{P}\left|\psi\right\rangle] \geq 1 - \mathsf{negl}(\lambda)$, then $\left|\psi\right\rangle$ must have negligible projection onto the eigenspace of the projective implementation of $(\mathcal{P}, \mathcal{Q})$ with eigenvalues $\leq 1 - 1/p$ for any $p = \mathsf{poly}(\lambda)$. Thus, $\left|\psi\right\rangle$ is accepted by ATI_γ with probability $1 - \delta = 1 - 2^{-\lambda}$.

On the other hand, if $\mathrm{Tr}[\mathcal{P}\rho] < 1 - \gamma$ for every ρ, then the maximum eigenvalue of projective implementation of $(\mathcal{P}, \mathcal{Q})$ is $< \gamma$. Therefore every state ρ is in the image of $1 - \Pi_{\geq 1-\gamma/2}$ and thus ATI_γ rejects ρ with probability $1 - \delta = 1 - 2^{-\lambda}$. □

3.4 Quantum Complexity

Let $\mathcal{B}$ denote the Hilbert space of a single qubit, and let $D(\mathcal{B}^{\otimes p})$ be the set of p-qubit density matrices, i.e. $\rho \in D(\mathcal{B}^{\otimes p})$ iff ρ is PSD and has trace 1.

Definition 3.8 (QMA Promise Problem)**.** A promise problem $(\mathcal{L}_{yes}, \mathcal{L}_{no}) \in \mathsf{QMA}$ if there exists a quantum polynomial-size family of circuits $Q = \{Q_n\}_{n\in\mathbb{N}}$, a polynomial $p(\cdot)$, and an $\epsilon(\cdot)$ such that $2^{-\Omega(\cdot)} \leq \epsilon(\cdot) \leq \frac{1}{3}$, such that

- For all $x \in \mathcal{L}_{yes}$, there exists $\left|\psi\right\rangle \in \mathcal{B}^{\otimes p(|x|)}$ such that $\Pr\left[Q_{|x|}(x, \left|\psi\right\rangle) = 1\right] \geq 1 - \epsilon(|x|)$.
- For all $x \in \mathcal{L}_{no}$ and all $\left|\psi\right\rangle \in \mathcal{B}^{\otimes p(|x|)}$, $\Pr\left[Q_{|x|}(x, \left|\psi\right\rangle) = 1\right] \leq \epsilon(|x|)$.

Given a quantum polynomial-size family of circuits $Q = \{Q_n\}_{n\in\mathbb{N}}$ where each Q_n takes as input a string $x \in \{0,1\}^n$ and a state $\left|\psi\right\rangle$ on $p(n)$ qubits, and a function $\gamma : \mathbb{N} \to [0,1]$, we define the sets

$$R_{Q,\gamma} \coloneqq \bigcup_{n\in\mathbb{N}} \left\{(x,\rho) \in \{0,1\}^n \otimes D(\mathcal{B}^{\otimes p(|x|)}) \mid \Pr[Q_n(x,\rho) = 1] \geq \gamma(n)\right\},$$

and

$$N_{Q,\gamma} \coloneqq \bigcup_{n\in\mathbb{N}} \left\{x \in \{0,1\}^n \mid \forall \rho \in D(\mathcal{B}^{\otimes p(|x|)}), \Pr[Q_n(x,\rho) = 1] < \gamma(n)\right\}.$$

Using this notation, we can define the notion of a QMA relation.

Definition 3.9 (QMA Relation)**.** A QMA language $(\mathcal{L}_{yes}, \mathcal{L}_{no})$ is specified by a QMA Relation (Q, α, β), where $Q = \{Q_n\}_{n\in\mathbb{N}}$ is a quantum polynomial-size family of circuits, and $\alpha, \beta : \mathbb{N} \to [0,1]$ are functions, if

$$\mathcal{L}_{yes} \subseteq \bigcup_{n\in\mathbb{N}} \left\{x \in \{0,1\}^n \mid \exists \rho \in D(\mathcal{B}^{\otimes p(|x|)}) \text{ s.t. } (x,\rho) \in R_{Q,\alpha}\right\},$$

and $\mathcal{L}_{no} \subseteq N_{Q,\beta}$.

Next, we discuss the Local Hamiltonian problem.

Definition 3.10 (2-local ZX-Hamiltonian problem [BL08, CM16, MNS16])**.** The 2-local ZX-Hamiltonian promise problem $(\mathcal{L}_{yes}, \mathcal{L}_{no})$, with functions a, b where $b(n) > a(n)$ and gap $b(n) - a(n) > \mathsf{poly}(n)^{-1}$

for all $n \in \mathbb{N}$ is defined as follows. An instance is a Hermitian operator on some number n of qubits, taking the following form:

$$H = \sum_{\substack{i<j \\ S\in\{Z,X\}}} p_{i,j} P_{i,j,S}$$

where probability $p_{i,j} \in [0,1]$ with $\sum_{i<j} 2p_{i,j} = 1$, and projector $P_{i,j,S} = \frac{I+(-1)^{\beta_{i,j}} S_i S_j}{2}$ for $\beta_{i,j} \in \{0,1\}$.

- $H \in \mathcal{L}_{yes}$ if the smallest eigenvalue of H is at most $a(n)$.
- $H \in \mathcal{L}_{no}$ if the smallest eigenvalue of H is at least $b(n)$.

Theorem 3.11 (2-local ZX-Hamiltonian is QMA-complete [BL08])**.** *The 2-local ZX-Hamiltonian problem with functions a, b (Theorem 3.10) is QMA-complete if $b(n) - a(n) > \mathsf{poly}(n)^{-1}$.*

Definition 3.12 (2-local ZX-Hamiltonian Verifier [MNS16])**.** Let $(\mathcal{L}_{yes}, \mathcal{L}_{no})$ be a 2-local ZX-Hamiltonian promise problem. There exists functions p_{yes}, p_{no} where $p_{yes}, p_{no} : \mathbb{N} \to [0,1]$ and $p_{yes}(n) - p_{no}(n) \geq \mathsf{poly}(n)^{-1}$ for all n such that the following construction has the subsequent properties:

Construction.

- $(i, j, S) \leftarrow \mathsf{Samp}(H; r)$: The classical polynomial-size circuit Samp on input instance H outputs indices i, j and choice of basis $S \in \{Z, X\}$ with probability $p_{i,j}$ using uniform randomness r.
- $\mathsf{ZXVer}(H, |\psi\rangle) \in \{0,1\}$: The quantum polynomial-size circuit ZXVer on input instance H and witness $|\psi\rangle$,
  1. Sample projector indices $(i, j, S) \leftarrow \mathsf{Samp}(H)$.
  2. Measure the ith and jth qubits of $|\psi\rangle$ with the projector $\{M_0 = \frac{I-S}{2}, M_1 = \frac{I+S}{2}\}$ to get b_i and b_j.
  3. Output $b_i \oplus b_j \oplus \beta_{i,j}$.

Properties.

- **Correctness.** For every $H \in \mathcal{L}_{yes}$ with witness $|\psi\rangle$,

$$\Pr[\mathsf{ZXVer}(H, |\psi\rangle) = 1] \geq p_{yes}(n)$$

- **Soundness.** For every $H \in \mathcal{L}_{no}$, and for every ρ,

$$\Pr[\mathsf{ZXVer}(H, \rho) = 1] \leq p_{no}(n)$$

3.5 Encryption

Public-key encryption. We first define standard (post-quantum) public-key encryption.

Definition 3.13 (Post-Quantum Public-Key Encryption)**.** $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is a post-quantum public-key encryption scheme if it has the following syntax and properties.

Syntax.

- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$: The polynomial-time algorithm Gen on input security parameter 1^λ outputs a public key pk and a secret key sk.
- $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$: The polynomial-time algorithm Enc on input a public key pk, message m and randomness $r \in \{0,1\}^{r(\lambda)}$ outputs a ciphertext ct.
- $m \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$: The polynomial-time algorithm Dec on input a secret key sk and a ciphertext ct outputs a message m.

Properties.

- **Perfect Correctness**: For every $\lambda \in \mathbb{N}^+$ and every m, r,

$$\Pr_{(\mathsf{pk},\mathsf{sk})\leftarrow\mathsf{Gen}(1^\lambda)}[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m; r)) = m] = 1.$$

- **Indistinguishability under Chosen-Plaintext (IND-CPA) Secure**: There exists a negligible function $\mathsf{negl}(\cdot)$ such that for every polynomial-size quantum circuit $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ and every sufficiently large $\lambda \in \mathbb{N}^+$

$$\left| \Pr_{\substack{(\mathsf{pk},\mathsf{sk})\leftarrow\mathsf{Gen}(1^\lambda)\\(m_0,m_1,\zeta)\leftarrow\mathcal{A}_0(1^\lambda,\mathsf{pk})\\ \mathsf{ct}\leftarrow\mathsf{Enc}(\mathsf{pk},m_0)}} [\mathcal{A}_1(1^\lambda, \mathsf{ct}, \zeta) = 1] - \Pr_{\substack{(\mathsf{pk},\mathsf{sk})\leftarrow\mathsf{Gen}(1^\lambda)\\(m_0,m_1,\zeta)\leftarrow\mathcal{A}_0(1^\lambda,\mathsf{pk})\\ \mathsf{ct}\leftarrow\mathsf{Enc}(\mathsf{pk},m_1)}} [\mathcal{A}_1(1^\lambda, \mathsf{ct}, \zeta) = 1] \right| \leq \mathsf{negl}(\lambda).$$

Theorem 3.14.

Functional encryption. Next, we define a flavor of (1-key) functional encryption described in [JLLW23], with the additional requirement that it is post-quantum secure.

Definition 3.15 (Post-quantum 1-key FE)**.** A post-quantum (public-key) 1-key functional encryption scheme (for circuits) has the following syntax and properties.

Syntax.

- $(\mathsf{pk}, \mathsf{sk}_f) \leftarrow \mathsf{Gen}(1^\lambda, f)$: The Gen algorithm takes a circuit $f : \{0,1\}^n \to \{0,1\}^*$ and outputs a public (encryption) key pk and a secret (decryption) key for f.
- $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, z)$: The Enc algorithm takes a public key and a plaintext $z \in \{0,1\}^n$ and outputs a ciphertext ct.
- $f(z) \leftarrow \mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct})$: The Dec algorithm takes a secret key for f and a ciphtertext and outputs a string $f(z)$.

Properties.

- **Perfect correctness.** For all $\lambda \in \mathbb{N}$, circuit $f : \{0,1\}^n \to \{0,1\}^*$, and input $z \in \{0,1\}^n$, it holds that

$$\Pr\left[\mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}) = f(z) : \begin{array}{r}(\mathsf{pk}, \mathsf{sk}_f) \leftarrow \mathsf{Gen}(1^\lambda, f)\\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, z)\end{array}\right] = 1.$$

- **Subquadratic-sublinear efficiency.** Enc runs in time $(n^{2-2\epsilon} + m^{1-\epsilon})\mathsf{poly}(\lambda)$ for some constant $\epsilon > 0$, where $n = |z|$ is the input length of f and $m = |f|$ is the circuit size of f.
- **Post-quantum adaptive security.** For any $b \in \{0,1\}$ and adversary $\mathcal{A}$, let $\mathsf{Exp}^{\mathcal{A},b}_{\text{1-key}}$ be the following experiment.
  - $\mathcal{A}(1^\lambda)$ outputs a circuit $f : \{0,1\}^n \to \{0,1\}^*$. Run $(\mathsf{pk}, \mathsf{sk}_f) \leftarrow \mathsf{Gen}(1^\lambda, f)$, and send $(\mathsf{pk}, \mathsf{sk}_f)$ to $\mathcal{A}$.
  - $\mathcal{A}$ chooses two inputs $z_0, z_1 \in \{0,1\}^n$. Run $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, z_b)$ and send ct to $\mathcal{A}$.
  - $\mathcal{A}$ outputs a bit $b' \in \{0,1\}$. The outcome of the experiment is b' if $f(z_0) = f(z_1)$, and is otherwise set to 0.

  There exists an $\epsilon > 0$ such that for any QPT adversary $\mathcal{A}$, it holds that

$$\left| \Pr\left[\mathsf{Exp}^{\mathcal{A},0}_{\text{1-key}} = 0\right] - \Pr\left[\mathsf{Exp}^{\mathcal{A},1}_{\text{1-key}} = 0\right] \right| \leq 2^{-\lambda^\epsilon}.$$

3.6 NIZK for NP

Definition 3.16 (Post-Quantum NIZK for NP in the CRS Model)**.** Let NP relation $\mathcal{R}$ with corresponding language $\mathcal{L}$ be given such that they can be indexed by a security parameter $\lambda \in \mathbb{N}$.

$\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ is a post-quantum non-interactive zero-knowledge argument for NP in the URS model if it has the following syntax and properties.

Syntax. The input 1^λ is left out when it is clear from context.

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

Properties.

- **Uniform Random String.** $\mathsf{Setup}(1^\lambda)$ outputs a uniformly random string crs.
- **Perfect Completeness.** For every $\lambda \in \mathbb{N}$ and every $(x, w) \in \mathcal{R}_\lambda$,

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

- **Adaptive Statistical (Computational) Soundness.** There exists a negligible function $\mathsf{negl}(\cdot)$ such that for every unbounded (polynomial-size) quantum circuit $\mathcal{A}$ and every sufficiently large $\lambda \in \mathbb{N}$,

$$\Pr_{\substack{\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda) \\ (x,\pi) \leftarrow \mathcal{A}(\mathsf{crs})}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge x \notin \mathcal{L}_\lambda] \leq \mathsf{negl}(\lambda).$$

- **Non-Adaptive Computational T-Soundness.** There exists a negligible function $\mathsf{negl}(\cdot)$ such that for every $\mathsf{poly}(T)$-size quantum circuit $\mathcal{A}$ and every sufficiently large $\lambda \in \mathbb{N}$ and $x \notin \mathcal{L}_\lambda$,

$$\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\ \pi\leftarrow\mathcal{A}(\mathsf{crs})}}[\mathsf{V}(\mathsf{crs},x,\pi)=1] \leq \mathsf{negl}(T(\lambda)).$$

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

$$\left|\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\ (x,w,\zeta)\leftarrow\mathcal{D}_0(\mathsf{crs})\\ \pi\leftarrow\mathsf{P}(\mathsf{crs},x,w)}}[\mathcal{D}_1(\mathsf{crs},x,\pi,\zeta)=1\wedge x\in\mathcal{L}_\lambda] - \Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Sim}_0(1^\lambda)\\ (x,w,\zeta)\leftarrow\mathcal{D}_0(\mathsf{crs})\\ \pi\leftarrow\mathsf{Sim}_1(\mathsf{crs},x)}}[\mathcal{D}_1(\mathsf{crs},x,\pi,\zeta)=1\wedge x\in\mathcal{L}_\lambda]\right| \leq \mathsf{negl}(\lambda).$$

- **Non-Adaptive Statistical Zero-Knowledge.** There exists a probabilistic polynomial-size circuit Sim and a negligible function $\mathsf{negl}(\cdot)$ such that for every unbounded quantum circuit $\mathcal{D}$, and every sufficiently large $\lambda \in \mathbb{N}$ and every $(x,w)\in\mathcal{R}_\lambda$,

$$\left|\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\ \pi\leftarrow\mathsf{P}(\mathsf{crs},x,w)}}[\mathcal{D}(\mathsf{crs},x,\pi)=1] - \Pr_{(\mathsf{crs},\pi)\leftarrow\mathsf{Sim}(x)}[\mathcal{D}(\mathsf{crs},x,\pi)=1]\right| \leq \mathsf{negl}(\lambda).$$

Theorem 3.17 (Post-Quantum NIZK proof for NP with CRS [PS19])**.** *Assuming the polynomial quantum hardness of LWE, there exists an adaptively statistically sound, adaptively computationally zero-knowledge non-interactive protocol for* NP *having a common reference string (Theorem 3.16).*

Theorem 3.18 (Post-Quantum NISZK argument for NP with URS [PS19])**.** *Assuming the polynomial quantum hardness of LWE, there exists a non-adaptively computationally sound, non-adaptively statistically zero-knowledge non-interactive protocol for* NP *having a uniform random string (Theorem 3.16).*

Corollary 3.19 (Post-Quantum NISZK sub-exp argument for NP with URS)**.** Assuming the sub-exponential quantum hardness of LWE, there exists a non-adaptively computationally sound, non-adaptively statistically zero-knowledge non-interactive protocol for NP with sub-exponential computational soundness error having a uniform random string (Theorem 3.16).

Proof. This follows from Theorem 3.18. □

3.7 Obfuscation

Definition 3.20 (Indistinguishability obfuscation)**.** An indistinguishability obfuscator has the following syntax.

- $\mathsf{Obf}(1^\lambda, C) \to \widetilde{C}$. The obfuscation algorithm takes as input the security parameter and a circuit C, and outputs an obfuscated circuit $\widetilde{C}$.

- $\mathsf{Eval}(\widetilde{C}, x) \to y$. The evaluation algorithm takes as input an obfuscated circuit $\widetilde{C}$ and an input x and outputs y.

It should satisfy the following properties.

- **Functionality-preservation**. For any circuit C, $\widetilde{C} \in \mathsf{Obf}(1^\lambda, C)$, and x, $\mathsf{Eval}(\widetilde{C}, x) = C(x)$.
- **(Sub-exponential) security**. There exists a constant $\epsilon > 0$ such that for any QPT adversary $\mathcal{A}$ and C_0, C_1 such that $C_0 \equiv C_1$,

$$\left| \Pr\left[\mathcal{A}\left(\mathsf{Obf}(1^\lambda, C_0)\right) = 1\right] - \Pr\left[\mathcal{A}\left(\mathsf{Obf}(1^\lambda, C_1)\right) = 1\right] \right| \leq 2^{-\lambda^\epsilon}.$$

Before stating the next imported theorem, we introduce the following notation. For any set S, define $C[S]$ to the membership-checking circuit that, on input a vector $v \in \mathbb{F}_2^n$, outputs 1 if $v \in S$, and outputs 0 otherwise.

Theorem 3.21 (Subspace-hiding obfuscation [Zha21])**.** *Let* $(\mathsf{Obf}, \mathsf{Eval})$ *be a sub-exponentially secure indistinguishability obfuscator, and suppose that sub-exponentially secure injective one-way functions exist. Let* $S \subset \mathbb{F}_2^n$ *be a subspace of* $\mathbb{F}_2^n$ *of dimension* d_0*, let* d_1 *be such that* $d_0 < d_1 < n$*, and define* $\lambda = n - d_1$*. There exists a polynomial* $p(\cdot)$ *such that for any QPT adversary* $\mathcal{A}$*,*

$$\left| \Pr\left[\mathcal{A}\left(\mathsf{Obf}(1^{p(\lambda)}, C[S])\right) = 1\right] - \left[\mathcal{A}\left(\mathsf{Obf}(1^{p(\lambda)}, C[T])\right) = 1 : T \leftarrow \mathsf{Sup}_{d_1}(S)\right] \right| = 2^{-\Omega(\lambda)},$$

where $\mathsf{Sup}_{d_1}(S)$ *is the set of superspaces of* S *of dimension* d_1*.*

We remark that [Zha21] proves the slightly different statement that, assuming polynomially-secure iO and injective one-way functions, the above advantage is at most negligible in some parameter λ, as long as $n - d_1$ is linear in λ. It is straightforward to port their proof to our setting of sub-exponential security.

Finally, we note that the following notion of point-function obfuscation follows as a corollary.

Theorem 3.22 (Point-function obfuscation)**.** *Let* $(\mathsf{Obf}, \mathsf{Eval})$ *be a sub-exponentially secure indistinguishability obfuscator, and suppose that sub-exponentially secure injective one-way functions exist. There exists a polynomial* $p(\cdot)$ *such that for any QPT adversary* $\mathcal{A}$*,*

$$\left| \Pr\left[\mathcal{A}\left(\mathsf{Obf}(1^{p(\lambda)}, C[\{\}])\right) = 1\right] - \left[\mathcal{A}\left(\mathsf{Obf}(1^{p(\lambda)}, C[\{x\}])\right) = 1 : x \leftarrow \{0,1\}^\lambda\right] \right| = 2^{-\Omega(\lambda)}.$$

3.8 The (Q)PrO model

First, we define the quantum-accessible pseudorandom oracle (QPrO) model, which extends the psueudorandom oracle model introduced in [JLLW23] to allow for quantum queries.

Definition 3.23 (QPrO Model)**.** Let $F = \{f_k\}_k$ be a pseudorandom function. The quantum-accessible pseudorandom oracle model for F consists of the following interface, which internally

use a uniformly random permutation $\pi : \{0,1\}^\lambda \to \{0,1\}^\lambda$, and may be queried in quantum superposition.

- $\mathsf{QPrO}(\mathsf{Gen}, k) \to \pi(k)$
- $\mathsf{QPrO}(\mathsf{Eval}, h, x) \to f_{\pi^{-1}(h)}(x)$

The $p(\lambda)$-QPrO model allows the querier to access independent $p(\lambda)-$QPrO oracles for some polynomial p, i.e., oracle access to QPrO is shorthand for allowing query access to $p(\lambda)$-independent QPrO instantiations $\mathsf{QPrO}_0, \mathsf{QPrO}_1, \ldots, \mathsf{QPrO}_{p(\lambda)-1}$.

Next, we present the construction of obfuscation in the pseudorandom oracle model due to [JLLW23]. While [JLLW23] show that this scheme satisfies ideal obfuscation in the PrO, we will show in Section 8 that this scheme in fact satisfies post-quantum ideal obfuscation in the QPrO (as long as the building blocks are post-quantum). Before presenting the obfuscator, we define ideal obfuscation (in an oracle model). We use $C^\bullet$ to denote an oracle-aided circuit.

Definition 3.24 (Ideal obfuscation)**.** An obfuscation scheme in an idealized model with oracle $\mathcal{O}$ is an efficient algorithm $\mathsf{Obf}^{\mathcal{O}}(1^\lambda, C)$ that, given a circuit C as input, outputs an oracle circuit $\widehat{C}^\bullet$. The scheme must be **correct**, i.e. for all $\lambda \in \mathbb{N}$, circuit $C : \{0,1\}^D \to \{0,1\}^*$, and input $x \in \{0,1\}^D$, it holds that

$$\Pr\left[\widehat{C}^{\mathcal{O}}(x) = C(x) : \widehat{C}^\bullet \leftarrow \mathsf{Obf}^{\mathcal{O}}(1^\lambda, C)\right] = 1.$$

It satisfies (post-quantum) **ideal obfuscation relative to an oracle** $\mathcal{R}$ if there exists a QPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ such that for all QPT adversaries $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$,

$$\left|\Pr\left[\mathcal{A}_2^{\mathcal{O}}(\widehat{C}^\bullet) = 1 : \begin{array}{c} C \leftarrow \mathcal{A}_1^{\mathcal{O}(\lambda),\mathcal{R}} \\ \widehat{C}^\bullet \leftarrow \mathsf{Obf}^{\mathcal{O},\mathcal{R}}(1^\lambda, C) \end{array}\right] - \Pr\left[\mathcal{A}_2^{\mathcal{S}_3^C,\mathcal{R}}(\widetilde{C}^\bullet) = 1 : \begin{array}{c} C \leftarrow \mathcal{A}_1^{\mathcal{S}_1,\mathcal{R}}(1^\lambda) \\ \widetilde{C}^\bullet \leftarrow \mathcal{S}_2^C(1^\lambda, 1^D, 1^S) \end{array}\right]\right| = \mathsf{negl}(\lambda),$$

where $D = |x|$ in the input length of C and $S = |C|$ is the circuit size of C.

An important difference from [JLLW23]'s definition of ideal obfuscation is the addition of a relativizing oracle $\mathcal{R}$. The additional of this oracle is crucial for composability with other primitives that might exist in the PrO. In the plain model, ideal obfuscation is naturally composeable via a hybrid argument. However, when constructed in an oracle model such as the PrO, the simulator for an individual obfuscation might seize control of the global oracle. Unfortunately, this can interfere with the simulators for the other instances, which *also* need control over the global oracle.

By introducing the relativizing oracle $\mathcal{R}$, which the simulator is not allowed to claim control of, we ensure that the other simulators can also operate. As a simple example, one can imagine a world in which multiple hash functions (or a single hash function with different salts) define distinct PrOs.

Construction in the PrO. Now, we describe the construction of obfuscation in the PrO due to [JLLW23]. We first specify the ingredients:

- D the input length of the circuit C to be obfuscated.
- S the circuit size of C
- L the block length (determined as in [JLLW23]).
- B the number of blocks (determined as in [JLLW23]).

- $H : \{0,1\}^\lambda \times \{0,1\}^D \to \{0,1\}^L$ the (quantum-query secure) PRF used by the QPrO model.
- $G_{sr} : \{0,1\}^\lambda \to \{0,1\}^{4\lambda}$ the (post-quantum) PRG for encryption randomness.
- $G_v : \{0,1\}^\lambda \to \{0,1\}^L$ the (post-quantum) PRG for decryption result simulation.
- $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ a (post-quantum) 1-key FE scheme (Theorem 3.15) such that Enc uses λ-bit uniform randomness.

Construction 3.25 (JLLW Obfuscator)**.** *The JLLW obfuscator is defined as follow, where* QPrO *is the pseudorandom oracle model defined in Theorem 3.23, using PRF H.*

$\mathsf{JLLWObf}^{\mathsf{QPrO}}(1^\lambda, C)$*:*

- *Set up $(D+1)$ FE instances:*

$$(\mathsf{pk}_D, \mathsf{sk}_D) \leftarrow \mathsf{Gen}(1^\lambda, \mathsf{Eval}),$$
$$(\mathsf{pk}_d, \mathsf{sk}_d) \leftarrow \mathsf{Gen}(1^\lambda, \mathsf{Expand}_d[\mathsf{pk}_{d+1}]) \quad \text{for } d = D-1, \ldots, 0,$$

  where Eval *and* Expand_d *are defined below.*

- *Sample keys of H and obtain their handles:*

$$k_{i,j} \leftarrow \{0,1\}^\lambda, \quad h_{i,j} \leftarrow \mathsf{QPrO}(\mathsf{Gen}, k_{i,j}) \quad \text{for } 0 \le i < D, 1 \le j \le B.$$

- *Sample PRG seed and encryption randomness for the root ciphertext, set its flag and information, and compute* ct_ϵ*:*

$$s_\epsilon \leftarrow \{0,1\}^\lambda, \quad r_\epsilon \leftarrow \{0,1\}^\lambda$$
$$\mathsf{flag}_\epsilon := \mathsf{normal}, \quad \mathsf{info}_\epsilon := (C, \{k_{i,j}\}_{0 \le i < D, 1 \le j \le B}, s_\epsilon)$$
$$\mathsf{ct}_\epsilon := \mathsf{Enc}(\mathsf{pk}_0, \mathsf{flag}_\epsilon, \epsilon, \mathsf{info}_\epsilon; r_\epsilon).$$

- *Output the circuit* $\widehat{C}^\bullet[\mathsf{ct}_\epsilon, \{\mathsf{sk}_d\}_{0 \le d \le D}, \{h_{i,j}\}_{0 \le i < D, 1 \le j \le B}]$*, which operates as follows on input x:*
  - *For $d = 0, \ldots, D-1$:*
    - $\chi_d := x_{\le d}$
    - $v_{\chi_d} \leftarrow \mathsf{Dec}(\mathsf{sk}_d, \mathsf{ct}_{\chi_d})$
    - $\mathsf{otp}_{\chi_d} := \mathsf{QPrO}(\mathsf{Eval}, h_{d,1}, \chi_d \| 0^{D-d}) \| \ldots \| \mathsf{QPrO}(\mathsf{Eval}, h_{d,B}, \chi_d \| 0^{D-d})$
    - $\mathsf{ct}_{\chi_d \| 0} \| \mathsf{ct}_{\chi_d \| 1} := v_{\chi_d} \oplus \mathsf{otp}_{\chi_d}$
  - *Output* $\mathsf{Dec}(\mathsf{sk}_D, \mathsf{ct}_x)$.

Next, we define the helper functions that were used in the definition of the JLLW obfuscation scheme above.

$\mathsf{Expand}_d[\mathsf{pk}_{d+1}](\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$:

$$\text{Output} \begin{cases} \mathsf{Expand}_{d,\mathsf{normal}}[\mathsf{pk}_{d+1}](\chi, \mathsf{info}_\chi) \ \text{ if } \mathsf{flag}_\chi = \mathsf{normal}, \\ \mathsf{Expand}_{d,\mathsf{hyb}}[\mathsf{pk}_{d+1}](\chi, \mathsf{info}_\chi) \ \text{ if } \mathsf{flag}_\chi = \mathsf{hyb}, \\ \mathsf{Expand}_{d,\mathsf{sim}}(\chi, \mathsf{info}_\chi) \ \text{ if } \mathsf{flag}_\chi = \mathsf{sim} \end{cases}$$

$\mathsf{Eval}(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$:

$$\text{Output} \begin{cases} \mathsf{Eval}_{\mathsf{normal}}(\chi, \mathsf{info}_\chi) \ \text{ if } \mathsf{flag}_\chi = \mathsf{normal}, \\ \mathsf{Eval}_{\mathsf{sim}}(\chi, \mathsf{info}_\chi) \ \text{ if } \mathsf{flag}_\chi = \mathsf{sim} \end{cases}$$

$\mathsf{Expand}_{d,\mathsf{normal}}[\mathsf{pk}_{d+1}](\chi, \mathsf{info}_\chi)$:

- Parse $\mathsf{info}_\chi = (C, \{k_{i,j}\}_{d \le i < D, 1 \le j \le B}, s_\chi)$
- Set $s_{\chi\|0} \| r_{\chi\|0} \| s_{\chi\|1} \| r_{\chi\|1} \coloneqq G_{sr}(s_\chi)$
- For $\eta = 0, 1$:
  - $\mathsf{flag}_{\chi\|\eta} \coloneqq \mathsf{normal}$
  - $\mathsf{info}_{\chi\|\eta} \coloneqq (C, \{k_{i,j}\}_{d+1 \le i < D, 1 \le j \le B}, s_{\chi\|\eta})$
  - $\mathsf{ct}_{\chi\|\eta} \coloneqq \mathsf{Enc}(\mathsf{pk}_{d+1}, \mathsf{flag}_{\chi\|\eta}, \chi\|\eta, \mathsf{info}_{\chi\|\eta}; r_{\chi\|\eta})$
- $\mathsf{otp}_\chi \coloneqq H(k_{d,1}, \chi \| 0^{D-d}) \| \ldots \| H(k_{d,B}, \chi \| 0^{D-d})$
- Output $v_\chi \coloneqq (\mathsf{ct}_{\chi\|0} \| \mathsf{ct}_{\chi\|1}) \oplus \mathsf{otp}_\chi$

$\mathsf{Eval}_{\mathsf{normal}}(\chi, \mathsf{info}_\chi)$:

- Parse $\mathsf{info}_\chi = (C, s_\chi)$
- Output $C(\chi)$, computed by evaluating a universal circuit at (C, χ)

$\mathsf{Expand}_{d,\mathsf{hyb}}[\mathsf{pk}_{d+1}, \mathsf{info}_\chi]$:

- Parse $\mathsf{info}_\chi = (C, \{k_{i,j}\}_{d < i < D, 1 \le j \le B}, s_\chi, \beta, \{\sigma_{\chi,j}\}_{1 \le j < \beta}, w_\chi, \{k_{d,j}\}_{\beta < j \le B})$
- Set $s_{\chi\|0} \| r_{\chi\|0} \| s_{\chi\|1} \| r_{\chi\|1} \coloneqq G_{sr}(s_\chi)$
- For $\eta = 0, 1$:
  - $\mathsf{flag}_{\chi\|\eta} \coloneqq \mathsf{normal}$
  - $\mathsf{info}_{\chi\|\eta} \coloneqq (C, \{k_{i,j}\}_{d+1 \le i < D, 1 \le j \le B}, s_{\chi\|\eta})$
  - $\mathsf{ct}_{\chi\|\eta} \coloneqq \mathsf{Enc}(\mathsf{pk}_{d+1}, \mathsf{flag}_{\chi\|\eta}, \chi\|\eta, \mathsf{info}_{\chi\|\eta}; r_{\chi\|\eta})$
- Output

$$\begin{aligned} v_\chi \coloneqq\ & G_v(\sigma_{\chi,1}) \| \ldots \| G_v(\sigma_{\chi,\beta-1}) \| w_\chi \\ & \| [\mathsf{ct}_{\chi\|0} \| \mathsf{ct}_{\chi\|1}]_{\beta+1} \oplus H(k_{d,\beta+1}, \chi \| 0^{D-d}) \| \ldots \\ & \| [\mathsf{ct}_{\chi\|0} \| \mathsf{ct}_{\chi\|1}]_B \oplus H(k_{d,B}, \chi \| 0^{D-d}) \end{aligned}$$

$\mathsf{Expand}_{d,\mathsf{sim}}(\chi, \mathsf{info}_\chi)$:

- Parse $\mathsf{info}_\chi = \{\sigma_{\chi,j}\}_{1\le j\le B}$
- Output $v_\chi := G_v(\sigma_{\chi,1})\|\ldots\|G_v(\sigma_{\chi,B})$

$\mathsf{Eval}_{\mathsf{sim}}(\chi, \mathsf{info}_\chi)$:

- Parse $\mathsf{info}_\chi = y_\chi$
- Output y_χ

Finally, we have the following theorem, which we will prove in Section 8.

Theorem 3.26. *The JLLW obfuscation* $\mathsf{JLLWObf}^{\mathsf{QPrO}}(1^\lambda, C)$ *given in Theorem 3.25 satisfies post-quantum ideal obfuscation (Theorem 3.24) in the quantum-accessible pseudorandom oracle model (Theorem 3.23)*

4 QMA Verification with Strong Completeness

We first define a special class of "ZX" QMA verifiers satisfying a notion of "strong" completeness, which demands that for an honest witness, every ZX measurement the verifier may apply will accept with overwhelming probability.

Definition 4.1 (ZX verifier with strong completeness)**.** A ZX verifier with strong completeness for a QMA language $(\mathcal{L}_{yes}, \mathcal{L}_{no})$ consists of, for each instance H (an n-qubit Hermitian matrix), a family $\{\theta_{H,i}, f_{H,i}\}_{i\in[N]}$ of binary-outcome ZX measurements (Theorem 3.3), where $N = N(n) = 2^{\mathsf{poly}(n)}$ is some (potentially exponential) function of the instance size. It satisfies the following properties.

- **Strong completeness**. For each $H \in \mathcal{L}_{yes}$, there exists a state $|\psi\rangle$ such that for all $i \in [N]$,

$$\left\| M[\theta_{H,i}, f_{H,i}]\,|\psi\rangle \right\|^2 \ge 1 - 2^{-n^3}.$$

- **Soundness**. For each $H \in \mathcal{L}_{no}$ and any state $|\psi\rangle$,

$$\mathop{\mathbb{E}}_{i\leftarrow[N]}\left[\left\| M[\theta_{H,i}, f_{H,i}]\,|\psi\rangle \right\|^2\right] = \mathsf{negl}(n).$$

Theorem 4.2. *Every language in QMA has a ZX verifier with strong completeness (Theorem 4.1).*

The main ingredient to our proof is a lemma that turns any QMA verifier which applies a random projective measurement to the witness into one with strong correctness. The theorem follows from applying the lemma to the protocol given in [MNS16] for QMA verification via single qubit ZX measurements. For completeness, we state the protocol for permuting ZX verifiers in Section 4.1.

Lemma 4.3 (Permuting QMA Verifiers)**.** Let $\mathcal{L} = (\mathcal{L}_{yes}, \mathcal{L}_{no})$ be a QMA language with instance size n. Let $\{(P_j, I - P_j)\}_{j\in\mathcal{J}}$ be a $\mathsf{poly}(n)$-sized set of binary-outcome projective measurements on n qubits and let $(\mathcal{P} = \sum_{j\in\mathcal{J}} p_j P_j, \mathcal{Q} = \sum_{j\in\mathcal{J}} p_j(I - P_j))$ be a POVM which decides $\mathcal{L}$ with correctness a and soundness error b.

Then there is a verifier for $\mathcal{L}$ with strong correctness and soundness error negl which only performs measurements from $\{P_j, I - P_j\}_{j\in J}$.

Proof. The permuting verifier operates on $k = \left\lceil \frac{4}{a-b}\left(\frac{n^3\ln(2)}{a-b} + |\mathcal{J}|\right)\right\rceil + 1$ registers $\mathcal{R}_j$ each containing n qubits. Let List be an ordered list containing each $j \in \mathcal{J}$ a total of $\lfloor kp_j \rfloor$ times. The family of possible measurements the verifier can make is given by all possible permutations of List. The deciding function f accepts if at least $k\frac{a+b}{2}$ of the measurement outcomes are P. In other words, the family of measurements is

$$\{\sigma(\mathsf{List}), f\}_{\sigma\in\mathsf{Sym}_k}$$

The distribution $\mathsf{Samp}(H)$ samples a uniform $\sigma \leftarrow \mathsf{Sym}_k$ and outputs $(\sigma(\mathsf{List}), f)$.

Claim 4.4. The verifier above has strong completeness.

Proof. For any $H \in \mathcal{L}_{yes}$, there exists an n-qubit witness $|w\rangle$ such that $\mathrm{Tr}[\mathcal{P}|w\rangle] \geq a$. The witness for the permuting verifier is k copies of this witness, i.e. $|w\rangle^{\otimes k}$. Since the witness is separable across the registers $\mathcal{R}_j$ and the verifier applies its projectors on disjoint registers, the outcome for each copy is independent. Let S_k be the random variable representing the number of accepting measurement results (outcome P_j). Hoeffding's inequality allows us to bound the probability that the sum of the outcomes differs from its expected value by

$$\Pr_{x_i:i\in[k]}\left[S_k \leq \mathop{\mathbb{E}}_{x_i:i\in k}[S_k] - t\right] \leq 2\exp\left(-2\frac{t^2}{k}\right)$$

The expectation of the summation is

$$\begin{aligned}\sum_{i\in[k]} \mathrm{Tr}\left[P_{\sigma(\mathsf{List}[i])}|w\rangle\right] &= \mathrm{Tr}\left[\sum_{j\in\mathcal{J}} \lfloor kp_j\rfloor P_j|w\rangle\right] \\ &\geq k\,\mathrm{Tr}\left[\sum_{j\in\mathcal{J}} p_i P_j|w\rangle\right] - |\mathcal{J}| \\ &= ka - |\mathcal{J}|\end{aligned}$$

Setting $t = k\frac{a-b}{2} - |\mathcal{J}|$, we have $t \geq 4\frac{n^3\ln(2)}{a-b}$ and $t/k \geq \frac{a-b}{4}$. Therefore the probability of $S_k \leq k\frac{a+b}{2}$, i.e. the verifier rejecting, is

$$\leq 2\exp\left(-2n^3\ln(2)\right) = 2^{-2n^3+1}$$

□

Claim 4.5. The verifier above has $\mathsf{negl}(n)$ soundness error.

Proof. For any $H \notin \mathcal{L}$, every state ρ satisfies $\mathrm{Tr}[\mathcal{P}\rho] \leq b$. We will relate the number of accepting repetitions in the k-wise parallel repetition of the decision procedure to the number of accepting repetitions in the permuted procedure.

For any repetition i in the parallel case, the probability of accepting *any* mixed state is at most b. Thus, conditioned on any outcome of the other repetitions, the probability of repetition i accepting is still at most b. Therefore the probability of obtaining $\geq n$ accepts is upper bounded by the probability of sampling $\geq n$ in a binomial distribution with success rate b for any n. Let S_{par} be

distributed according to this binomial distribution. Hoeffding's inequality bounds the probability of $S_{\mathsf{par}} \geq kb + t$ as

$$\Pr\left[S_{\mathsf{par}} \geq n^3 b + t\right] \leq \exp\left(-2t^2/k\right)$$

Now we show how this relates to the number of accepting repetitions in the permuted procedure. Observe that the parallel procedure can be equivalently stated as sampling a vector random variable $\mathsf{COUNT} \in \mathbb{N}^{|\mathcal{J}|}$ where $\mathsf{COUNT}[j]$ determines the number of times $(P_j, I - P_j)$ is applied, then randomly permuting the corresponding list of projectors. The probability density function is

$$\Pr[\mathsf{COUNT} = \mathsf{count}] = \prod_{j \in [|\mathcal{J}|]} p_j^{\mathsf{count}[j]}$$

Let count^* be the count corresponding to the permuted procedure, i.e. $\mathsf{count}^*[i] = \lfloor kp_i \rfloor$. Note that $\mathbb{E}[\mathsf{COUNT}[i]] = kp_i$, so

$$\|\mathbb{E}[\mathsf{COUNT}[i]] - \mathsf{count}^*\|_1 \leq |\mathcal{J}|$$

We claim that with overwhelming probability,

$$\|\mathsf{COUNT} - \mathbb{E}[\mathsf{COUNT}]\|_1 \leq \frac{a-b}{4}k$$

To see this, consider each parallel repetition to sample an indicator vector indicating which term $j \in \mathcal{J}$ is chosen. Then Theorem 3.2 implies that the probability of $\|\mathsf{COUNT} - \mathbb{E}[\mathsf{COUNT}]\|_1 \geq \frac{a-b}{4}k$ is at most

$$|\mathcal{J}| \exp\left(\frac{-\left(\frac{a-b}{4}k\right)^2}{2(2k + \left(\frac{a-b}{4}k\right)/3)}\right)$$

since $k > n^3/(a-b)$ and $a - b \leq 1$, this probability is negligible in n.

By triangle inequality, with overwhelming probability

$$\|\mathsf{COUNT} - \mathbb{E}[\mathsf{COUNT}]\|_1 \leq \lambda(a-b)/4 + |\mathcal{J}| \tag{1}$$

Claim 4.6. Consider two sequences of projective measurements $(\Pi_1^i)_{i\in[\lambda]}$ and $(\Pi_2^i)_{i\in[\lambda]}$ which are respectively applied to (disjoint) registers $\mathcal{R}_i)_{i\in[\lambda]}$. Let X_1 and X_2 be the random variables denoting the number of times the measurement result is 1 (corresponding to Π_1^i or Π_2^i), respectively. If the number of indices i such that $\Pi_1^i \neq \Pi_2^i$ is at most ℓ, then for any state ρ and any $n \in \mathbb{N}$,

$$\Pr[X_1 \geq n] \leq \Pr[X_2 \geq n - \ell]$$

Proof. The two experiments can be thought of as performing two steps. In the first step we measure all indices where the two sequences match, and then in the second step we measure the remaining indices according to the first sequence in the first experiment, and according to the second sequence in the second experiment. We can obtain greater than equal to n accepting measurements in the first experiment only if in the first step we obtain atleast $n - k$ accepting measurements. Let X represent the number of accepting measurements in the first step. Therefore,

$$\Pr[X_1 \geq n] \leq \Pr[X \geq n - \ell]$$

In the second experiment, if we obtain $n - \ell$ accepting measurements in the first step, the final number of accepting measurements will be at least $n - k$. Therefore,

$$\Pr[X_2 \geq n - k] \geq \Pr[X \geq n - \ell]$$

Putting both together concludes the proof of the claim. □

For any permutation σ, state ρ, and vectors count^* and COUNT, consider the following experiments. In the first experiment, we permute the list of measurements specified by count^* using the permutation σ and apply the measurements to ρ. Let v_0 be the number of accepting measurements. In the second experiment, we perform the parallel procedure: first sample COUNT, then permute it randomly. Let v_1 be the number of accepting measurements in the second experiment. Let

$$\delta := \|\mathsf{COUNT} - \mathsf{count}^*\|_1.$$

By the above claim, for all t and all δ,

$$\Pr[v_0 \geq kb + t] \leq \Pr[v_1 \geq kb + t - \delta].$$

Now consider sampling COUNT as in the parallel repeated experiment. Recall from Equation (1) that with overwhelming probability

$$\delta \leq k\frac{a-b}{4} + |\mathcal{J}|.$$

Additionally, when σ is also sampled randomly, v_1 is distributed as S_{par} which means that

$$\Pr[v_1 \geq kb + t - \delta] \leq \exp\left(\frac{-2(t-\delta)^2}{k}\right).$$

Setting t to be $k\frac{a-b}{3} - |\mathcal{J}|$ we get that with overwhelming probability

$$v_0 \leq k(b + a/3 - b/3) < k\frac{a+b}{2}.$$

Finally, by noting that the probability that the permuting verifier accepts is at most the probability that v_0 exceeds $k\frac{a+b}{2}$, we obtain that the permuting verifier has negligible soundness error. □

□

4.1 Permuting ZX Verifier for QMA

We state here the full verification procedure for the ZX verifier with strong completeness that is guaranteed by Theorem 4.2 for every QMA language. As a corollary of Theorem 4.3 and the ZX verifier from [MNS16], the following is a ZX verifier with strong completeness for any QMA language $\mathcal{L} = (\mathcal{L}_{yes}, \mathcal{L}_{no})$.

Construction 4.7 (Permuting ZX Verifier). *Let H be an instance of the language $(\mathcal{L}_{yes}, \mathcal{L}_{no})$ with completeness and soundness energy thresholds $a', b' \in [-1, 1]$. Without loss of generality [BL08], H is a ZX Hamiltonian*

$$H = \sum_{\substack{i<j<\ell \\ S\in\{Z,X\}}} p_{ij} P_{i,j,S}$$

on $\ell = \ell(n)$ qubits where $p_{ij} \in [0, 1]$ with $\sum_{i<j} 2p_{ij} = 1$ and $P_{i,j,S} = \frac{I+(-1)^{\beta_{i,j}} S_i S_j}{2}$ for $\beta_{i,j} \in \{0, 1\}$.

For each instance size $n \in \mathbb{N}$, define the following.

- *Completeness and soundness thresholds $a = \frac{1}{2}\left(1 - \frac{a'}{\sum_{i<j} p_{i,j}}\right)$ and $b = \frac{1}{2}\left(1 - \frac{b'}{\sum_{i<j} p_{i,j}}\right)$ for [MNS16]'s protocol. Note that $a, b \in [0, 1]$.*
- *A number of repetitions $k \left\lceil \frac{4}{a-b}\left(\frac{n^3 \ln(2)}{a-b} + (2\ell^2)\right)\right\rceil + 1$.*
- $\mathsf{List}_{H,k}$*: A list of $(\theta_{i,j,S}, f_{i,j,S})$ for*
  - *the basis $\theta_{i,j,S} = 0^n$ if $S = Z$, and $\theta_{i,j,S} = 1^n$ otherwise, and*
  - *the function $f_{i,j,S}(m_1\|\ldots\|m_n)$ outputs 1 iff $m_i \oplus m_j \oplus \beta_{i,j} = 1$,*

  where each $(\theta_{i,j,S}, f_{i,j,S})$ appears $\lfloor p_{ij} k \rfloor$ times.
- $(\theta_{H,k,r}, f_{H,k,r}) \leftarrow \mathsf{Samp}(H; r)$*: On input an instance H and randomness r,* Samp *samples a random permutation $\sigma \leftarrow \mathsf{Sym}_k$ using randomness r, computes* PermList $= \sigma(\mathsf{List}_{H,k})$*, and outputs $(\theta_{H,k,r}, f_{H,k,r})$ where*
  - *$\theta_{H,k,r}$ as a concatenation of all $\theta_{i,j,S}$ in* PermList*, and*
  - *$f_{H,k,r}$ as dividing its input in-order amongst the $f_{i,j,S}$ in* PermList *and outputting 1 iff at least $k\frac{a+b}{2}$ of $f_{i,j,S}$ accept their respective inputs.*
- *For $H \in \mathcal{L}_{yes}$, let $|\psi\rangle$ be a state such that $\mathrm{Tr}[\langle\psi| H |\psi\rangle] \geq a$. Then $|\psi\rangle^{\otimes k}$ is a witness for the permuting ZX verifier.*

5 Coset State Authentication

We recall the coset state authentication scheme, first introduced by [BBV24]. We describe a variant of the scheme that does not involve CNOT-homomorphism, and where each qubit is encoded with an independently sampled subspace. We will use the following notation. Given a subspace $S \subset \mathbb{F}_2^{2\lambda+1}$ and a vector $\Delta \in \mathbb{F}_2^{2\lambda+1} \setminus S$, define the subspace

$$S_\Delta \coloneqq S \cup (S + \Delta).$$

Let the dual subspace of S_Δ be $\widehat{S} \coloneqq S_\Delta^\perp$, let $\widehat{\Delta}$ be an arbitrary choice of a vector such that $S^\perp = \widehat{S} \cup (\widehat{S} + \widehat{\Delta})$, and define

$$\widehat{S}_{\widehat{\Delta}} \coloneqq S^\perp = \widehat{S} \cup (\widehat{S} + \widehat{\Delta}).$$

Finally, given a projector Π and a state $|\psi\rangle$, we write $|\psi\rangle \in \mathsf{im}(\Pi)$ to indicate that $\Pi |\psi\rangle = |\psi\rangle$.

5.1 Construction

Construction 5.1 (Coset state authentication)**.** *The coset state authentication scheme is defined by the following algorithms.*

- $\mathsf{KeyGen}(1^\lambda, 1^n)$*: For each* $i \in [n]$*, sample a random subspace* $S_i \subset \mathbb{F}_2^{2\lambda+1}$ *of dimension* λ*, a random vector* $\Delta \in \mathbb{F}_2^{2\lambda+1} \setminus S_i$*, and random vectors* $x_i, z_i \in \mathbb{F}_2^{2\lambda+1}$*. Output* $k := \{S_i, \Delta_i, x_i, z_i\}_{i\in[n]}$*.*
- $\mathsf{Enc}_k(|\psi\rangle)$*: Parametrized by a key* $k = \{S_i, \Delta_i, x_i, z_i\}_{i\in[n]}$*, the encoding algorithm is an* n*-qubit to* $(2\lambda+1)n$*-qubit isometry that first applies*

$$\bigotimes_i |b_i\rangle \to \bigotimes_i |S_i + b_i\Delta_i\rangle\,,$$

and then applies the quantum one-time pad $X^x Z^z$*, where* $x = (x_1, \ldots, x_n)$ *and* $z = (z_1, \ldots, z_n)$*.*

- $\mathsf{Dec}_{k,\theta,f}(v) \to \{0,1\}$*: Parameterized by a key* $k = \{S_i, \Delta_i, x_i, z_i\}_{i\in[n]}$ *and the description of a ZX measurement* θ, f*, the decode algorithm takes as input a vector* $v \in \mathbb{F}_2^{n\cdot(2\lambda+1)}$ *and does the following. Parse* $v = (v_1, \ldots, v_n)$ *where each* $v_i \in \mathbb{F}_2^{2\lambda+1}$*, and, for each* $i \in [n]$*, compute*

$$m_i := \begin{cases} 0 & \text{if } (\theta_i = 0 \text{ and } v_i \in S_i + x_i) \text{ or } (\theta_i = 1 \text{ and } v_i \in \widehat{S}_i + z_i) \\ 1 & \text{if } (\theta_i = 0 \text{ and } v_i \in S_i + \Delta_i + x_i) \text{ or } (\theta_i = 1 \text{ and } v_i \in \widehat{S}_i + \widehat{\Delta}_i + z_i) \\ \perp & \text{otherwise} \end{cases}.$$

If any $m_i = \perp$*, then output* 0*. Otherwise output* $f(m)$*.*

- $\mathsf{Ver}_{k,\theta}(v) \to \{0,1\}$*: Parameterized by a key* $k = \{S_i, \Delta_i, x_i, z_i\}_{i\in[n]}$ *and bases* $\theta \in \{0,1\}^n$*, the verification algorithm takes as input a vector* $v \in \mathbb{F}_2^{n\cdot(2\lambda+1)}$ *and does the following. Parse* $v = (v_1, \ldots, v_n)$ *where each* $v_i \in \mathbb{F}_2^{2\lambda+1}$*, and, for each* $i \in [n]$*, output 0 if* $\theta_i = 0$ *and* $v_i \notin S_{i,\Delta_i} + x_i$ *or* $\theta_i = 1$ *and* $v_i \notin \widehat{S}_{i,\widehat{\Delta}_i} + z_i$*. Otherwise, output 1.*

5.2 Properties

We introduce new properties of this authentication scheme. First, we state some imported lemmas that follow from [BBV24].

Lemma 5.2 (Correctness)**.** For any bases $\theta \in \{0,1\}^n$, function $f : \{0,1\}^n \to \{0,1\}$, and key $k \in \mathsf{KeyGen}(1^\lambda, 1^n)$,

$$\mathsf{Enc}_k^\dagger (H^{\otimes 2\lambda+1})^\theta \left(\sum_{v:\mathsf{Dec}_{k,\theta,f}(v)=1} |v\rangle\langle v| \right) (H^{\otimes 2\lambda+1})^\theta \mathsf{Enc}_k = M[\theta, f].$$

Lemma 5.3 (Hiding)**.** Let Obf be a sub-exponentially secure indistinguishability obfuscator (Theorem 3.20), and suppose that sub-exponentially secure injective one-way functions exist. Then there exists polynomials $d(\cdot,\cdot), q(\cdot,\cdot)$ such that for any two n-qubit states $|\psi_0\rangle, |\psi_1\rangle$, and QPT adversary

$\mathcal{A}$,

$$\left|\Pr\left[\mathcal{A}(|\widetilde{\psi}_0\rangle, \widetilde{\mathsf{Ver}}) = 1 : \begin{array}{r} k \leftarrow \mathsf{KeyGen}(1^d, 1^n) \\ |\widetilde{\psi}_0\rangle \leftarrow \mathsf{Enc}_k(|\psi_0\rangle) \\ \widetilde{\mathsf{Ver}} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}_{k,(\cdot)}(\cdot)) \end{array}\right]\right.$$
$$\left. - \Pr\left[\mathcal{A}(|\widetilde{\psi}_1\rangle, \widetilde{\mathsf{Ver}}) = 1 : \begin{array}{r} k \leftarrow \mathsf{KeyGen}(1^d, 1^n) \\ |\widetilde{\psi}_1\rangle \leftarrow \mathsf{Enc}_k(|\psi_1\rangle) \\ \widetilde{\mathsf{Ver}} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}_{k,(\cdot)}(\cdot)) \end{array}\right]\right| = 2^{-\Omega(\lambda)},$$

where $d := d(\lambda, n)$, and $q := q(\lambda, n)$.

We next show the following characterizing the codespace.

Lemma 5.4. For any key $k \in \mathsf{KeyGen}(1^\lambda, 1^n)$, define

$$\Pi_k := X^x Z^z |S\rangle\langle S| X^x Z^z + X^x Z^z |S+\Delta\rangle\langle S+\Delta| X^x Z^z$$

to be projector onto the image of the isometry Enc_k. Then

$$\Pi_k = \left(H^{\otimes 2\lambda+1}\right)^{1^n} \left(\sum_{v:\mathsf{Ver}_{k,1^n}(v)=1} |v\rangle\langle v|\right) \left(H^{\otimes 2\lambda+1}\right)^{1^n} \left(\sum_{v:\mathsf{Ver}_{k,0^n}(v)=1} |v\rangle\langle v|\right).$$

Proof. We show the claim for $n = 1$, which naturally generalizes to any n. Given a key $k = (S, \Delta, x, z)$, define $\Pi[S_\Delta]$ to be the projector onto $v \in S_\Delta$, define $\Pi[S^\perp]$ analogously, and re-write the RHS on the final line of the claim as

$$H^{\otimes 2\lambda+1} X^z \Pi[S^\perp] X^z H^{\otimes 2\lambda+1} X^x \Pi[S_\Delta] X^x := V_k.$$

Then it suffices to show that (i) $V_k X^x Z^z |S\rangle = X^x Z^z |S\rangle$, (ii) $V_k X^x Z^z |S+\Delta\rangle = X^x Z^z |S+\Delta\rangle$, and (iii) for any $|\psi\rangle$ such that $\Pi_k |\psi\rangle = 0$, $V_k |\psi\rangle = 0$.

The first two follow by inspection, so we just show (iii). Writing

$$X^x Z^z |\psi\rangle = \sum_v \alpha_v |v\rangle,$$

we have that $\sum_{v\in S} \alpha_v = 0$ and $\sum_{v\in S+\Delta} \alpha_v = 0$. Then

$$\begin{aligned}
V_k |\psi\rangle &= H^{\otimes 2\lambda+1} X^z \Pi[S^\perp] X^z H^{\otimes 2\lambda+1} X^x \Pi[S_\Delta] X^x |\psi\rangle \\
&= H^{\otimes 2\lambda+1} X^z Z^x \Pi[S^\perp] H^{\otimes 2\lambda+1} \Pi[S_\Delta] \sum_v \alpha_v |v\rangle \\
&= H^{\otimes 2\lambda+1} X^z Z^x \Pi[S^\perp] H^{\otimes 2\lambda+1} \sum_{v\in S_\Delta} \alpha_v |v\rangle \\
&= H^{\otimes 2\lambda+1} X^z Z^x \Pi[S^\perp] \sum_{v\in S_\Delta} \alpha_v \sum_w (-1)^{v\cdot w} |w\rangle \\
&= H^{\otimes 2\lambda+1} X^z Z^x \sum_{w\in S^\perp} \left(\left(\sum_{v\in S} \alpha_v\right) + (-1)^{\Delta\cdot w} \left(\sum_{v\in S+\Delta} \alpha_v\right)\right) |w\rangle \\
&= 0.
\end{aligned}$$

□

Finally, we prove a new privacy property of the coset state authentication scheme.

Theorem 5.5 (Measurement Indistinguishability)**.** *Let* Obf *be a sub-exponentially secure indistinguishability obfuscator (Theorem 3.20), and suppose that sub-exponentially secure injective one-way functions exist. Then there exists polynomials* $d(\cdot,\cdot), q(\cdot,\cdot)$ *such that for any bases* $\theta \in \{0,1\}^n$*, functions* $f_0, f_1 : \{0,1\}^n \to \{0,1\}$*, and* n*-qubit states* $|\psi_0\rangle, |\psi_1\rangle$ *such that* $|\psi_0\rangle \in \mathsf{im}\,(\mathcal{I} - M[\theta, f_0])$ *and* $|\psi_1\rangle \in \mathsf{im}\,(\mathcal{I} - M[\theta, f_1])$*, it holds that for any QPT adversary* $\mathcal{A}$*,*

$$\left| \Pr\left[\mathcal{A}(|\widetilde{\psi_0}\rangle, \widetilde{\mathsf{Ver}}, \widetilde{\mathsf{Dec}_0}) = 1 : \begin{array}{r} k \leftarrow \mathsf{KeyGen}(1^d, 1^n) \\ |\widetilde{\psi_0}\rangle \leftarrow \mathsf{Enc}_k(|\psi_0\rangle) \\ \widetilde{\mathsf{Ver}} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}_{k,(\cdot)}(\cdot)) \\ \widetilde{\mathsf{Dec}_0} \leftarrow \mathsf{Obf}(1^q, \mathsf{Dec}_{k,\theta,f_0}(\cdot)) \end{array}\right] - \Pr\left[\mathcal{A}(|\widetilde{\psi_1}\rangle, \widetilde{\mathsf{Ver}}, \widetilde{\mathsf{Dec}_1}) = 1 : \begin{array}{r} k \leftarrow \mathsf{KeyGen}(1^d, 1^n) \\ |\widetilde{\psi_1}\rangle \leftarrow \mathsf{Enc}_k(|\psi_1\rangle) \\ \widetilde{\mathsf{Ver}} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}_{k,(\cdot)}(\cdot)) \\ \widetilde{\mathsf{Dec}_1} \leftarrow \mathsf{Obf}(1^q, \mathsf{Dec}_{k,\theta,f_1}(\cdot)) \end{array}\right] \right| = 2^{-\Omega(\lambda)},$$

where $d := d(\lambda, n)$*, and* $q := q(\lambda, n)$*.*

Proof. Let $d = 2 \cdot \max\{n^2, \lambda\}$ and $q = p(d)$, where p is the polynomial from Theorem 3.21. Now, for each $b \in \{0,1\}$, we proceed via the following sequence of hybrids.

- $\mathsf{Hyb}_{0,b}$: This is the distribution over the output of $\mathcal{A}$ as defined in the lemma statement using f_b and $|\psi_b\rangle$.
- $\mathsf{Hyb}_{1,b}$: We "bloat" the subspaces used by Ver. Given $k = \{S_i, \Delta_i, x_i, z_i\}_{i\in[n]}$, define $k' \leftarrow \mathsf{Bloat}(k)$ to be the following procedure.
  - For each $i \in [n]$, sample $T_i \leftarrow \mathsf{Sup}_{3d/2+1}(S_{i,\Delta_i})$, $R_i \leftarrow \mathsf{Sup}_{3d/2+1}(\widehat{S}_{i,\widehat{\Delta}_i})$.
  - Output $k' := \{T_i, R_i, x_i, z_i\}_{i\in[n]}$.

  Then, define $\mathsf{Ver}'_{k',\theta}(v)$ as follows.
  - Parse $v = (v_1, \ldots, v_n)$.
  - For each $i \in [n]$, output 0 if $\theta_i = 0$ and $v_i \notin T_i + x_i$ or $\theta_i = 1$ and $v_i \notin R_i + z_i$. Otherwise, output 1.

  Finally, this hybrid is defined as follows.
  - $k \leftarrow \mathsf{KeyGen}(1^d, 1^n)$
  - $|\widetilde{\psi_b}\rangle \leftarrow \mathsf{Enc}_k(|\psi_b\rangle)$
  - $k' \leftarrow \mathsf{Bloat}(k)$
  - $\widetilde{\mathsf{Ver}'} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}'_{k',(\cdot)}(\cdot))$
  - $\widetilde{\mathsf{Dec}_b} \leftarrow \mathsf{Obf}(1^q, \mathsf{Dec}_{k,\theta,f_b}(\cdot))$
  - Output $\mathcal{A}(\widetilde{H^\theta |x\rangle}, \widetilde{\mathsf{Ver}'}, \widetilde{\mathsf{Dec}_b})$

- $\mathsf{Hyb}_{2,b}$: We measure $|\psi_b\rangle$ in the θ-basis before encoding. Let M_θ denote the n-qubit measurement that measures the i'th qubit in basis θ_i, and $H^\theta|x\rangle \leftarrow M_\theta(|\psi_b\rangle)$ denote the process of applying M_θ to the state $|\psi_b\rangle$. Then $\mathsf{Hyb}_{2,b}$ is defined as follows.
  - $k \leftarrow \mathsf{KeyGen}(1^d, 1^n)$
  - $H^\theta|x\rangle \leftarrow M_\theta(|\psi_b\rangle)$
  - $\widetilde{H^\theta|x\rangle} \leftarrow \mathsf{Enc}_k(H^\theta|x\rangle)$
  - $k' \leftarrow \mathsf{Bloat}(k)$
  - $\widetilde{\mathsf{Ver}'} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}'_{k',(\cdot)}(\cdot))$
  - $\widetilde{\mathsf{Dec}_b} \leftarrow \mathsf{Obf}(1^q, \mathsf{Dec}_{k,\theta,f_b}(\cdot))$
  - Output $\mathcal{A}(\widetilde{H^\theta|x\rangle}, \widetilde{\mathsf{Ver}'}, \widetilde{\mathsf{Dec}_b})$
- $\mathsf{Hyb}_{3,b}$: Sample $\widetilde{\mathsf{Dec}_b}$ as $\widetilde{\mathsf{Dec}_b} \leftarrow \mathsf{Obf}(1^q, \mathsf{null})$, where null is the circuit that always outputs 0.
- $\mathsf{Hyb}_{4,b}$: Undo the measurement from $\mathsf{Hyb}_{2,b}$.
- $\mathsf{Hyb}_{5,b}$: Undo the change in $\mathsf{Ver}_{k,(\cdot)}(\cdot)$ from $\mathsf{Hyb}_{1,b}$.

The proof follows by combining the following sequence of claims.

Claim 5.6. For any $b \in \{0,1\}$,

$$|\Pr[\mathsf{Hyb}_{0,b} = 1] - \Pr[\mathsf{Hyb}_{1,b} = 1]| = 2^{-\Omega(\lambda)}.$$

Proof. This follows directly by applying the security of subspace-hiding obfuscation (Theorem 3.21) for each $i \in [n]$, and using the fact that $d/2 \geq \lambda$. □

Claim 5.7. For any $b \in \{0,1\}$,

$$|\Pr[\mathsf{Hyb}_{1,b} = 1] - \Pr[\mathsf{Hyb}_{2,b} = 1]| = 2^{-\Omega(\lambda)}.$$

Proof. Fix any choice of $y \in \{0,1\}^n$ such that $f_b(y) = 0$, any choice of $y' \neq y$, and consider the following experiment Exp_0.

$\underline{\mathsf{Exp}_0}$

- $k \leftarrow \mathsf{KeyGen}(1^d, 1^n)$
- $\widetilde{H^\theta|y\rangle} \leftarrow \mathsf{Enc}_k(H^\theta|y\rangle)$
- $k' \leftarrow \mathsf{Bloat}(k)$
- $\widetilde{\mathsf{Ver}'} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}'_{k',(\cdot)}(\cdot))$
- $\widetilde{\mathsf{Dec}_b} \leftarrow \mathsf{Obf}(1^q, \mathsf{Dec}_{k,\theta,f_b}(\cdot))$

- $(v_1, \ldots, v_n) \leftarrow \mathcal{A}(\widetilde{H^\theta |y\rangle}, \widetilde{\mathsf{Ver}'}, \widetilde{\mathsf{Dec}_b})$
- Output 1 if
  - for all $i : \theta_i = 0$, $v_i \in S_i + y'_i \cdot \Delta_i + x_i$, and
  - For all $i : \theta_i = 1$, $v_i \in \widehat{S}_i + y'_i \cdot \widehat{\Delta}_i + z_i$.

By Theorem 3.6, it suffices to show that

$$\sqrt{\Pr[\mathsf{Exp}_0 = 1]} \cdot 2^n = 2^{-\Omega(\lambda)}.$$

Next, we'll define experiment Exp_1 with a less-restrictive win condition.

$\underline{\mathsf{Exp}_1}$

- $k \leftarrow \mathsf{KeyGen}(1^d, 1^n)$
- $\widetilde{H^\theta |y\rangle} \leftarrow \mathsf{Enc}_k(H^\theta |y\rangle)$
- $k' \leftarrow \mathsf{Bloat}(k)$
- $\widetilde{\mathsf{Ver}'} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}'_{k',(\cdot)}(\cdot))$
- $\widetilde{\mathsf{Dec}_b} \leftarrow \mathsf{Obf}(1^q, \mathsf{Dec}_{k,\theta,f_b}(\cdot))$
- $(v_1, \ldots, v_n) \leftarrow \mathcal{A}(\widetilde{H^\theta |y\rangle}, \widetilde{\mathsf{Ver}'}, \widetilde{\mathsf{Dec}_b})$
- Output 1 if there exists an $i \in [n]$ such that
  - if $\theta_i = 0$, then $v_i \in S_i + y'_i \cdot \Delta_i + x_i$, or
  - if $\theta_i = 1$, then $v_i \in \widehat{S}_i + y'_i \cdot \widehat{\Delta}_i + z_i$.

Clearly, we have that

$$\Pr[\mathsf{Exp}_0 = 1] \leq \Pr[\mathsf{Exp}_1 = 1].$$

Next, we change variables to make the notation more convenient. Define $f[y] \coloneqq f_b \oplus y$, let $H^{\theta,d} \coloneqq (H^{\otimes 2d+1})^{\theta_1} \otimes \ldots \otimes (H^{\otimes 2d+1})^{\theta_n}$, and consider the following experiment.

$\underline{\mathsf{Exp}_2}$

- $k \leftarrow \mathsf{KeyGen}(1^d, 1^n)$
- $\widetilde{|0^n\rangle} \leftarrow \mathsf{Enc}_k(|0^n\rangle)$
- $k' \leftarrow \mathsf{Bloat}(k)$
- $\widetilde{\mathsf{Ver}'} \leftarrow \mathsf{Obf}(1^q, \mathsf{Ver}'_{k',(\cdot)}(\cdot))$
- $\widetilde{\mathsf{Dec}} \leftarrow \mathsf{Obf}(1^q, \mathsf{Dec}_{k,0^n,f[y]}(\cdot))$
- $(v_1, \ldots, v_n) \leftarrow \mathcal{A}(H^{\theta,d}\widetilde{|0^n\rangle}, \widetilde{\mathsf{Ver}}, \widetilde{\mathsf{Dec}})$.

- Output 1 if there exists an $i \in [n]$ such that $v_i \in S_i + \Delta_i + x_i$

Since this is just a change of variables, we have that

$$\Pr[\mathsf{Exp}_1 = 1] = \Pr[\mathsf{Exp}_2 = 1].$$

Next, we consider n experiments $\mathsf{Exp}_{2,1}, \ldots, \mathsf{Exp}_{2,n}$, where in $\mathsf{Exp}_{2,j}$, the circuit $\mathsf{Dec}_{k,0^n,f[y],j}$, defined as follows, is obfuscated.

$\underline{\mathsf{Dec}_{k,0^n,f[y],j}}$

- Parse $v = (v_1, \ldots, v_n)$ where each $v_i \in \mathbb{F}_2^{2d+1}$
- For each $i \in [0, \ldots, j]$, compute

$$m_i \coloneqq \begin{cases} 0 & \text{if } v_i \in S_i + x_i \\ \bot & \text{otherwise} \end{cases}.$$

- For each $i \in [j+1, \ldots, n]$, compute

$$m_i \coloneqq \begin{cases} 0 & \text{if } v_i \in S_i + x_i \\ 1 & \text{if } v_i \in S_i + \Delta_i + x_i \\ \bot & \text{otherwise} \end{cases}.$$

- If any $m_i = \bot$, then output 0. Otherwise output $f[y](m)$.

Note that Δ_i is sampled as a uniformly random coset of S_i within T_i, which is a set of size $d/2 + 1$. Thus, by Theorem 3.22, we have that

$$|\Pr[\mathsf{Exp}_{2,j-1} = 1] - \Pr[\mathsf{Exp}_{2,j} = 1]| = 2^{-\Omega(d)}$$

for each $j \in [n]$.

Next, since $f[y](0^n) = f(y) = 0$, $\mathsf{Dec}_{k,0^n,f[y],n}$ is the null circuit, and thus

$$\Pr[\mathsf{Exp}_{2,n} = 1] \leq \frac{|S_i|}{|T_i \setminus S_i|} = 2^{-\Omega(d)}.$$

Combining everything so far, we have that $\Pr[\mathsf{Exp}_0 = 1] = 2^{-\Omega(d)}$, which implies that

$$\sqrt{\Pr[\mathsf{Exp}_0 = 1]} \cdot 2^n = 2^{-\Omega(d)} \cdot 2^n = 2^{-\Omega(\lambda)},$$

since $d \geq n^2$.

□

Claim 5.8. For any $b \in \{0, 1\}$,

$$|\Pr[\mathsf{Hyb}_{2,b} = 1] - \Pr[\mathsf{Hyb}_{3,b} = 1]| = 2^{-\Omega(\lambda)}.$$

Proof. This follows via the same sequence of hybrids $\mathsf{Exp}_{2,1}, \ldots, \mathsf{Exp}_{2,n}$ used in the proof of Theorem 5.7. □

Claim 5.9. For any $b \in \{0,1\}$,

$$|\Pr\left[\mathsf{Hyb}_{3,b} = 1\right] - \Pr\left[\mathsf{Hyb}_{4,b} = 1\right]| = 2^{-\Omega(\lambda)}.$$

Proof. This follows from the same argument as the proof of Theorem 5.7 □

Claim 5.10. For any $b \in \{0,1\}$,

$$|\Pr\left[\mathsf{Hyb}_{4,b} = 1\right] - \Pr\left[\mathsf{Hyb}_{5,b} = 1\right]| = 2^{-\Omega(\lambda)}.$$

Proof. This follows from the same argument as the proof of Theorem 5.6 □

Claim 5.11.

$$|\Pr\left[\mathsf{Hyb}_{5,0} = 1\right] - \Pr\left[\mathsf{Hyb}_{5,1} = 1\right]| = 2^{-\Omega(\lambda)}.$$

Proof. This follows directly from Theorem 5.3, since we have exactly the same setup except that the adversary also receives an obfuscated null circuit. □

□

Corollary 5.12. Let Obf be a sub-exponentially secure indistinguishability obfuscator (Theorem 3.20), and suppose that sub-exponentially secure injective one-way functions exist. Then there exists a polynomial $p(\cdot,\cdot)$ such that for any bases $\theta \in \{0,1\}^n$, functions $f_0, f_1 : \{0,1\}^n \to \{0,1\}$, and n-qubit states $|\psi_0\rangle, |\psi_1\rangle$ such that $|\psi_0\rangle \in \mathsf{im}\,(M[\theta, f_0])$ and $|\psi_1\rangle \in \mathsf{im}\,(M[\theta, f_1])$, it holds that for any QPT adversary $\mathcal{A}$,

$$\left| \Pr\left[\mathcal{A}(|\widetilde{\psi_0}\rangle, \widetilde{\mathsf{Ver}}, \widetilde{\mathsf{Dec}_0}) = 1 : \begin{array}{r} k \leftarrow \mathsf{KeyGen}(1^{p(\lambda,n)}, 1^n) \\ |\widetilde{\psi_0}\rangle \leftarrow \mathsf{Enc}_k(|\psi_0\rangle) \\ \widetilde{\mathsf{Ver}} \leftarrow \mathsf{Obf}(\mathsf{Ver}_{k,(\cdot)}(\cdot)) \\ \widetilde{\mathsf{Dec}_0} \leftarrow \mathsf{Obf}(\mathsf{Dec}_{k,\theta,f_0}(\cdot)) \end{array} \right] \right.$$
$$\left. - \Pr\left[\mathcal{A}(|\widetilde{\psi_1}\rangle, \widetilde{\mathsf{Ver}}, \widetilde{\mathsf{Dec}_1}) = 1 : \begin{array}{r} k \leftarrow \mathsf{KeyGen}(1^{p(\lambda,n)}, 1^n) \\ |\widetilde{\psi_1}\rangle \leftarrow \mathsf{Enc}_k(|\psi_1\rangle) \\ \widetilde{\mathsf{Ver}} \leftarrow \mathsf{Obf}(\mathsf{Ver}_{k,(\cdot)}(\cdot)) \\ \widetilde{\mathsf{Dec}_1} \leftarrow \mathsf{Obf}(\mathsf{Dec}_{k,\theta,f_1}(\cdot)) \end{array} \right] \right| = 2^{-\Omega(\lambda)}.$$

Proof. Suppose there exist $\theta, f_0, f_1, |\psi_0\rangle, |\psi_1\rangle, \mathcal{A}$ for which the above does not hold. Define $\overline{f}_0, \overline{f}_1$ to be the complements of f_0, f_1, and note that $|\psi_0\rangle \in \mathsf{im}\left(\mathcal{I} - M[\theta, \overline{f}_0]\right)$, $|\psi_1\rangle \in \mathsf{im}\left(\mathcal{I} - M[\theta, \overline{f}_1]\right)$. For any $b \in \{0,1\}$, given $|\widetilde{\psi_b}\rangle, \widetilde{\mathsf{Ver}}, \widetilde{\mathsf{Dec}_b}'$, where $k \leftarrow \mathsf{KeyGen}(1^{p(\lambda,n)}, 1^n)$, $|\widetilde{\psi_b}\rangle \leftarrow \mathsf{Enc}_k(|\psi_b\rangle)$, $\widetilde{\mathsf{Ver}} \leftarrow \mathsf{Obf}(\mathsf{Ver}_{k,(\cdot)}(\cdot))$, and $\widetilde{\mathsf{Dec}_b}' \leftarrow \mathsf{Obf}(\mathsf{Dec}_{k,\theta,\overline{f}_b}(\cdot))$, consider a reduction that samples

$$\widetilde{\mathsf{Dec}_b}(\cdot) \leftarrow \mathsf{Obf}\left(1 \text{ if } \widetilde{\mathsf{Dec}_b}'(\cdot) = 0 \wedge \widetilde{\mathsf{Ver}}(\cdot) = 1\right)$$

and runs $\mathcal{A}$ on $|\widetilde{\psi_b}\rangle, \widetilde{\mathsf{Ver}}, \widetilde{\mathsf{Dec}_b}$. Note that the program obfuscated is functionally-equivalent to $\mathsf{Dec}_{k,\theta,f_b}(\cdot)$, and thus by the security of Obf, this reduction violates Theorem 5.5. □

6 Post-Quantum NIZK Arguments of Knowledge for NP

6.1 Knowledge Soundness Definition

Definition 6.1 (Post-Quantum NIZKPoK (AoK) for NP in CRS Model)**.** Let NP relation $\mathcal{R}$ with corresponding language $\mathcal{L}$ be given such that they can be indexed by a security parameter $\lambda \in \mathbb{N}$.

$\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ is a post-quantum, non-interactive, zero-knowledge proof (argument) of knowledge for NP in the CRS model if it has the following syntax and properties.

Syntax. The input 1^λ is left out when it is clear from context.

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

Properties.

- **Uniform Random String.** Π satisfies the uniform random string property of Theorem 3.16.
- **Perfect Completeness.** Π satisfies the perfect completeness property of Theorem 3.16.
- **Adaptive T-Proof (Argument) of Knowledge.** There exists a polynomial-size circuit extractor $\mathsf{Ext} = (\mathsf{Ext}_0, \mathsf{Ext}_1)$ and a negligible functions $\mathsf{negl}_0(\cdot), \mathsf{negl}_1(\cdot)$ such that:
  1. for every unbounded (polynomial-size) quantum circuit $\mathcal{D}$, every sufficiently large $\lambda \in \mathbb{N}$,
  $$\left| \Pr_{\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)} [\mathcal{D}(\mathsf{crs}) = 1] - \Pr_{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda)} [\mathcal{D}(\mathsf{crs}) = 1] \right| \leq \mathsf{negl}_0(T(\lambda))$$
  2. and, for every unbounded (polynomial-size) quantum circuit $\mathcal{A}$, every sufficiently large $\lambda \in \mathbb{N}$,
  $$\Pr_{\substack{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}_1(\mathsf{crs}, \mathsf{td}, x, \pi)}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge (x, w) \notin \mathcal{R}_\lambda] \leq \mathsf{negl}_1(T(\lambda)).$$
- **Adaptive Computational (Non-Adaptive Statistical) Zero-Knowledge.** Π satisfies the adaptive computational (non-adaptive statistical) zero-knowledge property of Theorem 3.16.

6.2 Proofs of Knowledge for NP with CRS

Let NP relation $\mathcal{R}$ with corresponding language $\mathcal{L}$ be given such that they can be indexed by a security parameter $\lambda \in \mathbb{N}$. Let Π be a post-quantum NIZK for NP (Theorem 3.16). Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a IND-CPA post-quantum encryption scheme.

$\mathsf{Setup}(1^\lambda)$:

1. Generate a CRS for the NIZK scheme Π. Formally,
   (a) $\mathsf{crs}_\Pi \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$.
2. Sample a public and secret key for the encryption scheme.
   (a) $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.
3. Output $\mathsf{crs} = (\mathsf{crs}_\Pi, \mathsf{pk})$.

$\mathsf{P}(\mathsf{crs}, x, w)$:

1. Compute an encryption of the witness. Formally,
   (a) Compute $\mathsf{ct} = \mathsf{Enc}(\mathsf{pk}, w; r)$ for uniformly random r.
2. Compute a NIZK to prove that the ciphertext contains a witness for the instance. Formally,
   (a) Let $\mathcal{L}_{\mathsf{Enc}}$ be an NP language defined as follows

$$\mathcal{L}_{\mathsf{Enc}} = \left\{ (x, \mathsf{ct}) \ : \ \exists (w, r), \ \begin{array}{c} (x, w) \in \mathcal{R}_\lambda \text{ and} \\ \mathsf{ct} = \mathsf{Enc}(\mathsf{pk}, w; r) \end{array} \right\}.$$

   (b) Compute $\pi_\Pi \leftarrow \Pi.\mathsf{P}(\mathsf{crs}_\Pi, (x, \mathsf{ct}), (w, r))$ with respect to language $\mathcal{L}_{\mathsf{Enc}}$.
3. Output $\pi = (\mathsf{ct}, \pi_\Pi)$.

$\mathsf{V}(\mathsf{crs}, x, \pi)$:

1. Verify that the NIZK verifier accepts the NIZK proof. Formally,
   (a) Verify that $\Pi.\mathsf{V}(\mathsf{crs}_\Pi, (x, \mathsf{ct}), \pi_\Pi) = 1$.
2. Output b.

Theorem 6.2. *Given that*

- Π *is a post-quantum adaptively statistically sound, (adaptively) computationally zero-knowledge NIZK protocol for* NP *with common reference string (Theorem 3.16) and*
- $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a perfectly-correct post-quantum IND-CPA encryption scheme,*

then this construction is a post-quantum adaptive proof of knowledge, (adaptively) computationally zero-knowledge NIZK for NP *with common reference string (Theorem 6.1).*

Proof. **Perfect Completeness.** This follows from perfect completeness of Π.

Adaptive Proof of Knowledge. We define Ext_0 as follows:

Input: 1^λ.

(1) $\mathsf{crs}_\Pi \leftarrow \Pi.\mathsf{Setup}(1^\lambda)$.

(2) $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.

(3) Output $\mathsf{crs} = (\mathsf{crs}_\Pi, \mathsf{pk})$ and $\mathsf{td} = \mathsf{sk}$.

We define Ext_1 as follows:

Input: $\mathsf{crs}, \mathsf{td}, x, \pi$.

(1) Output $w := \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$.

Since Ext_0 and Setup output crs from identical distributions, we have that for every unbounded (polynomial-size) quantum circuit $\mathcal{D}$, every sufficiently large $\lambda \in \mathbb{N}$,

$$\left| \Pr_{\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda)} [\mathcal{D}(\mathsf{pp}) = 1] - \Pr_{(\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda)} [\mathcal{D}(\mathsf{pp}) = 1] \right| = 0.$$

We now argue by contradiction that the extractor $(\mathsf{Ext}_0, \mathsf{Ext}_1)$ satisfies the second property. Let a polynomial $p(\cdot)$ and an oracle-aided polynomial-size quantum circuit $\mathcal{A}$ be given such that for every sufficiently large $\lambda \in \mathbb{N}$,

$$\Pr_{\substack{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}_1(\mathsf{crs}, \mathsf{td}, x, \pi)}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge (x, w) \notin \mathcal{R}_\lambda] \geq \frac{1}{p(\lambda)}.$$

We consider two scenarios: either $(x, \mathsf{ct}) \notin \mathcal{L}_{\mathsf{Enc}}$, or $(x, \mathsf{ct}) \in \mathcal{L}_{\mathsf{Enc}}$. At least one of these scenarios must occur with at least $1/(2p(\lambda))$ probability. We will show that both scenarios reach a contradiction.

Scenario One

Consider that

$$\Pr_{\substack{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}_1(\mathsf{crs}, \mathsf{td}, x, \pi)}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge (x, w) \notin \mathcal{R}_\lambda \wedge (x, \mathsf{ct}) \notin \mathcal{L}_{\mathsf{Enc}}] \geq \frac{1}{2p(\lambda)}.$$

If the verifier V accepts a proof π, then the verifier of the NIZK $\Pi.\mathsf{V}$ accepts the proof π_Π. Hence,

$$\Pr_{\substack{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}_1(\mathsf{crs}, \mathsf{td}, x, \pi)}} [\Pi.\mathsf{V}(\mathsf{crs}_\Pi, (x, \mathsf{ct}), \pi_\Pi) = 1 \wedge (x, w) \notin \mathcal{R}_\lambda \wedge (x, \mathsf{ct}) \notin \mathcal{L}_{\mathsf{Enc}})] \geq \frac{1}{2p(\lambda)}.$$

However, this contradicts the adaptive statistical (computational) soundness of Π.

Scenario Two

Consider that

$$\Pr_{\substack{(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ w \leftarrow \mathsf{Ext}_1(\mathsf{crs}, \mathsf{td}, x, \pi)}} [\mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge (x, w) \notin \mathcal{R}_\lambda \wedge (x, \mathsf{ct}) \in \mathcal{L}_{\mathsf{Enc}}] \geq \frac{1}{2p(\lambda)}.$$

If $(x, \mathsf{ct}) \in \mathcal{L}_{\mathsf{Enc}}$ then there exists (w, r) such that $(x, w) \in \mathcal{R}_\lambda$ and $\mathsf{ct} = \mathsf{Enc}(\mathsf{pk}, w; r)$. Coupled with the perfect correctness of the encryption scheme, this means that $(x, \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})) \in \mathcal{R}_\lambda$. However, by the definition of Ext_1, this contradicts with $(x, w) \notin \mathcal{R}_\lambda$.

Therefore, our protocol must be an adaptive proof of knowledge.

(Adaptive) Computational Zero-Knowledge. Let $\Pi.\mathsf{Sim} = (\Pi.\mathsf{Sim}_0, \Pi.\mathsf{Sim}_1)$ be the (adaptive) computational zero-knowledge simulator of the NIZK Π. We define Sim_0 as follows:

Input: 1^λ

(1) Compute $\mathsf{crs}_\Pi \leftarrow \Pi.\mathsf{Sim}_0(1^\lambda)$.

(2) Sample $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.

(3) Output $\mathsf{crs} = (\mathsf{crs}_\Pi, \mathsf{pk})$ and $\mathsf{td} = \mathsf{sk}$.

We define Sim_1 as follows:

Input: crs, x

(1) Compute $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$.

(2) Compute $\pi_\Pi \leftarrow \Pi.\mathsf{Sim}_1(\mathsf{crs}_\Pi, (x, \mathsf{ct}))$.

(3) Output $\pi = (\mathsf{ct}, \pi_\Pi)$.

Let a polynomial-size quantum circuit $\mathcal{D} = (\mathcal{D}_0, \mathcal{D}_1)$, and sufficiently large $\lambda \in \mathbb{N}$ be given. We construct the following series of hybrids to argue computational indistinguishability of the honest $\mathcal{H}_0$ and simulated $\mathcal{H}_3$ distributions:

$\mathcal{H}_0$ **:** $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda)$. $(x, w, \zeta) \leftarrow \mathcal{D}_0(\mathsf{crs})$. $\pi \leftarrow \mathsf{P}(\mathsf{crs}, x, w)$.

$\mathcal{H}_1$ **:** $\mathsf{crs}_\Pi \leftarrow \Pi.\mathsf{Sim}_0(1^\lambda)$. $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$. $\mathsf{crs} = (\mathsf{crs}_\Pi, \mathsf{pk})$. $(x, w, \zeta) \leftarrow \mathcal{D}_0(\mathsf{crs})$. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, w)$. $\pi_\Pi \leftarrow \Pi.\mathsf{Sim}_1(\mathsf{crs}_\Pi, (x, \mathsf{ct}))$. $\pi = (\mathsf{ct}, \pi_\Pi)$.

$\mathcal{H}_2$ **:** $\mathsf{crs}_\Pi \leftarrow \Pi.\mathsf{Sim}_0(1^\lambda)$. $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$. $\mathsf{crs} = (\mathsf{crs}_\Pi, \mathsf{pk})$. $(x, w, \zeta) \leftarrow \mathcal{D}_0(\mathsf{crs})$. $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$. $\pi_\Pi \leftarrow \Pi.\mathsf{Sim}_1(\mathsf{crs}_\Pi, (x, \mathsf{ct}))$. $\pi = (\mathsf{ct}, \pi_\Pi)$.

$\mathcal{H}_3$ **:** $\mathsf{crs} \leftarrow \mathsf{Sim}_0(1^\lambda)$. $(x, w, \zeta) \leftarrow \mathcal{D}_0(\mathsf{crs})$. $\pi \leftarrow \Pi.\mathsf{Sim}(\mathsf{crs}, x)$.

$\mathcal{H}_0$ and $\mathcal{H}_1$ are computationally indistinguishable by the post-quantum adaptive computational zero-knowledge of Π. $\mathcal{H}_1$ and $\mathcal{H}_2$ are computationally indistinguishable by the post-quantum IND-CPA property of encryption. $\mathcal{H}_2$ and $\mathcal{H}_3$ are identical. Therefore, our protocol is adaptive computational zero-knowledge. □

Corollary 6.3. Assuming the polynomial quantum hardness of LWE, there exists an adaptive proof of knowledge, adaptively computationally zero-knowledge NIZKPoK for NP having a common reference string (Theorem 6.1).

Proof. This follows from Theorem 6.2, Theorem 3.17, and Theorem 3.14. □

6.3 Adaptively Sound Arguments for NP with URS

It is well-known that any NIZK with non-adaptive, but sub-exponential, soundness can be compiled to an adaptively sound NIZK using complexity leveraging. We present its proof for completeness.

Theorem 6.4. *Let $\mathcal{R}$ be an* NP *relation indexed by $\lambda \in \mathbb{N}$ such that $\mathcal{R}_\lambda$ has instance size λ^c for some parameter $c < 1$. Assuming a non-interactive zero knowledge argument for* NP *with sub-exponential computational soundness error and statistical zero knowledge, then for every $c < 1$, there exists an* adaptively *sub-exponential computationally sound, (not necessarily adaptively) statistically zero-knowledge non-interactive protocol for $\mathcal{R}$ having a uniform random string (Theorem 3.16).*

Proof. Let Π be the non-interactive zero knowledge argument for NP with sub-exponential computational soundness error and statistical zero knowledge. Set $T(\lambda) = 2^{\lambda^c}$ and $\mathsf{negl}(T(\lambda))$ for the non-adaptive computational T-soundness of Π. This is possible since the NIZK Π has sub-exponential soundness error. We show that Π with these parameters is adaptively sound by reducing to the $\mathsf{negl}(2^{\lambda^c})$ soundness error.

We can decompose any adversary's computational advantage into their advantage when conditioning on the instance they output to see that

$$\begin{aligned}
&\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\(x,\pi)\leftarrow\mathcal{A}(\mathsf{crs})}}\left[\begin{array}{c}\mathsf{V}(\mathsf{crs},x,\pi)=1\\ \wedge\, x\notin\mathcal{L}_\lambda\end{array}\right]\\
&=\sum_{x\in\{0,1\}^{\lambda^c}:x\notin\mathcal{L}_\lambda}\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\(x',\pi)\leftarrow\mathcal{A}(\mathsf{crs})}}\left[\begin{array}{c}\mathsf{V}(\mathsf{crs},x',\pi)=1\\ \wedge\, x'\notin\mathcal{L}_\lambda\end{array}\;\middle|\; x'=x\right]\cdot\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\(x',\pi)\leftarrow\mathcal{A}(\mathsf{crs})}}[x=x']\\
&\leq\sum_{x\in\{0,1\}^{\lambda^c}:x\notin\mathcal{L}_\lambda}\mathsf{negl}(2^{\lambda^c})\cdot\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\(x',\pi)\leftarrow\mathcal{A}(\mathsf{crs})}}[x=x']\\
&\leq\mathsf{negl}(2^{\lambda^c})
\end{aligned}$$

where the first inequality follows from the non-adaptive $\mathsf{negl}(2^{\lambda^c})$ soundness error. □

6.4 Arguments of Knowledge for NP with URS

Theorem 6.5. *Let $\mathcal{R}$ be an* NP *relation indexed by $\lambda \in \mathbb{N}$ such that $\mathcal{R}_\lambda$ has instance size λ^c for some parameter $c < 1$. Given that*

- *Π is a post-quantum adaptively (sub-exponentially) computationally sound, computationally zero-knowledge NIZK protocol for $\mathcal{R}$ with uniformly random string (Theorem 3.16) and*
- *$(\mathsf{Gen},\mathsf{Enc},\mathsf{Dec})$ is a post-quantum IND-CPA encryption scheme with uniformly random public keys,*

then for every $c < 1$, this construction is a post-quantum adaptive (sub-exponentially) argument of knowledge, computationally zero-knowledge NIZKAoK for $\mathcal{R}$ with uniformly random string (Theorem 6.1).

Proof. This follows by observing the proof of Theorem 6.2. □

Corollary 6.6. Let $\mathcal{R}$ be an NP relation indexed by $\lambda \in \mathbb{N}$ such that $\mathcal{R}_\lambda$ has instance size λ^c for some parameter $c < 1$. Assuming the sub-exponential quantum hardness of LWE, for every $c < 1$,

there exists an adaptive sub-exponential argument of knowledge, statistically zero-knowledge non-interactive protocol for $\mathcal{R}$ having a uniform random string (Theorem 6.1).

Proof. This follows from Theorem 6.5, Theorem 6.4, Theorem 3.19, and Theorem 3.14. □

7 Provably-Correct Obfuscation

7.1 Definition

We define a notion of provably-correct obfuscation with two security properties: (standard) indistinguishability security and a notion of *secure composition* for obfuscating evasive families of circuits.

Definition 7.1 (Provably-correct obfuscation)**.** A provably-correct obfuscator has the following syntax.

- $\mathsf{pp} \leftarrow \mathsf{Setup}(1^{\lambda_{\mathsf{ZK}}})$. The setup algorithm takes as input the zero-knowledge security parameter $1^{\lambda_{\mathsf{ZK}}}$ and outputs public parameters pp. We say that the obfuscator is in the *URS* (uniform random string) model if pp just consists of uniform randomness.
- $\widetilde{C} \leftarrow \mathsf{Obf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \varphi, C)$. The obfuscation algorithm takes as input the obfuscation security parameter $1^{\lambda_{\mathsf{OBF}}}$, public parameters pp, a predicate φ, and a circuit C. It outputs an obfuscated circuit $\widetilde{C}$ that satisfies predicate φ.
- $y \leftarrow \mathsf{Eval}(\widetilde{C}, x)$. The evaluation algorithm takes as input an obfuscated circuit $\widetilde{C}$ and an input x and outputs y.
- $\mathsf{Ver}(\mathsf{pp}, \varphi, \widetilde{C}) \in \{0,1\}$. The verification algorithm takes as input the public parameters pp, a predicate φ, and an obfuscated circuit $\widetilde{C}$, and outputs either accept or reject.

It should satisfy the following properties.

- **Functionality-preservation**. Let C be any circuit and φ be any predicate such that $\varphi(C) = 1$. For all pp and randomness r, $\widetilde{C} := \mathsf{Obf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \varphi, C; r)$, and x, it holds that $\mathsf{Eval}(\widetilde{C}, x) = C(x)$.
- **Completeness**. For any circuit C,

$$\Pr_{\substack{\mathsf{pp} \leftarrow \mathsf{Setup}(1^{\lambda_{\mathsf{ZK}}}) \\ \widetilde{C} \leftarrow \mathsf{Obf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \varphi, C)}} \left[\mathsf{Ver}(\mathsf{pp}, \varphi, \widetilde{C}) = 1\right] = 1.$$

- **Adaptive Knowledge Soundness**. There exist PPT algorithms $\mathsf{Ext}_0, \mathsf{Ext}_1$ such that

$$\left\{\mathsf{pp} : \mathsf{pp} \leftarrow \mathsf{Setup}(1^{\lambda_{\mathsf{ZK}}})\right\} \approx_{\mathsf{negl}(\lambda_{\mathsf{ZK}})} \left\{\mathsf{pp} : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^{\lambda_{\mathsf{ZK}}})\right\},$$

  and, for any QPT adversary $\mathcal{A}$,

$$\Pr\left[\begin{array}{l} \Big(\mathsf{Ver}(\mathsf{pp}, \varphi, \widetilde{C}) = 0\Big) \vee \\ \Big(\forall x, C(x) = \mathsf{Eval}(\widetilde{C}, x) \wedge \varphi(C) = 1\Big) \end{array} : \begin{array}{r} (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^{\lambda_{\mathsf{ZK}}}) \\ \widetilde{C} \leftarrow \mathcal{A}(\mathsf{pp}) \\ C \leftarrow \mathsf{Ext}_1(\mathsf{pp}, \mathsf{td}, \varphi, \widetilde{C}) \end{array}\right] = 1 - \mathsf{negl}(\lambda_{\mathsf{ZK}}).$$

If this property holds against *unbounded* adversaries $\mathcal{A}$, then we say the scheme has **statistical knowledge soundness**.

- **Simulation Security** There exists a PPT simulator $\mathsf{Sim} = (\mathsf{SimGen}, \mathsf{SimObf})$ such that the following properties hold.
  - **Honest-to-Simulated Indistinguishability.** For every circuit C and every polynomial-size predicate φ such that $\varphi(C) = 1$,

$$\left\{(\mathsf{pp}, \widetilde{C}) : \begin{array}{r} \mathsf{pp} \leftarrow \mathsf{Setup}(1^{\lambda_{\mathsf{ZK}}}) \\ \widetilde{C} \leftarrow \mathsf{Obf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \varphi, C) \end{array}\right\} \approx_{\mathsf{negl}(\lambda_{\mathsf{ZK}})} \left\{(\mathsf{pp}, \widetilde{C}) : \begin{array}{r} (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{SimGen}(1^{\lambda_{\mathsf{ZK}}}) \\ \widetilde{C} \leftarrow \mathsf{SimObf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C) \end{array}\right\}.$$

  - **(Sub-exponential) Simulated-Circuit Indistinguishability**. There exists a constant $\epsilon > 0$ such that for every quantum polynomial-size circuit $\mathcal{A}$, every C_0, C_1 such that $C_0 \equiv C_1$, and every polynomial-size predicate φ,

$$\left| \Pr_{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{SimGen}(1^{\lambda_{\mathsf{ZK}}})} \left[\mathcal{A}\left(\mathsf{SimObf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C_0)\right) = 1\right] - \Pr_{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{SimGen}(1^{\lambda_{\mathsf{ZK}}})} \left[\mathcal{A}\left(\mathsf{SimObf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C_1)\right) = 1\right] \right| \leq 2^{-\lambda_{\mathsf{OBF}}^{\epsilon}}.$$

  - $\mathcal{S}$**-evasive composability**. This property is parameterized by a sampler S that outputs a set of circuits $\{C_i\}_{i\in[N]}$ along with side information in the form of circuit C and state $|\psi\rangle$. Let C_{null} be the null circuit, and, given a set of circuits $\{C_i\}_i$ and auxiliary circuit C, define the circuit $C||\mathsf{Combine}(\{C_i\}_i)$ to map $(i, x) \to C_i(x)$ for $i > 0$ and $(0, x)$ to $C(x)$. Likewise, let $C||C'$ be the circuit that maps $(0, x)$ to $C(x)$ and $(1, x)$ to $C'(x)$.

    IF for any $i \in [N]$, any predicate φ, and any QPT adversary $\mathcal{A}$:

$$\left| \Pr_{\substack{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{SimGen}(1^{\lambda_{\mathsf{ZK}}}) \\ (|\psi\rangle, C, \{C_j\}_j) \leftarrow \mathcal{S}}} \left[\mathcal{A}\left(|\psi\rangle, \mathsf{SimObf}\left(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C||C_i\right)\right) = 1\right] - \Pr_{\substack{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{SimGen}(1^{\lambda_{\mathsf{ZK}}}) \\ (|\psi\rangle, \{C_j\}_j) \leftarrow \mathcal{S}}} \left[\mathcal{A}\left(|\psi\rangle, \mathsf{SimObf}\left(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C||C_{\mathsf{null}}\right)\right) = 1\right] \right| = \mathsf{negl}(\lambda_{\mathsf{OBF}})/N,$$

    THEN:

$$\left| \Pr_{\substack{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{SimGen}(1^{\lambda_{\mathsf{ZK}}}) \\ (|\psi\rangle, C, \{C_j\}_j) \leftarrow \mathcal{S}}} \left[\mathcal{A}\left(|\psi\rangle, \mathsf{SimObf}\left(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C||\mathsf{Combine}\left(\{C_i\}_i\right)\right)\right) = 1\right] - \Pr_{\substack{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{SimGen}(1^{\lambda_{\mathsf{ZK}}}) \\ (|\psi\rangle, \{C_j\}_j) \leftarrow \mathcal{S}}} \left[\mathcal{A}\left(|\psi\rangle, \mathsf{SimObf}\left(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C||\mathsf{Combine}(\{C_{\mathsf{null}}\}_i)\right)\right) = 1\right] \right| = \mathsf{negl}(\lambda_{\mathsf{OBF}}).$$

7.2 Construction in QPrO Model

In this section, we show how to modify the JLLW construction [JLLW23] to permit provable correctness while still satisfying ideal obfuscation. The main technical nuance is that the keys and handles used in the construction may only be verified through an oracle interface.

First, we confirm in Theorem 7.2 below that any *ideal* obfuscator satisfies $\mathcal{S}$-evasively composability for any choice of sampler $\mathcal{S}$.

Lemma 7.2. If SimObf is an ideal obfuscator, then it satisfies S-evasive composeability for all samplers S.

Proof. Observe that the pre-condition of $\mathcal{S}$-evasive composability implies that no adversary given $\hat{C} \leftarrow \mathsf{SimObf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C_i)$ can find an input x such that $C(x) \neq 0$, except with $\mathsf{negl}(\lambda_{\mathsf{OBF}})/N$ probability.

Consider the following sequence of hybrids.

- Hyb_0: The original experiment where the adversary is run on $\mathsf{SimObf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{Combine}(\{C_i\}_i)$.
- Hyb_1: Instead of running the adversary using the obfuscation, run it using the ideal-world simulator $\mathcal{S}$. $\mathcal{S}$ only has oracle access to $\mathsf{Combine}(\{C_j\}_j$.
- Hyb_2: Replace the simulator's oracle access to $\mathsf{Combine}(\{C_j\}_j)$ by oracle access to the null program $C_\perp$.
- Hyb_3: Replace the simulator by $\mathsf{SimObf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C_\perp)$.

$\mathsf{Hyb}_0 \approx \mathsf{Hyb}_1$ and $\mathsf{Hyb}_2 \approx \mathsf{Hyb}_3$ by the security of ideal obfuscation. The main step is to show that $\mathsf{Hyb}_1 \approx \mathsf{Hyb}_2$. Theorem 3.5 shows that any adversary distinguishing the two must have noticeable probability of querying an input where the oracles $\mathsf{Combine}(\{C_j\}_j$ and $C_\perp$ differ – this holds in particular when considering the queries the adversary makes to $C_\perp$. Let ϵ be the probability of this occurring. Any differing input has the form (i, x) such that $C_i(x) \neq \perp$. Since there are N possible choices of i, there is at least one i^* such that the probability of outputting a differing input with $i = i^*$ is $\geq \epsilon/N$. But then the adversary could find an input x such that $C_{i^*}(x) \neq \perp$ with probability better than $\mathsf{negl}(\lambda_{\mathsf{OBF}})/N$ using only $\hat{C} \leftarrow \mathsf{SimObf}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C_{i^*})$, contradicting our earlier observation. □

Throughout the remainder of the section, we set $\lambda := \lambda_{\mathsf{ZK}}$.

Construction 7.3 (Provably-Correct Obfuscation in the $(\lambda+1)$-QPrO Model)**.** *We construct a provably-correct obfuscator in the quantum-accessible pseudorandom oracle model (Theorem 3.23). The obfuscator is defined as follows in the $(\lambda+1)$-QPrO model defined in Theorem 3.23, using PRF H.*

We first specify the ingredients:

- *Let* Com *be a sub-exponentially secure statistically binding non-interactive commitment scheme.*
- *Let* JLLWObf *be the (sub-exponentially secure) obfuscation scheme specified in Construction 3.25. We will make the following modification to the definition of the obfuscator. Instead of sampling key and handle pairs (k_{ij}, h_{ij}) internally in the second step,* JLLWObf *instead takes the set of key and*

handle pairs as an additional input. Note that if these pairs are sampled honestly this does not affect correctness or security.

- *Let* $(\mathsf{NIZK.Setup}, \mathsf{NIZK.P}, \mathsf{NIZK.V})$ *be a post-quantum non-interactive zero-knowledge argument of knowledge (Theorem 6.1) for an NP relation* $\mathcal{R}_\lambda$ *to be specified later.*

We now define the algorithms for the obfuscation scheme:

- $\mathsf{Setup}^{\mathsf{QPrO}}(1^\lambda)$*:*
  - $\mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(1^\lambda)$
  - $h^* \leftarrow \{0,1\}^\lambda$
  - *Return* $\mathsf{pp} := (\mathsf{crs}, h^*)$
- $\mathsf{Obf}^{\mathsf{QPrO}}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, C)$*:*
  - *Parse* pp *as* (crs, h^*)
  - *Let* D *be the input length of* C *and let* B *be the number of blocks (determined as in [JLLW23]).*
  - *For each* $t \in [\lambda]$*:*
    - $k^t_{i,j} \leftarrow \{0,1\}^{\lambda_{\mathsf{OBF}}}$ *for* $0 \le i < D, 1 \le j \le B$
    - $h^t_{i,j} \leftarrow \mathsf{QPrO}_t(\mathsf{Gen}, k^t_{i,j})$ *for* $0 \le i < D, 1 \le j \le B$
    - $\overline{k}_t := \{k^t_{i,j}\}_{i,j}$
    - $\overline{h}_t := \{h^t_{i,j}\}_{i,j}$
    - $r_t \leftarrow \{0,1\}^*$
    - $c_t \leftarrow \mathsf{com}(\overline{k}_t; r_t)$
  - $\mathsf{chal} := \mathsf{QPrO}_0(\mathsf{Eval}, h^*, (c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda))$ *where* $\mathsf{chal} \in \{0,1\}^\lambda$
  - *Let* $\mathsf{Open}(\mathsf{chal}) := \{t : \mathsf{chal}_t = 1\}$*. For* $t \notin \mathsf{Open}(\mathsf{chal})$ *:*
    - $\widetilde{r}_t \leftarrow \{0,1\}^*$
    - $\widetilde{C}_t \leftarrow \mathsf{JLLWObf}(1^\lambda, C, \overline{k}_t, \overline{h}_t; \widetilde{r}_t)$*. We note here that since we provide key and handle pairs as input,* $\mathsf{JLLWObf}$ *does not query the* QPrO *oracle.*
  - *For any* x *of the form* $(\varphi, \mathsf{chal}, \{c_t, \overline{h}_t, \widetilde{C}_t\}_{t \notin \mathsf{Open}(\mathsf{chal})})$ *and* w *of the form* $(C, \{r_t, \overline{k}_t, \widetilde{r}_t\}_{t \notin \mathsf{Open}(\mathsf{chal})})$ *define the NP relation* $\mathcal{R}_\lambda$ *as*

$$\mathcal{R}_\lambda := \left\{ (x,w) : \begin{array}{r} \varphi(C) = 1 \wedge \\ \forall t \notin \mathsf{Open}(\mathsf{chal}),\ c_t = \mathsf{com}(\overline{k}_t; r_t) \wedge \\ \forall t \notin \mathsf{Open}(\mathsf{chal}),\ \widetilde{C}_t = \mathsf{JLLWObf}(1^\lambda, C, \overline{k}_t, \overline{h}_t; \widetilde{r}_t) \end{array} \right\}$$

  - *Compute* $\pi \leftarrow \mathsf{NIZK.P}(1^\lambda, \mathsf{crs}, x, w)$ *for* $x := (\varphi, \mathsf{chal}, \{c_t, \overline{h}_t, \widetilde{C}_t\}_{t \notin \mathsf{Open}(\mathsf{chal})})$ *and* $w := (C, \{r_t, \overline{k}_t, \widetilde{r}_t\}_{t \notin \mathsf{Open}(\mathsf{chal})})$
  - *Return* $\widetilde{C} := (\{c_t, \overline{h}_t\}_{t \in [\lambda]}, \mathsf{chal}, \{\widetilde{C}_t\}_{t \notin \mathsf{Open}(\mathsf{chal})}, \{\overline{k}_t, r_t\}_{t \in \mathsf{Open}(\mathsf{chal})}, \pi)$

- $\mathsf{Ver}^{\mathsf{QPrO}}(\mathsf{pp}, \varphi, \widetilde{C})$:
  - *Parse* $\widetilde{C}$ *as* $(\{c_t, \overline{h}_t\}_{t\in[\lambda]}, \mathsf{chal}, \{\widetilde{C}_t\}_{t\notin\mathsf{Open}(\mathsf{chal})}, \{\overline{k}_t, r_t\}_{t\in\mathsf{Open}(\mathsf{chal})}, \pi)$. *Reject if any term is missing.*
  - *Parse* pp *as* (crs, h^*)
  - *Check that* $\mathsf{chal} = \mathsf{QPrO}_0(\mathsf{Eval}, h^*, (c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda))$. *Reject otherwise.*
  - *Check that for all* $t \in \mathsf{Open}(\mathsf{chal})$, $c_t = \mathsf{com}(\overline{k}_t; r_t)$. *Reject otherwise.*
  - *For each* $t \in \mathsf{Open}(\mathsf{chal})$ *parse* $\overline{h}_t$ *as* $\{h^t_{i,j}\}_{i,j}$ *and* $\overline{k}_t$ *as* $\{k^t_{i,j}\}_{i,j}$, *where* $0 \le i < D$ *and* $1 \le j \le B$.
  - *For each* $t \in \mathsf{Open}(\mathsf{chal})$, $0 \le i < D$ *and* $1 \le j \le B$:
    - *Check that* $h^t_{i,j} = \mathsf{QPrO}_t(\mathsf{Gen}, k^t_{i,j})$. *Reject otherwise.*
  - *Define* $x := (\varphi, \mathsf{chal}, \{c_t, \overline{h}_t, \widetilde{C}_t\}_{t\notin\mathsf{Open}(\mathsf{chal})})$
  - *If* $\mathsf{NIZK.V}(1^\lambda, \mathsf{crs}, x, \pi) = 1$ *then accept, else reject.*
- $\mathsf{Eval}^{\mathsf{QPrO}}(\widetilde{C}, z)$:
  - *Parse* $\widetilde{C}$ *as* $(\{c_t, \overline{h}_t\}_{t\in[\lambda]}, \mathsf{chal}, \{\widetilde{C}_t\}_{t\notin\mathsf{Open}(\mathsf{chal})}, \{\overline{k}_t, r_t\}_{t\in\mathsf{Open}(\mathsf{chal})})$
  - *For each* $t \notin \mathsf{Open}(\mathsf{chal})$:
    - $y_t := \mathsf{JLLWObf.Eval}^{\mathsf{QPrO}_t}(\widetilde{C}_t, \overline{h}_t, z)$
  - *Return the most frequent element in* $\{y_t\}_{t\notin\mathsf{Open}(\mathsf{chal})}$ *breaking ties arbitrarily.*

Theorem 7.4. *The obfuscator given in Theorem 7.3 satisfies Theorem 7.1 for all samplers* $\mathcal{S}$.

Proof. We prove that the construction satisfies the following properties.

Knowledge Soundness. We define Ext_0 and Ext_1 as follows.

- $\mathsf{Ext}_0(1^\lambda)$: Sample $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{NIZK.Ext}_0(1^\lambda)$, $h^* \leftarrow \{0,1\}^\lambda$, and return $((\mathsf{crs}, h^*), \mathsf{td})$.
- $\mathsf{Ext}_1(\mathsf{pp}, \mathsf{td}, \varphi, \widetilde{C}, \pi)$ does the following.
  - Parse $\widetilde{C}$ as $(\{c_t, \overline{h}_t\}_{t\in[\lambda]}, \mathsf{chal}, \{\widetilde{C}_t\}_{t\notin\mathsf{Open}(\mathsf{chal})}, \{\overline{k}_t, r_t\}_{t\in\mathsf{Open}(\mathsf{chal})})$. Output $\perp$ if any term is missing.
  - Parse pp as (crs, h^*)
  - Define $x := (\varphi, \mathsf{chal}, \{c_t, \overline{h}_t, \widetilde{C}_t\}_{t\notin\mathsf{Open}(\mathsf{chal})})$
  - Run $\mathsf{NIZK.Ext}_1(\mathsf{crs}, \mathsf{td}, x, \pi)$ to obtain $w = (C, \{r_t, \overline{k}_t, \widetilde{r}_t\}_{t\notin\mathsf{Open}(\mathsf{chal})})$
  - Return C

To prove that extraction succeeds, we will rely on the security of cut-and-choose.

Claim 7.5. Define the following set, where $i \in [0, D-1]$ and $j \in [B]$:

$$\mathsf{Good} := \left\{ \begin{array}{r} h = \{h_{i,j}\}_{i,j} \text{ where } h_{i,j} \in \{0,1\}^\lambda \\ (c, \overline{h}, \overline{k}, r, t) \text{ s.t. for } i \in [0, D-1] \text{ and } j \in [B] \ : \ k = \{k_{i,j}\}_{i,j} \text{ where } k_{i,j} = \mathsf{QPrO}_t(\mathsf{Gen}, h_{i,j}) \\ c = \mathsf{com}(\overline{k}; r) \end{array} \right\}$$

For all QPT adversaries $\mathcal{A}$, for large enough λ

$$\Pr\left[\begin{array}{l} (|\mathbb{B}| \geq (\lambda - |\mathsf{Open}(\mathsf{chal})|)/2) \ \wedge \\ \forall t \in \mathsf{Open}(\mathsf{chal}), \\ (c_t, \overline{h}_t, \overline{k}_t, r_t, t) \in \mathsf{Good} \end{array} \middle| \begin{array}{r} h^* \leftarrow \{0,1\}^\lambda \\ \{c_t, \overline{h}_t, \overline{k}_t, r_t\}_{t\in[\lambda]} \leftarrow \mathcal{A}^{\mathsf{QPrO}}(h^*) \\ \mathsf{chal} := \mathsf{QPrO}_0(\mathsf{Eval}, h^*, (c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda)) \\ \mathbb{B} := \{t : (c_t, \overline{h}_t, \overline{k}_t, r_t, t) \notin \mathsf{Good} \wedge c_t = \mathsf{com}(\overline{k}_t; r_t)\} \end{array}\right] \leq \mathsf{negl}(\lambda)$$

Proof. We prove by reduction to the security of post-quantum Fiat Shamir for statistically sound protocols. Let Π be the three round protocol that consists of $\mathcal{A}$ sending $\{c_t, \overline{h}_t\}_{t\in[\lambda]}$, receiving a random string chal, and sending $\{\overline{k}_t, r_t\}_{t\in[\lambda]}$. The adversary succeeds in breaking soundness if for $\mathbb{B} := \{t : (c_t, \overline{h}_t, \overline{k}_t, r_t, t) \notin \mathsf{Good} \wedge c_t = \mathsf{com}(\overline{k}_t; r_t)\}$, $|\mathbb{B}| \geq (\lambda - |\mathsf{Open}(\mathsf{chal})|)/2$ and $\forall t \in \mathsf{Open}(\mathsf{chal}), (c_t, \overline{h}_t, \overline{k}_t, r_t, t) \in \mathsf{Good}$. For any first message by the adversary, let the set $\mathbb{G} := \{t : \exists \overline{k}, r \text{ s.t. } (c_t, \overline{h}_t, \overline{k}, r, t) \in \mathsf{Good}\}$. By the perfect binding of the commitment scheme, $\mathbb{G} \cap \mathbb{B} = \emptyset$. Therefore the adversary can only break soundness if $|\mathbb{G}| \leq \lambda - (\lambda - |\mathsf{Open}(\mathsf{chal})|)/2 = (\lambda + |\mathsf{Open}(\mathsf{chal})|)/2$ since otherwise $|\mathbb{B}|$ will be too small. Additionally if $\mathsf{Open}(\mathsf{chal}) \not\subseteq \mathbb{G}$ then $\mathcal{A}$ cannot win, since if $t \notin \mathbb{G}$ then it must be the case that $(c_t, \overline{h}_t, \overline{k}_t, r_t, t) \notin \mathsf{Good}$. Therefore, whenever $\mathcal{A}$ succeeds at breaking soundness it must be the case that

- $\mathsf{Open}(\mathsf{chal}) \not\subseteq \mathbb{G}$ and
- $|\mathsf{Open}(\mathsf{chal})| \geq 2|\mathbb{G}| - \lambda$.

Note also that the first message fixes $\mathbb{G}$ and chal is chosen randomly independently of $\mathbb{G}$. Therefore

$$\Pr[\mathsf{Open}(\mathsf{chal}) \not\subseteq \mathbb{G}] \leq 1/2^{\lambda - |\mathbb{G}|}$$

and by Hoeffding inequality, if $|\mathbb{G}| \geq 3\lambda/4$,

$$\Pr[|\mathsf{Open}(\mathsf{chal})| \geq 2|\mathbb{G}| - \lambda] \leq \exp\left(-(4|\mathbb{G}| - 3\lambda)^2/\lambda\right)$$

If $|\mathbb{G}| \leq 3\lambda/4 + \lambda/4$ then $1/2^{\lambda - |\mathbb{G}|} \leq 1/2^{\lambda/8} \leq \mathsf{negl}(\lambda)$ while if $|\mathbb{G}| \geq 3\lambda/4 + \lambda/4$ then $\exp\left(-(4|\mathbb{G}| - 3\lambda)^2/\lambda\right) \leq \exp(-\lambda/4) \leq \mathsf{negl}(\lambda)$, so the adversary can break soundness with at most negligible probability.

Let $\mathsf{QPrO}' := (\mathsf{QPrO}_1, \mathsf{QPrO}_2, \ldots, \mathsf{QPrO}_\lambda)$, i.e. all but the first instantiation of QPrO. Applying the Fiat-Shamir transform [Unr17] to the protocol above yields for all adversaries $\mathcal{A}$ that make at most $\mathsf{poly}(q)$ queries to $\mathcal{O}$, for large enough λ

$$\Pr\left[\begin{array}{l} (|\mathbb{B}| \geq \lambda/4) \ \wedge \\ \forall t \in \mathsf{Open}(\mathsf{chal}), \\ (c_t, \overline{h}_t, \overline{k}_t, r_t, t) \in \mathsf{Good} \end{array} \middle| \begin{array}{r} \{c_t, \overline{h}_t, \overline{k}_t, r_t\}_{t\in[\lambda]} \leftarrow \mathcal{A}^{\mathcal{O}, \mathsf{QPrO}'} \\ \mathsf{chal} := \mathcal{O}(c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda)) \\ \mathbb{B} := \{t : (c_t, \overline{h}_t, \overline{k}_t, r_t, t) \notin \mathsf{Good} \wedge c_t = \mathsf{com}(\overline{k}_t; r_t)\} \end{array}\right] \leq \mathsf{negl}(\lambda)$$

where $\mathcal{O}$ is a random oracle.

Now, by Lemma 8.4 we know that for all QPT adversaries $\mathcal{A}$,

$$\left|\Pr\left[\mathcal{A}^{\mathsf{QPrO}_0}(h^*) = 1 : h \leftarrow \{0,1\}^\lambda\right] - \Pr\left[\mathcal{A}^{\mathsf{QPrO}_0[h^* \to k]}(h^*) = 1 : h^*, k \leftarrow \{0,1\}^\lambda\right]\right| = \mathsf{negl}(\lambda),$$

where $\mathsf{QPrO}_0[h^* \to k] = \mathsf{QPrO}_0$ except that on input (Eval, h^*, x) it outputs $f_k(x)$ instead of $f_{\pi^{-1}(h^*)}(x)$. That is, it answers PRF queries on handle h^* using an independently sampled PRF key. Additionally, by the post-quantum security of the PRF (and the fact that for any h^* and k, the oracle $\mathsf{QPrO}_0[h^* \to k]$ can be simulated given h^*, oracle access to f_k, and a post-quantum secure PRP)

$$\left|\Pr\left[\mathcal{A}^{\mathsf{QPrO}_0[h^*\to k]}(h^*) = 1 : h^*, k \leftarrow \{0,1\}^\lambda\right] - \Pr\left[\mathcal{A}^{\mathsf{QPrO}_0[h^*]}(h^*) = 1 : h \leftarrow \{0,1\}^\lambda\right]\right| = \mathsf{negl}(\lambda),$$

where $\mathsf{QPrO}_0[h^*] = \mathsf{QPrO}_0$ except that on input (Eval, h, x) it outputs $\mathcal{O}(x)$ instead of $f_{\pi^{-1}(h)}(x)$ for a random oracle $\mathcal{O}$. Let $\mathsf{QPrO}[h^*] = (\mathsf{QPrO}_0[h^*], \mathsf{QPrO}')$. Now, since for any h^* and k, the oracle $\mathsf{QPrO}_0[h^*]$ can be simulated given h^*, oracle access to $\mathcal{O}$, and a post-quantum secure PRP, we can replace access to $(\mathcal{O}, \mathsf{QPrO}')$ with $\mathsf{QPrO}[h^*]$ for random h^* in the post Fiat-Shamir protocol without any loss in soundness. That is, for all QPT adversaries $\mathcal{A}$, for large enough λ

$$\Pr\left[\begin{array}{l|r} (|\mathbb{B}| \geq \lambda/4) \wedge & h^* \leftarrow \{0,1\}^\lambda \\ \forall t \in \mathsf{Open}(\mathsf{chal}), & \{c_t, \overline{h}_t, \overline{k}_t, r_t\}_{t\in[\lambda]} \leftarrow \mathcal{A}^{\mathsf{QPrO}[h^*]} \\ (c_t, \overline{h}_t, \overline{k}_t, r_t) \in \mathsf{Good} & \mathsf{chal} := \mathsf{QPrO}_0[h^*](\mathsf{Eval}, h^*, c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda)) \\ & \mathbb{B} := \{t : (c_t, \overline{h}_t, \overline{k}_t, r_t) \notin \mathsf{Good} \wedge c_t = \mathsf{com}(\overline{k}_t; r_t)\} \end{array}\right] \leq \mathsf{negl}(\lambda)$$

Since no efficient adversary can distinguish $\mathsf{QPrO}_0[h^*]$ from QPrO_0 and QPrO' is efficiently simulatable, no efficient adversary can distinguish $\mathsf{QPrO}[h^*]$ from QPrO. Using this fact as well as by noting that the condition on the LHS is efficiently checkable, we obtain that for all QPT adversaries $\mathcal{A}$, for large enough λ

$$\Pr\left[\begin{array}{l|r} (|\mathbb{B}| \geq \lambda/4) \wedge & h^* \leftarrow \{0,1\}^\lambda \\ \forall t \in \mathsf{Open}(\mathsf{chal}), & \{c_t, \overline{h}_t, \overline{k}_t, r_t\}_{t\in[\lambda]} \leftarrow \mathcal{A}^{\mathsf{QPrO}}(h^*) \\ (c_t, \overline{h}_t, \overline{k}_t, r_t) \in \mathsf{Good} & \mathsf{chal} := \mathsf{QPrO}_0(\mathsf{Eval}, h^*, (c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda)) \\ & \mathbb{B} := \{t : (c_t, \overline{h}_t, \overline{k}_t, r_t) \notin \mathsf{Good} \wedge c_t = \mathsf{com}(\overline{k}_t; r_t)\} \end{array}\right] \leq \mathsf{negl}(\lambda)$$

which is the statement in the claim. □

Note that since

$$\{\mathsf{crs} : \mathsf{crs} \leftarrow \mathsf{NIZK.Setup}(1^\lambda)\} \approx \{\mathsf{crs} : (\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{NIZK.Ext}_0(1^\lambda)\}$$

and since $\mathsf{Ext}_0(1^\lambda)$ samples h^* honestly,

$$\{\mathsf{pp} : \mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda)\} \approx \{\mathsf{pp} : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda)\}$$

Suppose there exists a QPT adversary $\mathcal{A}$ such that

$$\Pr\left[\begin{array}{l:r} \left(\mathsf{Ver}(\mathsf{pp}, \varphi, \widetilde{C}, \pi) = \top\right) \wedge & (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ \left(\exists x \text{ s.t } C(x) \neq \mathsf{Eval}(\widetilde{C}, x) \vee \varphi(C) = 1\right) & (\widetilde{C}, \pi) \leftarrow \mathcal{A}^{\mathsf{QPrO}}(\mathsf{pp}) \\ & C \leftarrow \mathsf{Ext}_1(\mathsf{pp}, \mathsf{td}, \varphi, \widetilde{C}, \pi) \end{array}\right] \geq \epsilon(\lambda)$$

for some non-negligible function $\epsilon(\cdot)$, then since Ver includes verifying the underlying NIZK, by the knowledge soundness of the underlying NIZK it must be the case that

$$\Pr\left[\begin{array}{l:r} \left(\mathsf{Ver}(\mathsf{pp}, \varphi, \widetilde{C}, \pi) = \top\right) \wedge & (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ \left(\exists x \text{ s.t } C(x) \neq \mathsf{Eval}(\widetilde{C}, x) \vee \varphi(C) = 1\right) \wedge & (\widetilde{C}, \pi) \leftarrow \mathcal{A}^{\mathsf{QPrO}}(\mathsf{pp}) \\ (\mathcal{R}(x, w) = 1) & C \leftarrow \mathsf{Ext}_1(\mathsf{pp}, \mathsf{td}, \varphi, \widetilde{C}, \pi) \end{array}\right] \geq \epsilon(\lambda) - \mathsf{negl}(\lambda)$$

where $\mathcal{R}$ is the relation defined by Prove while x and w are the instance and witness extracted by $\mathsf{NIZK.Ext}_1$ run internally by Ext_1.

Now if $\mathcal{R}(x, w) = 1$ then it must be the case that

- $\varphi(C) = 1$ and
- for all $t \notin \mathsf{Open}(\mathsf{chal})$
  - $c_t = \mathsf{com}(\overline{k}_t; r_t)$ and
  - $\widetilde{C}_t = \mathsf{JLLWObf}(1^\lambda, C, \overline{k}_t, \overline{h}_t; \widetilde{r}_t)$

Additionally if $\mathsf{Ver}(\mathsf{pp}, \varphi, \widetilde{C}, \pi) = \top$ then it must be the case that $\mathsf{chal} = \mathsf{QPrO}(\mathsf{Eval}, h^*, (c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda))$ and for all $t \in \mathsf{Open}(\mathsf{chal})$, $(c_t, \overline{h}_t, \overline{k}_t, r_t) \in \mathsf{Good}$. This means that for all $t \notin \mathsf{Open}(\mathsf{chal})$, Obf_t is an honestly computed obfuscation of C using handles $\overline{h}_t$ and keys $\overline{k}_t$, and c_t is an honestly computed commitments to $\overline{k}_t$. By the perfect correctness of JLLWObf, if for all $k_{i,j} \in \overline{k}_t$ and $h_{i,j} \in \overline{h}_t$ it was the case that $\mathsf{QPrO}(\mathsf{Gen}, k_{i,j}) = h_{i,j}$ then for all x, $\mathsf{JLLWObf.Eval}(\widetilde{C}_t, x) = C(x)$. However, since $\mathsf{Eval}(\widetilde{C}, x)$ computes the most frequent element of $\{\mathsf{JLLWObf.Eval}(\widetilde{C}_t, x)\}_{t \notin \mathsf{Open}(\mathsf{chal})}$, then if there exists x such that $\mathsf{Eval}(\widetilde{C}, x) \neq C(x)$, it must mean that for the majority of $t \notin \mathsf{Open}(\mathsf{chal})$, the keys in $\overline{k}_t$ do not all map to their corresponding handle in $\overline{h}_t$ under $\mathsf{QPrO}(\mathsf{Gen}, \cdot)$. This means that the set $\mathbb{B} := \{t : (c_t, \overline{h}_t, \overline{k}_t, r_t) \notin \mathsf{Good} \wedge c_t = \mathsf{com}(\overline{k}_t; r_t)\}$ is of size atleast $\lambda - |\mathsf{Open}(\mathsf{chal})|$.

Let $\widetilde{\mathcal{A}}$ be the algorithm that receives h^*, samples $\mathsf{crs}, \mathsf{td} \leftarrow \mathsf{NIZK.Ext}_0(1^\lambda)$, runs $\mathcal{A}((\mathsf{crs}, h^*))$ to obtain $(\widetilde{C}, \pi)$, and runs $\mathsf{Ext}_1(\mathsf{pp}, \mathsf{td}, \varphi, \widetilde{C}, \pi)$ to obtain the set $\{c_t, \overline{h}_t, \overline{k}_t, r_t\}_{t \in [\lambda]}$, which $\widetilde{\mathcal{A}}$ then outputs. Then by the above

$$\Pr\left[\begin{array}{l|r} (|\mathbb{B}| \geq (\lambda - |\mathsf{Open}(\mathsf{chal})|)/2) \wedge & h^* \leftarrow \{0,1\}^\lambda \\ \forall t \in \mathsf{Open}(\mathsf{chal}), & \{c_t, \overline{h}_t, \overline{k}_t, r_t\}_{t \in [\lambda]} \leftarrow \widetilde{\mathcal{A}}^{\mathsf{QPrO}}(h^*) \\ (c_t, \overline{h}_t, \overline{k}_t, r_t) \in \mathsf{Good} & \mathsf{chal} := \mathsf{QPrO}_0(\mathsf{Eval}, h^*, (c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda)) \\ & \mathbb{B} := \{t : (c_t, \overline{h}_t, \overline{k}_t, r_t) \notin \mathsf{Good} \wedge c_t = \mathsf{com}(\overline{k}_t; r_t)\} \end{array}\right] \geq \epsilon(\lambda) - \mathsf{negl}(\lambda)$$

which contradicts Claim 7.5.

Simulation Security. We define SimGen and SimObf as follows.

$\mathsf{SimGen}^{\mathsf{QPrO}}(1^\lambda)$:

- $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{NIZK.SimGen}(1^\lambda)$
- $h^* \leftarrow \{0,1\}^\lambda$
- Return $((\mathsf{crs}, h^*), \mathsf{td})$

$\mathsf{SimObf}^{\mathsf{QPrO}}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C)$:

- Run $\mathsf{Obf}^{\mathsf{QPrO}}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \varphi, C)$ honestly, except generate π by running $\mathsf{NIZK.Sim}(\mathsf{crs}, \mathsf{td}, x)$.

Honest-vs-simulated indistinguishability follows from the zero-knowledge property of NIZK. We will next show that SimObf satisfies subexponential ideal obfuscation security.

We show security by building a simulator which "takes over" all the oracles $(\mathsf{QPrO}_0, \mathsf{QPrO}_1, \ldots, \mathsf{QPrO}_\lambda)$. Let $S = (S_1, S_2, S_3)$, where S_1 will compute $(\mathsf{pp}, \mathsf{td})$, S_2 will compute the obfuscated circuit $\widetilde{C}$, and

S_3 will simulate the oracles after the obfuscated circuit is sent to the adversary. Formally, we show for every C

$$\left| \Pr\left[A^{\mathsf{QPrO}}(\mathsf{pp}, \widetilde{C}) = 1 : \begin{array}{l} (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{SimGen}^{\mathsf{QPrO}}(1^\lambda) \\ \widetilde{C} \leftarrow \mathsf{SimObf}^{\mathsf{QPrO}}(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi, C) \end{array} \right] \right.$$
$$\left. - \Pr\left[A^{S_3^C}(\mathsf{pp}, \widetilde{C}) = 1 : \begin{array}{l} (\mathsf{pp}, \mathsf{td}) \leftarrow S_1(1^\lambda) \\ \widetilde{C} \leftarrow S_2^C(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi) \end{array} \right] \right| \leq 1/2^{\lambda^\epsilon_{\mathsf{OBF}}}$$

for some $\epsilon > 0$. For any function f and handle h, left $\mathsf{QPrO}[h^* \mapsto f]$ refer to an oracle identical to QPrO except Eval queries with handle h are answered using f instead. Define $S = (S_1, S_2, S_3)$, where all simulators share state, as follows.

$S_1(1^\lambda)$:

- $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{NIZK.SimGen}(1^\lambda)$
- $h^* \leftarrow \{0,1\}^\lambda$
- Return $((\mathsf{crs}, h^*), \mathsf{td})$

$S_2^C(1^{\lambda_{\mathsf{OBF}}}, \mathsf{pp}, \mathsf{td}, \varphi)$:

- Let $(S_{1,i}, S_{2,i}, S_{3,i})_{i\in[\lambda]}$ be λ separate instantiations of the ideal obfuscation simulators for $\mathsf{JLLWObf}$, where $(S_{1,i}, S_{2,i}, S_{3,i})$ are allowed to control QPrO_i.
- Compute $\widetilde{C}$ as in SimObf except:
  - Sample $\mathsf{chal} \leftarrow \{0,1\}^\lambda$
  - $\forall t \in [\lambda]$, set $\widetilde{C}_t \leftarrow S_{2,t}^C(1^{\lambda_{\mathsf{OBF}}}, D, S)$.
  - If $t \in \mathsf{Open}(\mathsf{chal})$, $\widetilde{C}_t$ is not sent to the adversary.
  - For all $t \in \mathsf{Open}(\mathsf{chal})$:
    - $c_t \leftarrow \mathsf{com}(0; r_t)$
    - $\overline{h}_t$ is extracted from $\widetilde{C}_t$

S_3^C:

- Let $\mathcal{O}$ be an efficient simulation of a random oracle. Note that such simulation is possible via the compressed oracle technique [Zha19].
- Let $\mathcal{O}'$ be identical to $\mathcal{O}$ except $(c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda)$ maps to chal.
- Let $\widetilde{\mathsf{QPrO}_0}$ be a simulated instance of QPrO_0 (using a PRP).
- Give query access to $(\widetilde{\mathsf{QPrO}_0}[h^* \mapsto \mathcal{O}'], S_{3,1}^C, \ldots, S_{3,\lambda}^C)$.

We prove indistinguishability via a sequence of hybrids.

- Hybrid_0: Identical to using $(\mathsf{SimGen}, \mathsf{SimObf})$.

- Hybrid$_1$: Identical to Hybrid$_0$, except A and SimObf are given query access to $(\mathsf{QPrO}_0[h^* \mapsto \mathcal{O}], \mathsf{QPrO}')$ instead of QPrO, where $\mathsf{QPrO}' := (\mathsf{QPrO}_1, \ldots, \mathsf{QPrO}_\lambda)$. By Lemma 8.4 and the post-quantum security of the PRF, as well as the observation that $\mathsf{QPrO}_0[h^* \mapsto f]$ can be simulated efficiently using query access to f, both hybrids are indistinguishable.
- Hybrid$_2$: Identical to Hybrid$_1$, except $\mathsf{chal} \leftarrow \{0,1\}^\lambda$ and A and SimObf are given query access to $(\mathsf{QPrO}_0[h^* \mapsto \mathcal{O}'], \mathsf{QPrO}')$, where $\mathcal{O}'$ is identical to $\mathcal{O}$ except $(c_1, \ldots, c_\lambda, \overline{h}_1, \ldots, \overline{h}_\lambda)$ maps to chal. Both hybrids are identical in the adversary's view.
- Hybrid$_3$: Identical to Hybrid$_2$, except for all $t \in \mathsf{Open}(\mathsf{chal})$, c_t is computed as a commitment to 0. Both hybrids are indistinguishable by the hiding property of the commitment scheme.
- Hybrid$_{4,i}$: For all $i \in [0, \lambda]$, Hybrid$_{4,i}$ is identical to Hybrid$_3$ except for all $j \leq i$:
  - $\widetilde{C}_j \leftarrow S^C_{2,j}(1^{\lambda_{\mathsf{OBF}}}, 1^D, 1^S)$
  - If $j \notin \mathsf{Open}(\mathsf{chal})$ then $\overline{h}_j$ is extracted from $\widetilde{C}_j$

  and A is given access to $(\mathsf{QPrO}_0[h^* \mapsto \mathcal{O}'], S^C_{3,1}, \ldots, S^C_{3,i}, \mathsf{QPrO}_{i+1}, \ldots, \mathsf{QPrO}_\lambda)$. Hybrid$_{4,0}$ is identical to Hybrid$_3$, and for all i, Hybrid$_{4,i-1}$ and Hybrid$_{4,i}$ are indistinguishable by the relativizing ideal obfuscation security of JLLW.
- Hybrid$_4$: Identical to Hybrid$_{4,\lambda}$ except $\widetilde{\mathsf{QPrO}_0}[h^* \mapsto \mathcal{O}']$ is used instead of $\mathsf{QPrO}_0[h^* \mapsto \mathcal{O}']$. Hybrids$_{u,\lambda}$ and Hybrid$_4$ are indistinguishable by PRP security.

The view of the adversary in Hybrid$_4$ is identical to the view when running with S. Since all primitives are subexponentially secure, the adversary has at most subexponential distinguishing advantage in λ_{OBF}.

Finally, by Theorem 7.2, evasive composability and indistinguishability obfuscation are implied by ideal obfuscation. □

We obtain the following as a direct corollary of the above theorem and Theorem 8.1.

Theorem 7.6. *Assuming functional encryption with subexponential security (Theorem 3.15) and post-quantum NIZK arguments of knowledge for NP with a URS setup, there exists a provable obfuscation scheme (Theorem 7.1) with a URS setup in the QPrO model (Theorem 3.23).*

8 Security of the JLLW Obfuscator in the QPrO Model

This section is dedicated to proving that the JLLW construction of ideal obfuscation in the pseudo-random oracle model is post-quantum secure.

Theorem 8.1. *Let $\mathcal{R}$ be an oracle. Assuming functional encryption and pseudorandom functions with subexponential security relative to $\mathcal{R}$, the JLLW obfuscation given in Theorem 3.25 satisfies (subexponential) post-quantum ideal obfuscation (Theorem 3.24) relative to $\mathcal{R}$ in the quantum-accessible pseudorandom oracle model (Theorem 3.23).*

There are two main differences between the post-quantum setting and the classical setting that arise here. The first, and biggest, difference is the difficulty of adaptively programming random oracle queries. This nuance prevents the JLLW trick of switching behavior on an exponential number

of inputs by adaptively reprogramming the oracle at only a polynomial number of queries. As such, our post-quantum result relies on the subexponential security of the underlying primitives, in contrast to JLLW's reliance on only polynomial hardness.

A second difference is the treatment of random permutations in the quantum setting. It is currently an open problem to perform efficient (statistical) simulation of a random permutation.[4] In theory, this means that oracle access to a random permutation could break computational security assumptions. However, this cannot occur if post-quantum *psuedo*-random permutations exist. A PRP would allow efficient simulation of the random permutation against any computationally efficient adversary. Fortunately, both (post-quantum) psuedorandom functions and functional encryption imply the existence of (post-quantum) PRPs [Zha25], so we do not need any additional assumptions.

Building on the observation that a QPrO can be efficiently simulated using post-quantum PRFs and PRPs, one can see that the parallel composition of JLLW is an ideal obfuscator when multiple QPRO oracles are available.[5]

Corollary 8.2. Assuming functional encryption and pseudorandom functions with subexponential security relative to $\mathcal{R}$, the parallel composition

$$\mathsf{Obf}(C_1, \ldots, C_n) = (\mathsf{Obf}(C_1), \ldots, \mathsf{Obf}(C_n))$$

of the JLLW obfuscation satisfies post-quantum ideal obfuscation (Theorem 3.24) in the n-time quantum-accessible pseudorandom oracle model (Theorem 3.23).

We also obtain the following as an immediate corollary, which allows us conclude that several objects from the literature that were previously only known in the classical oracle model can in fact be constructed in the QPrO model, assuming an appropriate flavor of functional encryption (e.g. witness encryption for QMA [BM22], copy-protection for all unlearable functionalities [ALL+21], obfuscation for various classes of quantum circuits [BKNY23, BBV24, HT25], and quantum fire [ÇGS25]).

Corollary 8.3. Assuming functional encryption as defined in Theorem 3.15, there exists post-quantum ideal obfuscation in the quantum-accessible pseudorandom oracle model.

8.1 QPrO Key Reprogramming

Arguing that the QPrO behaves as a random function on a random handle h requires first removing the key k that is associated with the handle h by the random permutation. To help with this step, we show the following lemma.

Lemma 8.4. Let $F = \{f_k\}_k$ be a pseudorandom function and QPrO a pseudorandom oracle for F. Then for any QPT adversary $\mathcal{A}$,

$$\left| \Pr\left[\mathcal{A}^{\mathsf{QPrO}}(h) = 1 : h \leftarrow \{0,1\}^\lambda\right] - \Pr\left[\mathcal{A}^{\mathsf{QPrO}[h \to k]}(h) = 1 : h, k \leftarrow \{0,1\}^\lambda\right] \right| = \mathsf{negl}(\lambda),$$

[4] The analogous question for random functions is solved by the influential compressed oracle technique [Zha19].

[5] Security can also be shown with a single QPrO by observing that each simulator really only needs to program a handful of key-handle pairs and the other handles can be answered relatively to the original QPrO. Proving this would further complicate an already technically involved proof for a small improvement in the model, so we omit the details.

where $\mathsf{QPrO}[h \to k] = \mathsf{QPrO}$ except that on input (Eval, h, x) it outputs $f_k(x)$ instead of $f_{\pi^{-1}(h)}(x)$. That is, it answers PRF queries on handle h using an independently sampled PRF key.

If F is subexponentially secure, then the probability of distinguishing is subexponential.

Proof. We consider a sequence of three hybrids, defined as follows. In each, begin by sampling a uniformly random permutation $\pi : \{0,1\}^\lambda \to \{0,1\}^\lambda$, and $k, k' \leftarrow \{0,1\}^\lambda$. Define $h = \pi(k)$ and $h' = \pi(k')$. For all $k'' \notin \{k, k'\}$, define $\mathsf{QPrO}(\mathsf{Gen}, k'') = \pi(k'')$. For all $h'' \notin \{h, h'\}$, define $\mathsf{QPrO}(\mathsf{Eval}, h'', \cdot) = f_{\pi^{-1}(h'')}(\cdot)$. Define $\mathsf{QPrO}(\mathsf{Eval}, h, \cdot) = f_k(\cdot)$. The adversary is initialized with h, given access to the QPrO oracle and outputs a bit, where the behavior of QPrO on the remaining inputs differs in each hybrid, as follows.

- $\mathcal{H}_0$: $\mathsf{QPrO}(\mathsf{Gen}, k) = h$, $\mathsf{QPrO}(\mathsf{Gen}, k') = h'$, $\mathsf{QPrO}(\mathsf{Eval}, h', \cdot) = f_{k'}(\cdot)$.
- $\mathcal{H}_1$: $\mathsf{QPrO}(\mathsf{Gen}, k) = h$, $\mathsf{QPrO}(\mathsf{Gen}, k') = h$, $\mathsf{QPrO}(\mathsf{Eval}, h', \cdot) = f_k(\cdot)$.
- $\mathcal{H}_2$: $\mathsf{QPrO}(\mathsf{Gen}, k) = h'$, $\mathsf{QPrO}(\mathsf{Gen}, k') = h$, $\mathsf{QPrO}(\mathsf{Eval}, h', \cdot) = f_k(\cdot)$.

Observe that $\mathcal{H}_0$ is distributed exactly as the LHS of the lemma statement, while $\mathcal{H}_2$ is distributed exactly as the RHS of the lemma statement, and thus it suffices to show indistinguishability between the hybrids.

- $\mathcal{H}_0 \approx \mathcal{H}_1$: By Theorem 3.5, it suffices to show that the adversary can output a string in $\{k', h'\}$ with only negligible probability in $\mathcal{H}_0$. This is straightforward to see due to the fact that k' is sampled independently of the rest of the experiment (including the string h that the adversary is initialized with) and $h' = \pi(k')$.
- $\mathcal{H}_1 \approx \mathcal{H}_2$: By Theorem 3.5, it suffices to show that the adversary can output k with only negligible probability in $\mathcal{H}_2$. Note that $\mathcal{H}_2$ can be simulated by a reduction given just oracle access to f_k using a post-quantum psuedorandom permutation (which is implied by post-quantum PRFs). Thus, this follows from a standard claim that, given oracle access to a PRF f_k, an adversary can output k with only negligible probability.

Assuming subexponential security of the underlying PRF and PRP, and observing that the key/handle spaces are exponentially sized, it is clear that the hybrids are subexponentially close. □

8.2 Proof of Security

Proof of Theorem 8.1. Correctness was shown in [JLLW23]. We show security against quantum adversaries here. We remark that the security of the construction can be tuned to any subexponential function by analogously tuning the subexponential parameters used in the subclaims of the proof.

We will consider a main series of hybrid experiments $\mathsf{Hyb}_{i,\$\$}$ which are preceded by two hybrids Hyb and $\mathsf{Hyb}_{\mathsf{PRP}}$. We show that

$$\mathsf{Hyb}_{\mathsf{real}} \approx \mathsf{Hyb}_{\mathsf{KH}} \approx \mathsf{Hyb}_{\mathsf{PRP}} \approx \mathsf{Hyb}^x_{0,\$\$} \approx \ldots \approx \mathsf{Hyb}^x_{D,\$\$} \approx \mathsf{Sim}.$$

The two initial hybrids make the following changes from the prior hybrid.

- Hyb_{KH}: Instead of generating the key-handle pairs $(k_{d,b}, h_{d,b})$ used for the obfuscation by sampling $k_{d,b} \leftarrow \{0,1\}^\lambda$ and setting $h_{d,b} = \pi(k_{d,b})$, independently sample $k_{d,b}, h_{d,b} \leftarrow \{0,1\}^\lambda$.

- $\mathsf{Hyb}_{\mathsf{PRP}}$: The random permutation π is replaced by a *pseudorandom* permutation $\pi_{k_{\mathsf{PRP}}}$. Note that $\pi_{k_{\mathsf{PRP}}}$ is not used for relating $(k_{d,b}, h_{d,b})$.

The intermediate hybrids $\mathsf{Hyb}_{\delta,\$\$}$ are specified in Figure 1 and Figure 2. At a high level, each $\mathsf{Hyb}_{\delta,\$\$}$ makes the following differences from $\mathsf{Hyb}_{\mathsf{PRP}}$:

- **Depth $d < \delta$:** The ciphertext ct_χ for input prefix $\chi \in \{0,1\}^d$ is in *simulation* mode and uses truly random r_χ in its encryption. Furthermore, every handle $h_{d,b}$ at this depth does not have a corresponding PRF key (i.e. $k_{d,b}$ is non-existent/unused).
- **Depth $d \geq \delta$:** The ciphertext ct_χ is in *normal* mode and uses pseudorandom r_χ expanded from $s_{\chi \leq \delta}$. The only exception is $d = \delta$, where r_χ is truly random. Additionally, every handle $h_{d,b}$ at this depth corresponds to some PRF key $k_{d,b}$.

Sim is identical to $\mathsf{Hyb}_{D,\$\$}$ except that it changes the final ciphertext ct_x corresponding to the full input $x \in \{0,1\}^D$ to an encryption of

$$(\mathsf{flag}_x, x, \mathsf{info}_x) := \begin{cases} (\mathsf{normal}, x, (C, s_x)) & \text{in } \mathsf{Hyb}_{D,\$\$} \\ (\mathsf{sim}, x, C(x)) & \text{in } \mathsf{Sim} \end{cases}$$

Each hybrid is expressed in terms of three oracles: Sim_1 corresponds to the QPrO oracle before obfuscation, Sim_2 corresponds to the obfuscation step, and Sim_3 corresponds to the QPrO oracle after obfuscation.

We begin by showing that the first two transitions and the last transition are indistinguishable, since these transitions are easier to see.

Claim 8.5.

$$\mathsf{Hyb}_{\mathsf{real}} \approx \mathsf{Hyb}_{\mathsf{KH}}$$

Proof. The only difference between these hybrids is when either $\mathsf{Sim}_1^{0,\$\$}$ or $\mathsf{Sim}_3^{0,\$\$}$, corresponding to the QPrO oracle, receive an Eval query on $(h_{d,b}, x)$ for some $h_{d,b} \in \mathsf{Handles}$. In this case, $\mathsf{Hyb}_{0,\$\$}$ uses an independently random $k_{d,b}$ to answer the query instead of $\pi^{-1}(h)$. Since d ranges over $0, \ldots, D$ and b ranges over $1, \ldots, B$, the claim follows by $(D+1) \cdot B = \mathsf{poly}(\lambda)$ applications of Theorem 8.4. □

Claim 8.6. Assuming post-quantum pseudorandom permutations relative to $\mathcal{R}$,

$$\mathsf{Hyb}_{\mathsf{KH}} \approx \mathsf{Hyb}_{\mathsf{PRP}}$$

Proof. This follows immediately from the post-quantum security of the pseudorandom permutation. □

Claim 8.7. If FE is 2^D-adaptively secure relative to $\mathcal{R}$ then

$$\mathsf{Hyb}_{D,\$\$} \approx \mathsf{Sim}$$

Shared State:

$\pi_{k_{\mathsf{PRP}}}$	The pseudorandom permutation used to map keys onto handles.
$\mathcal{C}$	The circuit being obfuscated. This is used in Sim_2 and Sim_3.
$\mathsf{Handles} := \{h_{d,b} \leftarrow \{0,1\}^\lambda\}_{\substack{0\le d\le D,\\ 1\le b\le B}}$	The handles of the QPrO in the obfuscation. $h_{d,b}$ corresponds to level d and block b. Generated uniformly at random.
$\mathsf{Keys} := \{k_{d,b} \leftarrow \{0,1\}^\lambda\}_{\substack{\delta\le i\le D\\ 1\le b\le B}}$	Keys of H used for the obfuscation. Generated uniformly at random.
$\{(\mathsf{pk}_d, \mathsf{sk}_d)\}_{0\le d\le D}$	Public and secret keys for each level d. Generated as in Theorem 3.25.
$F_{b,d} : \{0,1\}^D \to \{0,1\}^L$ For $0 \le d \le \delta$ and $1 \le b \le B$	Random functions for the non-programmed portion of evaluation queries.
$F_\sigma : \{0,1\}^{<\delta} \times \{1,\dots,B\} \to \{0,1\}^\lambda$	A random function used to generate one-time pads for the ciphertexts at depth $d < D$ and block $b \in \{1,\dots,B\}$.
$F_r : \{0,1\}^{\le\delta} \to \{0,1\}^\lambda$	A random function used to generate randomness r_χ for ciphertext ct_χ for input prefix $\chi \in \{0,1\}^{\le D}$.
$F_s : \{0,1\}^{\delta} \to \{0,1\}^\lambda$	Used to define r_χ and s_χ for $\lvert\chi\rvert = \delta$.
$\mathsf{ct}_\chi = \mathsf{Enc}(\mathsf{pk}_{\lvert\chi\rvert}, \mathsf{flag}_\chi, \mathsf{info}_\chi; r_\chi)$	The ciphertext corresponding to input prefix $\chi \in \{0,1\}^{\le D}$. This is implicitly defined by the other parameters and is used in Sim_2 and Sim_3.
$\mathsf{flag}_\chi := \begin{cases} \mathsf{sim} & \text{if } \lvert\chi\rvert < \delta \\ \mathsf{normal} & \text{if } \lvert\chi\rvert \ge \delta \end{cases}$	Flag used to specify whether the ciphertext ct_χ is in simulation mode or normal mode.
For $\lvert\chi\rvert = \delta$: $r_\chi := F_r(\chi)$ $s_\chi := F_s(\chi)$ For $\lvert\chi\rvert > \delta$: $s_{\chi\|0}\|r_{\chi\|0}\|s_{\chi\|1}\|r_{\chi\|1} := G_{sr}(s_\chi)$	r_χ is the randomness used for the encryption of each χ at depth $\ge \delta$. s_χ is a PRG seed used to generate downstream r and s.
$\mathsf{info} := \begin{cases} \{F_\sigma(\chi,b)\}_{1\le b\le B} & \text{if } \lvert\chi\rvert < \delta \\ \left\{C, \{k_{d,b}\}_{\substack{\lvert\chi\rvert\le d\le D,,\\ 1\le b\le B}}, s_\chi\right\} & \text{if } \lvert\chi\rvert \ge \delta \end{cases}$	Information used to evaluate the ciphertext ct_χ under the FE.
$G_{sr} : \{0,1\}^\lambda \to \{0,1\}^{4\lambda}$ $G_v : \{0,1\}^\lambda \to \{0,1\}^L$	PRGs used to generate the encryption randomness for each ct_χ and for succinctly hard-coding the programmed behavior into the FE ciphertexts, respectively.

Figure 1: $\mathsf{Hyb}_{\delta,\$\$}$: Shared state of algorithms $\mathsf{Sim}_1^{(\delta,\$\$)}, \mathsf{Sim}_2^{(\delta,\$\$)}, \mathsf{Sim}_3^{(\delta,\$\$)}$. Differences from $\mathsf{Hyb}_{\mathsf{real}}$ and dependence on δ highlighted.

Initialization. Sample the following according to their distributions as specified in Figure 1:

$$\pi,\ \mathsf{Handles},\ \mathsf{Keys},\ \{F_{b,d}\}_{\substack{0\le d\le D,\\ 1\le b\le B}},\ F_\sigma,\ F_r$$

$\mathsf{Sim}_1^{\$\$,\delta}$:
On input (Gen, k):
1. Output $\pi(k)$.

On input (Eval, h, x):
1. If $h = h_{d,b}$ for some $h_{d,b} \in \mathsf{Handles}$, output $f_{k_{d,b}}(x)$.
2. Otherwise, compute $k \leftarrow \pi^{-1}(h)$ and output $f_k(x)$.

$\mathsf{Sim}_2^{\$\$,\delta}$:
1. Generate $\{\mathsf{pk}_d, \mathsf{sk}_d\}_{d=0,\ldots,D}$ as specified in Theorem 3.25.
2. Output $\widehat{C}^{\bullet}\left[\mathsf{ct}_\epsilon, \{\mathsf{sk}_d\}_{0\le d\le D}, \{h_{d,b}\}_{\substack{0\le d<D,\\ 1\le b\le B}}\right]$, as defined in Theorem 3.25.

Sim_3:
On input (Gen, k):
1. Output $\pi(k)$.

On input (Eval, h, x):
1. If $h = h_{d,b}$ for some $h_{d,b} \in \mathsf{Handles}$:
   (a) If $d < \delta$:
      i. If $x = \chi \| 0^{D-d}$ for $\chi \in \{0,1\}^d$:
         A. Output $G_v(F_\sigma(\chi, b)) \oplus \left[\mathsf{ct}_{\chi\|0} \| \mathsf{ct}_{\chi\|1}\right]_b$.
      ii. Otherwise output $F_{d,b}(x)$.
   (b) Otherwise output $f_{k_{d,b}}(x)$.
2. Otherwise compute $k \leftarrow \pi^{-1}(h)$ and output $f_k(x)$.

Figure 2: $\mathsf{Hyb}_{\delta,\$\$}$: Specification of algorithms $\mathsf{Sim}_1^{(\delta,\$\$)}, \mathsf{Sim}_2^{(\delta,\$\$)}, \mathsf{Sim}_3^{(\delta,\$\$)}$. Differences from $\mathsf{Hyb}_{\mathsf{real}}$ and dependence on δ highlighted.

Proof. The only difference between these hybrids is the plaintext encrypted under ct_x for *each* full input $x \in \{0,1\}^D$. Specifically, it is an encryption of

$$(\mathsf{flag}_x, x, \mathsf{info}_x) := \begin{cases} (\mathsf{normal}, x, (C, s_x)) & \text{in } \mathsf{Hyb}_{D,\$\$} \\ (\mathsf{sim}, x, C(x)) & \text{in } \mathsf{Sim} \end{cases}$$

In both hybrids, these ciphertexts are encrypted using true randomness under the public key pk_D. The secret key sk_D is for the function Eval. Observe that the result of Eval is the same for both plaintexts:

$$\begin{aligned} \mathsf{Eval}(\mathsf{normal}, x, (C, s_x)) &= \mathsf{Eval}_{\mathsf{normal}}(x, (C, x)) \\ &= C(x) \\ &= \mathsf{Eval}_{\mathsf{sim}}(x, C(x)) = \mathsf{Eval}(\mathsf{sim}, xC(x)) \end{aligned}$$

Since there are 2^D inputs to switch the ciphertexts for, the claim follows from 2^D hybrids each reducing to the 2^D-query adaptive security of 1-key FE. □

Next, we move on to the main technical part of the proof – showing that the intermediate hybrids for depths δ and $\delta + 1$ are indistinguishable.

Claim 8.8. If f, G_v, and FE are $2^{|\delta|}$-secure relative to $\mathcal{R}$, then

$$\mathsf{Hyb}_{\delta,\$\$} \approx \mathsf{Hyb}_{\delta+1,\$\$}$$

Proof. To transition between δ and $\delta + 1$, we perform a hybrid argument over every input prefix $\chi \in \{0,1\}^\delta$ where we modify the corresponding intermediate ciphertext ct_χ in a block-by-block manner. We will transition along the following sequence of hybrids in lexicographical order over $\chi \in \{0,1\}^\delta$.

- $\mathsf{Hyb}_{\delta,\chi,\$\$}$ modifies how depth δ behaves on message prefixes $< \chi$. Specifically, $\chi' \in \{0,1\}^\delta$ at depth δ is treated as in $\mathsf{Hyb}_{\delta,\$\$}$ if $\chi' < \chi$ and is treated as in $\mathsf{Hyb}_{\delta+1,\$\$}$ if $\chi' \geq \chi$. This affects the following: the plaintext of $\mathsf{ct}_{\chi'}$, QPrO queries on $(\mathsf{Eval}, h_{\delta,\beta}, \chi' \| 0^{D-|\chi'|})$, the encryption randomness $r_{\chi'}$, and the PRG seed $s_{\chi'}$.
- $\mathsf{Hyb}_{\delta,\chi,sr}$ is the same as $\mathsf{Hyb}_{\delta,\chi+1,\$\$}$,[6] except that the PRG seed s_χ and encryption randomness r_χ are generated as uniformly random.

The main step is to show that

$$\mathsf{Hyb}_{\delta,\chi} \approx_{2^{-\ell}\mathsf{negl}(\lambda)} \mathsf{Hyb}_{\delta,\chi,sr} \tag{2}$$

$$\approx_{2^{-\ell}\mathsf{negl}(\lambda)} \mathsf{Hyb}_{\delta,\chi+1} \tag{3}$$

Invoking this $2^{|\delta|}$ times to cover each $\chi \in \{0,1\}^\delta$ gives the claim.

Equation (2) follows immediately from the $2^{|\delta|}$-security of G_{sr}. To show Equation (3), we iterate across the following hybrids over each block $\beta = 1$ to $\beta = B$.

[6] We emphasize the $\chi + 1$ here.

- $\mathsf{Hyb}_{\delta,\chi,\beta,1}$: Change $\mathsf{ct}_\chi = \mathsf{Enc}_{\mathsf{pk}_\delta}(\mathsf{flag}_\chi, \mathsf{info}_\chi; r_\chi)$ to hardcode the β'th block of $\mathsf{ct}_{\chi\|0}$ and $\mathsf{ct}_{\chi\|1}$, instead of computing them on the fly. Specifically, modify the plaintext to

$$\mathsf{flag}_\chi := \mathsf{hyb}$$
$$\mathsf{info}_\chi := \left(C,\ \{k_{d,b}\}_{\substack{\delta<d<D,\\ 1\le j\le B}},\ s_\chi,\ \beta,\ \{F_\sigma(\chi,b)\}_{1\le b\le\beta},\ w_{\chi,\beta},\ \{k_{\delta,b}\}_{\beta<j\le B}\right)$$

  where

$$w_{\chi,\beta} := \left[\mathsf{ct}_{\chi\|0}\|\mathsf{ct}_{\chi\|1}\right]_\beta \oplus f_{k_{\delta,\beta}}(\chi\|0^{D-\delta})$$

  Note that for $\beta > 1$, this recycles the space used for $w_{\chi,\beta-1} = G_v(F_\sigma(\chi,\beta-1))$ by directly storing the much-shorter seed $F_\sigma(\chi,\beta-1)$ in the other parts of the ciphertext.

- $\mathsf{Hyb}_{\delta,\chi,\beta,2}$: Replace $f_{k_{\delta,\beta}}(\chi\|0^{D-\delta})$ by $F_{\delta,\beta}(\chi\|0^{D-\delta})$. Note that this modifies both the hardcoded $w_{\chi,\beta}$ and the reply to $\mathsf{Sim}_3^{\delta,\$\$}(\mathsf{Eval}, h_{\delta,\beta}, \chi\|0^{D-\delta})$.

  ,

- $\mathsf{Hyb}_{\delta,\chi,\beta,3}$ Swap the role of QPrO queries on (Eval, with the role of w_χ. Specifically, set

$$w_\chi := F_{\delta,\beta}(\chi\|0^{D-\delta})$$

  and reply to $\mathsf{Sim}_3^{\delta,\$\$}(\mathsf{Eval}, h_{\delta,\beta}, \chi\|0^{D-\delta})$ with

$$\left[\mathsf{ct}_{\chi\|0}\|\mathsf{ct}_{\chi\|1}\right]_\beta \oplus F_{\delta,\beta}(\chi\|0^{D-\delta})$$

- $\mathsf{Hyb}_{\delta,\chi,\beta,4}$: Replace $F_{\delta,\beta}(\chi\|0^{D-\delta})$ by $G_v(F_\sigma(\chi,\beta)$. Note that this modifies both the hardcoded $w_{\chi,\beta}$ and the reply to $\mathsf{Sim}_3^{\delta,\$\$}(\mathsf{Eval}, h_{\delta,\beta}, \chi\|0^{D-\delta})$.

Using these hybrids, we show Equation (3) via the transitions

$$\begin{aligned}\mathsf{Hyb}_{\delta,\chi,sr} &\approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \mathsf{Hyb}_{\delta,\chi,1,1} \approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \cdots \approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \mathsf{Hyb}_{\delta,\chi,1,5}\\ &\cdots\\ &\approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \mathsf{Hyb}_{\delta,\chi,B,1} \approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \cdots \approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \mathsf{Hyb}_{\delta,\chi,B,5}\\ &\approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \mathsf{Hyb}_{\delta,\chi+1}\end{aligned}$$

Claim 8.9 (Subclaim of Theorem 8.8)**.** If FE is $2^{|\delta|}$-subexponentially adaptively secure, then

$$\mathsf{Hyb}_{\delta,\chi,sr} \approx_{2^{-\ell_{\mathsf{negl}}(\lambda)}} \mathsf{Hyb}_{\delta,\chi,1,1}$$

Proof. The only difference between these two hybrids is ct_χ. Specifically, it is an encryption of

$(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$ where[7]

$$\mathsf{flag}_\chi := \begin{cases} \mathsf{normal} & \text{in } \mathsf{Hyb}_{\delta,\chi,sr} \\ \mathsf{hyb} & \text{in } \mathsf{Hyb}_{\delta,\chi,1,1} \end{cases}$$

$$\mathsf{info}_\chi := \begin{cases} \left(C, \{k_{d,b}\}_{\substack{|\chi|\le d\le D,\\ 1\le b\le B}}, s_\chi\right) & \text{in } \mathsf{Hyb}_{\delta,\chi,sr} \\ \left(C,\ \{k_{d,b}\}_{\substack{\delta<d<D,\\ 1\le j\le B}},\ s_\chi,\ 1,\ \{F_\sigma(\chi,b)\}_{1\le b\le 1},\ w_{\chi,1},\ \{k_{\delta,b}\}_{1<j\le B}\right) & \text{in } \mathsf{Hyb}_{\delta,\chi,1,1} \end{cases}$$

sk_δ is a functional encryption key for Expand_δ. Observe that for both settings of $(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$ described above,

$$\mathsf{Expand}_\delta(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi) = \left(\mathsf{ct}_{\chi\|0}\|\mathsf{ct}_{\chi\|1}\right) \oplus \left(f_{k_{\delta,1}}(\chi\|0^{D-\delta})\|\ldots\|f_{k_{\delta,1}}(\chi\|0^{D-\delta})\right)$$

To see this, recall that in hybrid mode Expand_δ computes all blocks the same, except for block $\beta = 1$ in this case. In that position, it outputs the precomputed $w_{\chi,1}$, which matches the normal mode computation for that block.

Therefore the claim follows from the $2^{-\ell}$-subexponential adaptive security of FE. □

Claim 8.10 (Subclaim of Theorem 8.8)**.** If f and G_v are $2^{|\delta|}$-subexponentially secure then for all $\beta \in [1, B]$,

$$\mathsf{Hyb}_{\delta,\chi,\beta,1} \approx_{2^{-\ell}\mathsf{negl}(\lambda)} \mathsf{Hyb}_{\delta,\chi,\beta,2} = \mathsf{Hyb}_{\delta,\chi,\beta,3} \approx_{2^{-\ell}\mathsf{negl}(\lambda)} \mathsf{Hyb}_{\delta,\chi,\beta,4}$$

Proof. The first transition follows from the security of f as a pseudorandom function. The second follows from the perfect secrecy of the one-time pad. The third follows from the security of G_v as a pseudorandom generator. □

Claim 8.11 (Subclaim of Theorem 8.8)**.** If FE is $2^{|\delta|}$-subexponentially adaptively secure, then for all $\beta \in [1, B]$,

$$\mathsf{Hyb}_{\delta,\chi,\beta,5} \approx_{2^{-\ell}\mathsf{negl}(\lambda)} \mathsf{Hyb}_{\delta,\chi,\beta+1,1}$$

Proof. The only difference between these two hybrids is ct_χ. Specifically, it is an encryption of $(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$ where[8]

$\mathsf{flag}_\chi := \mathsf{hyb}$

$$\mathsf{info}_\chi := \begin{cases} \left(C,\ \{k_{d,b}\}_{\substack{\delta<d<D,\\ 1\le j\le B}},\ s_\chi,\ \beta,\ \{F_\sigma(\chi,b)\}_{1\le b\le \beta},\ w_{\chi,\beta},\ \{k_{\delta,b}\}_{\beta<j\le B}\right) & \text{in } \mathsf{Hyb}_{\delta,\chi,\beta,5} \\ \left(C,\ \{k_{d,b}\}_{\substack{\delta<d<D,\\ 1\le j\le B}},\ s_\chi,\ \beta+1,\ \{F_\sigma(\chi,b)\}_{1\le b\le \beta+1},\ w_{\chi,\beta+1},\ \{k_{\delta,b}\}_{\beta+1<j\le B}\right) & \text{in } \mathsf{Hyb}_{\delta,\chi,\beta+1,1} \end{cases}$$

sk_δ is a functional encryption key for Expand_δ. Observe that for both settings of $(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$ described above,

$$\mathsf{Expand}_\delta(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi) = \left(\mathsf{ct}_{\chi\|0}\|\mathsf{ct}_{\chi\|1}\right) \oplus \left(f_{k_{\delta,1}}(\chi\|0^{D-\delta})\|\ldots\|f_{k_{\delta,1}}(\chi\|0^{D-\delta})\right)$$

[7] Dependence on $\beta = 1$ highlighted.
[8] Dependence on $\beta = 1$ highlighted.

To see this, observe that the only difference in evaluation comes from evaluating blocks β and $\beta+1$. In $\mathsf{Hyb}_{\delta,\chi,\beta+1,1}$, block $\beta+1$ is output as the hard-coded $w_{\chi,\beta+1}$, which is pre-computed to be the same as the evaluation of block $\beta+1$ in $\mathsf{Hyb}_{\delta,\chi,\beta+1,1}$. Similarly, in $\mathsf{Hyb}_{\delta,\chi,\beta,5}$, block β is output as the hard-coded $w_{\chi,\beta}$, which is pre-computed to be the same as the evaluation of block β in $\mathsf{Hyb}_{\delta,\chi,\beta+1,1}$.

Therefore the claim follows from the $2^{-\ell}$-subexponential adaptive security of FE. □

Claim 8.12 (Subclaim of Theorem 8.8)**.** If FE is $2^{|\delta|}$-subexponentially adaptively secure, then

$$\mathsf{Hyb}_{\delta,\chi,B,5} \approx_{2^{-\ell}\mathsf{negl}(\lambda)} \mathsf{Hyb}_{\delta,\chi+1}$$

Proof. The only difference between these hybrids is the construction of ct_χ using a hybrid flag or a sim flag. Specifically, it is an encryption of $(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$ where

$$\mathsf{flag}_\chi := \begin{cases} \mathsf{hyb} & \text{in } \mathsf{Hyb}_{\delta,\chi,B,5} \\ \mathsf{sim} & \text{in } \mathsf{Hyb}_{\delta,\chi+1} \end{cases}$$

$$\mathsf{info}_\chi := \begin{cases} \left(C,\ \{k_{d,b}\}_{\substack{\delta<d<D \\ 1\le j\le B}},\ s_\chi,\ B,\ \{F_\sigma(\chi,b)\}_{1\le b\le B},\ w_{\chi,B}\right) & \text{in } \mathsf{Hyb}_{\delta,\chi,B,5} \\ \{F_\sigma(\chi,b)\}_{1\le b\le B} & \text{in } \mathsf{Hyb}_{\delta,\chi+1} \end{cases}$$

sk_δ is a functional encryption key for the function Expand_δ. So, to reduce to the security of FE we only need show that Eval behaves the same on both settings of $(\mathsf{flag}_\chi, \chi, \mathsf{info}_\chi)$. Observe that

$$\begin{aligned} &\mathsf{Expand}_\delta\left(\mathsf{hyb}, \chi, \left(C,\ \{k_{d,b}\}_{\substack{\delta<d<D \\ 1\le j\le B}},\ s_\chi,\ B,\ \{F_\sigma(\chi,b)\}_{1\le b\le B},\ w_{\chi,B}\right)\right) \\ &= G_v(F_\sigma(\chi,1))\|\ldots\|G_v(F_\sigma(\chi,B)) \\ &= \mathsf{Expand}_\delta(\mathsf{sim}, \chi, \{F_\sigma(\chi,b)\}_{1\le b\le B}) \end{aligned}$$

Thus the claim follows from the $2^{-\delta}$-subexponential adaptive security of FE. □

□

□

9 NIZK Arguments of Knowledge for QMA

9.1 Definition

Definition 9.1 (NIZKAoK (PoK) for QMA in the CRS Model)**.** Let $\mathcal{L} = (\mathcal{L}_{yes}, \mathcal{L}_{no})$ be a QMA promise problem with relation $(Q, \alpha(\cdot), \beta(\cdot))$, where $\alpha(n) = 1 - \mathsf{negl}(n)$ and $\beta(n) = \mathsf{negl}(n)$. Let $\gamma = \gamma(n)$ be a "witness quality" parameter, where $\gamma(n) \in [1/\mathsf{poly}(n), 1 - 1/\mathsf{poly}(n)]$. $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ is a non-interactive, zero-knowledge argument (resp. proof) of knowledge for $\mathcal{L}$ in the CRS model if it has the following syntax and properties.

Syntax.

- $\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)$: The classical polynomial-size circuit Setup on input 1^λ outputs a common reference string crs.
- $\pi \leftarrow \mathsf{P}(1^\lambda, \mathsf{crs}, x, |\psi\rangle)$: The quantum polynomial-size circuit P on input the security parameter 1^λ, a common random string crs, an instance $x \in \{0,1\}^n$ for $n \in [\lambda, \mathsf{poly}(\lambda)]$, and a witness $|\psi\rangle$, outputs a (potentially quantum) proof π.
- $\mathsf{V}(1^\lambda, \gamma, \mathsf{crs}, x, \pi) \in \{0,1\}$: The quantum polynomial-size circuit V on input the security parameter 1^λ, the quality parameter γ, a common reference string crs, an instance x, and a proof π, outputs a bit.

Properties.

- **Uniform Random String.** Π satisfies the uniform random string property of Theorem 3.16.
- **Completeness.** There exists a negligible function $\mathsf{negl}(\cdot)$ such that for every $\gamma(\cdot)$, every $\lambda \in \mathbb{N}$, every $n \in [\lambda, \mathsf{poly}(\lambda)]$, and every $(x, |\psi\rangle) \in R_{Q,\alpha(n)}$,

$$\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\ \pi\leftarrow\mathsf{P}(1^\lambda,\mathsf{crs},x,|\psi\rangle)}}[\mathsf{V}(1^\lambda, \gamma(n), \mathsf{crs}, x, \pi) = 1] = 1 - \mathsf{negl}(\lambda).$$

- **Computational (resp. Statistical) Soundness.** There exists a negligible function $\mathsf{negl}(\cdot)$ such that for every $\gamma(\cdot)$, every quantum polynomial-size (resp. unbounded) adversary $\mathcal{A}$, every sufficiently large $\lambda \in \mathbb{N}$, every $n \in [\lambda, \mathsf{poly}(\lambda)]$, and every $x \in N_{Q,\beta(n)}$,

$$\Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)\\ \pi\leftarrow\mathcal{A}(\mathsf{crs})}}[\mathsf{V}(1^\lambda, \gamma(n), \mathsf{crs}, x, \pi) = 1] = \mathsf{negl}(\lambda).$$

- **Adaptive Argument (resp. Proof) of Quantum Knowledge.** There exists a quantum polynomial-size extractor $\mathsf{Ext} = (\mathsf{Ext}_0, \mathsf{Ext}_1)$, and negligible functions $\mathsf{negl}_0(\cdot), \mathsf{negl}_1(\cdot), \mathsf{negl}_2(\lambda)$ such that:
  1. For every quantum polynomial-size (resp. unbounded) distinguisher $\mathcal{D}$, every sufficiently large $\lambda \in \mathbb{N}$, and every $n \in [\lambda, \mathsf{poly}(\lambda)]$,

$$\left|\Pr_{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)}[\mathcal{D}(\mathsf{crs}) = 1] - \Pr_{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)}[\mathcal{D}(\mathsf{crs}) = 1]\right| \leq \mathsf{negl}_0(\lambda).$$

  2. Given $\lambda, x, \gamma, \mathsf{crs}, \mathsf{td}$, and π, define

$$p_V := \Pr\Big[\mathsf{V}(1^\lambda, \gamma(n), \mathsf{crs}, x, \pi) = 1\Big], \quad p_E := \Pr_{\rho\leftarrow\mathsf{Ext}_1(1^\lambda,\gamma(n),\mathsf{crs},\mathsf{td},x,\pi)}[(x,\rho) \in R_{Q,\gamma(n)}].$$

  Then, for every $\gamma(\cdot)$, every quantum polynomial-size (resp. unbounded) adversary $\mathcal{A}$, and every sufficiently large $\lambda \in \mathbb{N}$,

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)\\ (x,\pi)\leftarrow\mathcal{A}(\mathsf{crs})}}[p_E \geq p_V - \mathsf{negl}_1(\lambda)] \geq 1 - \mathsf{negl}_2(\lambda).$$

- **Non-Adaptive Computational Zero-Knowledge.** There exists a quantum polynomial-size Sim and a negligible function $\mathsf{negl}(\cdot)$ such that for every quantum polynomial-size distinguisher $\mathcal{D}$, every sufficiently large $\lambda \in \mathbb{N}$, every $n \in [\lambda, \mathsf{poly}(\lambda)]$, and every $(x, |\psi\rangle) \in R_{Q,\alpha(n)}$,

$$\left| \Pr_{\substack{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda) \\ \pi\leftarrow\mathsf{P}(1^\lambda,\mathsf{crs},x,|\psi\rangle)}} [\mathcal{D}(\mathsf{crs}, x, \pi) = 1] - \Pr_{(\mathsf{crs},\pi)\leftarrow\mathsf{Sim}(1^\lambda)} [\mathcal{D}(\mathsf{crs}, x, \pi) = 1] \right| \leq \mathsf{negl}(\lambda).$$

Finally, we formulate a novel definition for the notion of argument / proof of quantum knowledge where the extractor can extract from *already-verified* proofs.

- **Adaptive Post-Verified Argument (resp. Proof) of Quantum Knowledge.** There exists a quantum polynomial-size extractor $\mathsf{Ext} = (\mathsf{Ext}_0, \mathsf{Ext}_1)$, and negligible functions $\mathsf{negl}_0(\cdot), \mathsf{negl}_1(\cdot)$ such that:

  1. For every quantum polynomial-size (resp. unbounded) distinguisher $\mathcal{D}$, every sufficiently large $\lambda \in \mathbb{N}$, and every $n \in [\lambda, \mathsf{poly}(\lambda)]$,

$$\left| \Pr_{\mathsf{crs}\leftarrow\mathsf{Setup}(1^\lambda)} [\mathcal{D}(\mathsf{crs}) = 1] - \Pr_{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)} [\mathcal{D}(\mathsf{crs}) = 1] \right| \leq \mathsf{negl}_0(\lambda).$$

  2. For every quantum polynomial-size (resp. unbounded) adversary $\mathcal{A}$, every sufficiently large $\lambda \in \mathbb{N}$, and every $n \in [\lambda, \mathsf{poly}(\lambda)]$,

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda) \\ (x,\pi)\leftarrow\mathcal{A}(\mathsf{crs}) \\ (b,\pi')\leftarrow\mathsf{V}(1^\lambda,\gamma(n),\mathsf{crs},x,\pi) \\ \rho\leftarrow\mathsf{Ext}_1(1^\lambda,\gamma(n),\mathsf{crs},\mathsf{td},x,\pi')}} \left[b = 1 \wedge (x, \rho) \notin R_{Q,\gamma(n)}\right] \leq \mathsf{negl}_1(\lambda).$$

REMARK 9.1. It is easy to see that the notion of *post-verified* argument (resp. proof) of quantum knowledge implies the standard notion. Indeed, we can define the extractor for the standard notion to apply the honest verifier followed by the post-verified extractor. It is also straightforward to see that, for any choice of $\gamma(n) \in [1/\mathsf{poly}(n), 1 - 1/\mathsf{poly}(n)]$, the standard notion of argument (resp. proof) of knowledge implies computational (resp. statistical) soundness.

9.2 Protocol

Let $\{\theta_{H,i}, f_{H,i}\}_{i\in[N]}$ denote a ZX verifier with strong completeness (Theorem 4.1) for some QMA language $\mathcal{L} = (\mathcal{L}_{yes}, \mathcal{L}_{no})$, where each instance H is an n-qubit Hamiltonian.

Let $\mathsf{CSA} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Ver})$ be a publicly-verifiable CSA (Theorem 5.1).

Let $\mathcal{O} = (\mathsf{Setup}, \mathsf{Obf}, \mathsf{Eval}, \mathsf{Ver})$ be a provably-correct obfuscation scheme (Theorem 7.1).

Let $G : \{0,1\}^\lambda \to \{0,1\}^{\mathsf{poly}(\lambda)}$ be a pseudorandom generator with arbitrary polynomial stretch.

$\mathsf{Setup}(1^\lambda)$:

1. Generate the public parameters for the provably-correct obfuscation scheme Obf. Formally, compute $\mathsf{pp} \leftarrow \mathcal{O}.\mathsf{Setup}(1^{\lambda_{\mathsf{ZK}}})$, where $\lambda_{\mathsf{ZK}} = \lambda$, and output $\mathsf{crs} = \mathsf{pp}$.

$\mathsf{P}(1^\lambda, \mathsf{crs}, H, |\psi\rangle)$:

1. Encode the witness using the CSA scheme. Formally,
   (a) Sample a key $k \leftarrow \mathsf{CSA.KeyGen}(1^{d(\lambda^2,m)}, 1^m)$, where m is the number of qubits in $|\psi\rangle$, and $d(\cdot,\cdot)$ is maximum of the functions specified in the statements of Theorem 5.3 and Theorem 5.5.
   (b) Encode the witness as $|\phi\rangle = \mathsf{CSA.Enc}_k(|\psi\rangle)$.
2. Define a classical circuit $\mathcal{M}_k$ which outputs a decoding of the witness. Formally,
   (a) Define $\mathcal{M}_k$ hardwired with k which, on input (r, s):
      i. Compute $i = G(r)$, and set $(\theta, f) = (\theta_{H,i}, f_{H,i})$.
      ii. Output $\mathsf{CSA.Dec}_{k,\theta,\overline{f}}(s)$, where $\overline{f} = 1 - f$.
3. Obfuscate the CSA verifier and classical circuit $\mathcal{M}[k]$. Formally,
   (a) Define predicate φ as $\varphi(C) = 1$ iff there exists k' such that $C = \mathsf{CSA.Ver}_{k',\bullet} \| \mathcal{M}_{k'}$.
   (b) Compute $\widetilde{C} \leftarrow \mathcal{O}.\mathsf{Obf}(1^{q(\lambda^2,m)}, \mathsf{pp}, \varphi, \mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k)$, where $q(\cdot,\cdot)$ is the maximum of $\lambda^{2/\epsilon}$ and the functions specified in the statements of Theorem 5.3 and Theorem 5.5. Here, ϵ is the constant from the Simulated-Circuit Indistinguishability property of the provable-obfuscation (Theorem 7.1).
4. Output $\pi = (|\phi\rangle, \widetilde{C})$.

$\mathsf{V}(1^\lambda, \gamma, \mathsf{crs}, H, \pi)$:

1. Parse all inputs. Formally,
   (a) Parse $\mathsf{crs} = \mathsf{pp}$.
   (b) Parse $\pi = (\rho, \widetilde{C})$, define $\widetilde{\mathcal{V}} = \widetilde{C}(0, \bullet, \bullet)$, and define $\widetilde{\mathcal{M}} = \widetilde{C}(1, \bullet, \bullet)$.
2. Verify that the provable obfuscation's verifier accepts. Formally,
   (a) Define predicate φ as $\varphi(C) = 1$ iff there exists k' such that $C = \mathsf{CSA.Ver}_{k',\bullet} \| \mathcal{M}_{k'}$.
   (b) Verify that $\mathcal{O}.\mathsf{Ver}(\mathsf{pp}, \varphi, \widetilde{C}) = 1$.
3. Verify that the witness was encoded correctly using the obfuscated CSA verifier and check that the obfuscated $\mathcal{M}$ accepts. Formally,
   (a) Define a POVM $(\mathcal{P}_1, \mathcal{P}_0)$ where $\mathcal{P}_1 = \frac{1}{2^\lambda} \sum_{r \in \{0,1\}^\lambda} P_r$, $\mathcal{P}_0 = \frac{1}{2^\lambda} \sum_{r \in \{0,1\}^\lambda} (I - P_r)$, and P_r applied to state ρ performs the following checks:
      i. $(1, \rho') = \mathcal{O}.\mathsf{Eval}(\widetilde{\mathcal{V}}, (0^m, \rho))$,
      ii. $(1, \mathsf{Had}^{1^m}(\rho'')) = \mathcal{O}.\mathsf{Eval}(\widetilde{\mathcal{V}}, (1^m, \mathsf{Had}^{1^m}(\rho')))$, and

iii. $(1, \rho''') = I - \widetilde{\mathcal{M}}(r, \mathsf{Had}^{\theta_{H,G(r)}}(\rho''))$.

(b) Let $\mathsf{ATI}_{1-\gamma'}$ (Theorem 3.7) be defined according to the POVM $(\mathcal{P}_1, \mathcal{P}_0)$ for $\gamma' \in [1/\mathsf{poly}(n), 1 - 1/\mathsf{poly}(n)]$ such that there is some polynomial $p(\cdot)$ where $\gamma' = \gamma + \frac{1}{p(n)}$.

(c) Compute $(b, \rho_b^*) \leftarrow \mathsf{ATI}_{1-\gamma'}(\rho)$.

4. Reconstruct the proof. Formally,

(a) Define $\pi' = (\rho_b^*, \widetilde{C})$.

5. Output (b, π').

9.3 Analysis

Theorem 9.2. *Given that*

- $\{\theta_{H,i}, f_{H,i}\}_{i\in[N]}$ *is a ZX verifier with strong completeness (Theorem 4.1) for QMA language* $\mathcal{L} = (\mathcal{L}_{yes}, \mathcal{L}_{no})$*,*
- CSA *is a publicly-verifiable CSA (Theorem 5.1),*
- $\mathcal{O} = (\mathsf{Setup}, \mathsf{Obf}, \mathsf{Eval}, \mathsf{Ver})$ *is a sub-exponentially secure provably-correct obfuscator (Theorem 7.1) with computational (statistical) knowledge soundness, and*
- G *is a pseudorandom generator,*

then, for any $\gamma \in [1/\mathsf{poly}(n), 1 - 1/\mathsf{poly}(n)]$*, the construction in Section 9.2 is a non-interactive zero-knowledge argument (proof) of knowledge for language* $\mathcal{L}$ *(Theorem 9.1) that satisfies*

- *Completeness,*
- *Adaptive post-verified argument (proof) of quantum knowledge,*
- *Non-adaptive computational zero-knowledge.*

If the obfuscator is in the URS model, then the NIZK argument of knowledge for QMA is in the URS model.

REMARK 9.2. Since every promise problem in QMA has a ZX verifier with strong completeness (Theorem 4.2), the above theorem gives an argument of quantum knowledge for all of QMA.

Proof. **Correctness.** This follows from the correctness of CSA (Theorem 5.2), completeness of ZX verifier with strong completeness (Theorem 4.2), functionality-preservation and completeness of provably-correct obfuscation (Theorem 7.1), and correctness of $\mathsf{ATI}_{1-\gamma'}$ (Theorem 3.7).

Adaptive Post-Verified Argument (Proof) of Quantum Knowledge.

Let $(\mathcal{O}.\mathsf{Ext}_0, \mathcal{O}.\mathsf{Ext}_1)$ be the proof of knowledge extractor of $\mathcal{O}$. We define Ext_0 with oracle access to $\mathcal{O}.\mathsf{Ext}_0$ as follows:

Input: 1^λ

1. Compute $(\mathsf{pp}, \mathsf{td}) \leftarrow \mathcal{O}.\mathsf{Ext}_0(1^\lambda)$.
2. Output $\mathsf{crs} = \mathsf{pp}$ and td.

We define Ext_1 with oracle access to $\mathcal{O}.\mathsf{Ext}_1$ as follows:

Input: γ, $\mathsf{crs} = \mathsf{pp}$ and td, H, $\pi^* = (\rho^*, \widetilde{C})$.

1. Define predicate φ as $\varphi(C) = 1$ iff there exists k' such that $C = \mathsf{CSA.Ver}_{k',\bullet} \| \mathcal{M}_{k'}$ for $\mathcal{M}$ defined in item 2.
2. Compute $C \leftarrow \mathcal{O}.\mathsf{Ext}_1(\mathsf{pp}, \mathsf{td}, \varphi, \widetilde{C})$.
3. Parse $C = \mathsf{CSA.Ver}_{k',\bullet} \| \mathcal{M}_{k'}$.
4. Compute $(_, \rho') \leftarrow \mathsf{CSA.Ver}_{k',0^n}(\rho^*)$.
5. Compute $(_, \mathsf{Had}^{1^n}(\rho'')) \leftarrow \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))$.
6. Compute $\rho_\psi = \mathsf{CSA.Enc}^{\dagger}_{k'}(\rho'')$.
7. Output ρ_ψ.

The output of Setup and Ext_0 is computationally indistinguishable by the computational indistinguishability of $\mathcal{O}.\mathsf{Setup}$ and $\mathcal{O}.\mathsf{Ext}_0$ from the knowledge soundness of provably-correct obfuscation (Theorem 7.1).

Let a polynomial $p(\cdot)$ and a polynomial-size quantum circuit $\mathcal{A}$ be given such that for every sufficiently large $\lambda \in \mathbb{N}$,

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (H,\pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ (b,\pi') \leftarrow \mathsf{V}(\gamma,\mathsf{crs},H,\pi) \\ \rho_\psi \leftarrow \mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}} [b = 1 \wedge (H, \rho_\psi) \notin \mathcal{R}_{Q,\gamma}] \geq \frac{1}{p(\lambda)}. \tag{4}$$

By the knowledge soundness of provably-correct obfuscation (Theorem 7.1), we have that there exists a negligible function $\mathsf{negl}(\cdot)$

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (H,\pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ (b,\pi') \leftarrow \mathsf{V}(\gamma,\mathsf{crs},H,\pi) \\ C \leftarrow \mathcal{O}.\mathsf{Ext}_1(\mathsf{pp},\mathsf{td},\varphi,\widetilde{C})}} \left[\begin{array}{l} (b = 0) \vee \\ \left(\forall x, C(x) = \mathsf{Eval}(\widetilde{C}, x) \wedge \varphi(C) = 1\right) \end{array} \right] \geq 1 - \mathsf{negl}(\lambda). \tag{5}$$

Hence, by Equation (4) and Equation (5), we have that there exists a polynomial $p'(\cdot)$ such that

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{Ext}_0(1^\lambda) \\ (H,\pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ (b,\pi') \leftarrow \mathsf{V}(\gamma,\mathsf{crs},H,\pi) \\ \rho_\psi \leftarrow \mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}} \left[\begin{array}{l} b = 1 \wedge \left(\forall x, C(x) = \mathsf{Eval}(\widetilde{C}, x)\right) \wedge \\ \varphi(C) = 1 \wedge (H, \rho_\psi) \notin R_{Q,\gamma} \end{array} \right] \geq \frac{1}{p'(\lambda)}. \tag{6}$$

Let the variables sampled according to Equation (6) be given. When $\varphi(C) = 1$, this implies that there exists k' such that $C = \mathsf{CSA.Ver}_{k',\bullet} \| \mathcal{M}_{k'}$ (by definition of φ). Additionally, when $\forall x, C(x) = \mathsf{Eval}(\widetilde{C}, x)$ and $b = 1$, this means that $(\rho^*, 1) = \mathsf{ATI}_{1-\gamma'}(\rho)$ for POVM $(\mathcal{P}'_1, \mathcal{P}'_0)$ where $\mathcal{P}'_1 = \frac{1}{N}\sum_r P'_r$, $\mathcal{P}'_0 = \frac{1}{N}\sum_r (I - P'_r)$, and P'_r applied to state ρ performs the following checks:

1. $(1, \rho') = \mathsf{CSA.Ver}_{k',0^n}(\rho)$,
2. $(1, \mathsf{Had}^{1^n}(\rho'')) = \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))$, and
3. $(1, \rho''') = (I - \mathsf{CSA.Dec}_{k',\theta_r,\overline{f_r}}(\mathsf{Had}^{\theta_r}(\cdot)))$ where $(\theta_r, f_r) \coloneqq \mathsf{Samp}(H; r)$ for all $r \in [N]$.

Therefore,

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)\\(H,\pi)\leftarrow\mathcal{A}(\mathsf{crs})\\(b,\pi'=\rho_b^*)\leftarrow\mathsf{V}(\gamma,\mathsf{crs},H,\pi)\\\rho_\psi\leftarrow\mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}}\left[\ b = 1 \wedge (H, \rho_\psi) \notin R_{Q,\gamma}\ \right] \geq \frac{1}{p'(\lambda)}. \tag{7}$$

Let the variables sampled according to Equation (7) be given. By the soundness of $\mathsf{ATI}_{1-\gamma'}$ (Theorem 3.7), if $b = 1$ when running $\mathsf{ATI}_{1-\gamma'}$ with POVM $(\mathcal{P}'_1, \mathcal{P}'_0)$ defined previously, we have that $\mathrm{Tr}[\mathcal{P}'_1\rho_b^*] \geq \gamma'(n)$ with overwhelming probability. That is, there exists a polynomial $p''(\cdot)$ such that

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)\\(H,\pi)\leftarrow\mathcal{A}(\mathsf{crs})\\(b,\pi')\leftarrow\mathsf{V}(\gamma,\mathsf{crs},H,\pi)\\\rho_\psi\leftarrow\mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}}\left[\ (\mathrm{Tr}[\mathcal{P}'_1\rho_b^*] \geq \gamma'(n)) \wedge (H, \rho_\psi) \notin R_{Q,\gamma}\ \right] \geq \frac{1}{p''(\lambda)}. \tag{8}$$

Let the variables sampled according to Equation (8) be given. We note that Ext_1 performs item 4 followed by item 5 as in the first two steps of the POVM $(\mathcal{P}'_1, \mathcal{P}'_0)$. Hence, we use the same variables for comparison in the following analysis:

$$\mathrm{Tr}\left[\mathcal{P}'_1\rho_b^*\right] = \frac{1}{N}\sum_r \Pr_{\substack{(b',\rho')\leftarrow\mathsf{CSA.Ver}_{k',0^n}(\rho_b^*)\\(b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow\mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))\\(b''',\rho''')\leftarrow(I-\mathsf{CSA.Dec}_{k',\theta_r,\overline{f_r}}(\mathsf{Had}^{\theta_r}(\rho'')))}}[b' = b'' = 1 \wedge b''' = 1].$$

When the above event occurs, $(1, \rho') = \mathsf{Ver}_{k',0^n}(\rho_b^*)$ and $(1, \mathsf{Had}^{1^n}(\rho'')) = \mathsf{Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))$, then by CSA soundness (Theorem 5.4) we have that $\rho'' \in \mathsf{Enc}_{k'}$. Hence, using the above argument we have

$$\mathrm{Tr}\left[\mathcal{P}'_1\rho_b^*\right] \leq \frac{1}{N}\sum_r \Pr_{\substack{(b',\rho')\leftarrow\mathsf{CSA.Ver}_{k',0^n}(\rho_b^*)\\(b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow\mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))\\(b''',\rho''')\leftarrow(I-\mathsf{CSA.Dec}_{k',\theta_r,\overline{f_r}}(\mathsf{Had}^{\theta_r}(\rho'')))}}[\rho'' \in \mathsf{Enc}_{k'} \wedge b''' = 1].$$

Now, since $\rho'' \in \mathsf{Enc}_{k'}$ and $\rho_\psi = \mathsf{Enc}^\dagger_{k'}(\rho'')$ (definition of Ext_0), by the correctness of CSA (Theo-

rem 5.2),

$$
\begin{aligned}
&\mathrm{Tr}\left[\mathcal{P}'_1\rho^*_b\right]\\
&\leq \frac{1}{N}\sum_r \Pr_{\substack{(b',\rho')\leftarrow \mathsf{CSA.Ver}_{k',0^n}(\rho^*_b)\\ (b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))\\ (b''',\rho''')\leftarrow (I-\mathsf{CSA.Dec}_{k',\theta_r,\overline{f_r}}(\mathsf{Had}^{\theta_r}(\rho'')))}} [\rho''\in \mathsf{Enc}_{k'}\wedge b'''=1]\\
&= \frac{1}{N}\sum_r \Pr_{\substack{(b',\rho')\leftarrow \mathsf{CSA.Ver}_{k',0^n}(\rho^*_b)\\ (b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))}} [\rho''\in \mathsf{Enc}_{k'}\wedge \mathsf{CSA.Dec}_{k',\theta_r,\overline{f_r}}(\mathsf{Had}^{\theta_r}(\rho''))=0]\\
&= \frac{1}{N}\sum_r \Pr_{\substack{(b',\rho')\leftarrow \mathsf{CSA.Ver}_{k',0^n}(\rho^*_b)\\ (b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))}} [\rho''\in \mathsf{Enc}_{k'}\wedge \mathsf{CSA.Dec}_{k',\theta_r,f_r}(\mathsf{Had}^{\theta_r}(\rho''))=1]\\
&\leq \frac{1}{N}\sum_r \Pr_{\substack{(b',\rho')\leftarrow \mathsf{CSA.Ver}_{k',0^n}(\rho^*_b)\\ (b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))\\ \rho_\psi=\mathsf{Enc}^\dagger_{k'}(\rho'')}} [\mathsf{CSA.Dec}_{k',\theta_r,f_r}(\mathsf{Had}^{\theta_r}(\mathsf{Enc}_{k'}(\rho_\psi))=1]\\
&= \frac{1}{N}\sum_r \Pr_{\substack{(b',\rho')\leftarrow \mathsf{CSA.Ver}_{k',0^n}(\rho^*_b)\\ (b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))\\ \rho_\psi=\mathsf{Enc}^\dagger_{k'}(\rho'')}} [M[\theta_r,f_r](\rho_\psi)\text{ accepts}]\\
&= \mathbb{E}_{\substack{r\leftarrow\{0,1\}^\lambda\\ i\leftarrow G(r)\\ (b',\rho')\leftarrow \mathsf{CSA.Ver}_{k',0^n}(\rho^*_b)\\ (b'',\mathsf{Had}^{1^n}(\rho''))\leftarrow \mathsf{CSA.Ver}_{k',1^n}(\mathsf{Had}^{1^n}(\rho'))\\ \rho_\psi=\mathsf{Enc}^\dagger_{k'}(\rho'')}} \left[\mathrm{Tr}(M[\theta_{H,\lambda,i},f_{H,\lambda,i}]\rho_\psi)\right].
\end{aligned}
$$

Hence, using the above argument with Equation (8), we have that

$$
\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)\\ (H,\pi)\leftarrow\mathcal{A}(\mathsf{crs})\\ (b,\pi')\leftarrow\mathsf{V}(\gamma,\mathsf{crs},H,\pi)\\ \rho_\psi\leftarrow\mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}} \left[\begin{array}{l}\left(\mathbb{E}_{\substack{r\leftarrow\{0,1\}^\lambda\\ i\leftarrow G(r)}}\left[\mathrm{Tr}(M[\theta_{H,\lambda,i},f_{H,\lambda,i}]\rho_\psi)\right]\geq\gamma'(n)\right)\wedge\\ (H,\rho_\psi)\notin R_{Q,\gamma}\end{array}\right] \geq \frac{1}{p''(\lambda)}. \tag{9}
$$

We will now argue that we can switch a pseudorandom choice of measurement to a uniformly random choice of measurement.

Claim 9.3. There exists a negligible function $\mathsf{negl}(\cdot)$ such that for all $\rho\in D(\mathcal{B}^{\otimes p(|x|)})$,

$$
\left|\mathbb{E}_{\substack{r\leftarrow\{0,1\}^\lambda\\ i\leftarrow G(r)}}\left[\mathrm{Tr}(M[\theta_{H,\lambda,i},f_{H,\lambda,i}]\rho)\right]-\mathbb{E}_{i\leftarrow[\mathsf{poly}(\lambda)]}\left[\mathrm{Tr}(M[\theta_{H,\lambda,i},f_{H,\lambda,i}]\rho)\right]\right|\leq\mathsf{negl}(\lambda).
$$

Proof. This follows from the security of the pseudorandom generator G against non-uniform adversaries and the observation that the measurements $\{M[\theta_{H,\lambda,i},f_{H,\lambda,i}]\}_{i\in[\mathsf{poly}(\lambda)]}$ are efficiently computable. □

Therefore, by Equation (9) and Theorem 9.3, there exists a negligible function $\mathsf{negl}(\cdot)$ such that

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)\\(H,\pi)\leftarrow\mathcal{A}(\mathsf{crs})\\(b,\pi')\leftarrow\mathsf{V}(\gamma,\mathsf{crs},H,\pi)\\\rho_\psi\leftarrow\mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}}\left[\begin{array}{l}\left(\mathbb{E}_{i\leftarrow[\mathsf{poly}(\lambda)]}\left[\mathrm{Tr}(M[\theta_{H,\lambda,i},f_{H,\lambda,i}]\rho_\psi)\right]\geq\gamma'(n)-\mathsf{negl}(\lambda)\right)\wedge\\(H,\rho_\psi)\notin R_{Q,\gamma}\end{array}\right]\geq\frac{1}{p''(\lambda)},\quad(10)$$

and, by the definition of γ', this implies that

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)\\(H,\pi)\leftarrow\mathcal{A}(\mathsf{crs})\\(b,\pi')\leftarrow\mathsf{V}(\gamma,\mathsf{crs},H,\pi)\\\rho_\psi\leftarrow\mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}}\left[\begin{array}{l}\left(\mathbb{E}_{i\leftarrow[\mathsf{poly}(\lambda)]}\left[\mathrm{Tr}(M[\theta_{H,\lambda,i},f_{H,\lambda,i}]\rho_\psi)\right]\geq\gamma(n)\right)\wedge\\(H,\rho_\psi)\notin R_{Q,\gamma}\end{array}\right]\geq\frac{1}{p''(\lambda)}.\quad(11)$$

By the definition of $\mathcal{R}_{Q,\gamma}$ as the $\gamma(n)$-relation (Theorem 3.9) with respect to the ZX verifier with strong completeness (Theorem 4.1),

$$\Pr_{\substack{(\mathsf{crs},\mathsf{td})\leftarrow\mathsf{Ext}_0(1^\lambda)\\(H,\pi)\leftarrow\mathcal{A}(\mathsf{crs})\\(b,\pi')\leftarrow\mathsf{V}(\gamma,\mathsf{crs},H,\pi)\\\rho_\psi\leftarrow\mathsf{Ext}_1(\gamma,\mathsf{crs},\mathsf{td},H,\pi')}}\left[\ (H,\rho_\psi)\in R_{Q,\gamma}\wedge(H,\rho_\psi)\notin R_{Q,\gamma}\ \right]\geq\frac{1}{p''(\lambda)}.\quad(12)$$

Since this is a contradiction, we have proven knowledge soundness.

Computational Zero-Knowledge.

Let $\mathcal{O}.\mathsf{Sim}=(\mathcal{O}.\mathsf{SimGen},\mathcal{O}.\mathsf{SimObf})$ be the simulator from the simulation security of the provably-correct obfuscation $\mathcal{O}$. We define Sim_0 with oracle access to $\mathcal{O}.\mathsf{SimGen}$ as follows:

Input: 1^λ

1. Compute $(\mathsf{pp},\mathsf{td})\leftarrow\mathcal{O}.\mathsf{SimGen}(1^\lambda)$.
2. Output $\mathsf{crs}=\mathsf{pp}$ and td.

We define Sim_1 with oracle access to $\mathcal{O}.\mathsf{SimObf}$ as follows:

Input: crs, td, H

1. Sample a key $k\leftarrow\mathsf{CSA.KeyGen}(1^{d(\lambda^2,m)},1^m)$.
2. Encode dummy witness as $|\phi\rangle=\mathsf{CSA.Enc}_k(|0^m\rangle)$.
3. Define predicate φ as $\varphi(C)=1$ iff there exists k' such that $C=\mathsf{CSA.Ver}_{k',\bullet}\|\mathcal{M}_{k'}$ for $\mathcal{M}$ defined in item 2.
4. Compute $\widetilde{C}\leftarrow\mathcal{O}.\mathsf{SimObf}(1^{q(\lambda^2,m)},\mathsf{pp},\mathsf{td},\varphi,\mathsf{CSA.Ver}_{k,\bullet}\|C_{\mathsf{null}})$.
5. Output $\pi=(|\phi\rangle,\widetilde{C})$.

Let a polynomial-size quantum circuit $\mathcal{D}$, sufficiently large $\lambda\in\mathbb{N}$, and $(H,|\psi\rangle)\in R_{Q,1-\mathsf{negl}(n)}$ be given. We construct the following series of hybrids to argue $\mathsf{negl}(\lambda)$ indistinguishability of b from the honest distribution $\mathcal{H}_0$ and simulated distribution $\mathcal{H}_4$:

$\mathcal{H}_0$: Honest protocol: $\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)$. $\pi \leftarrow \mathsf{P}(1^\lambda, \mathsf{crs}, H, |\psi\rangle)$. $b \leftarrow \mathcal{D}(\mathsf{crs}, H, \pi)$.

$\mathcal{H}_1$: Same as $\mathcal{H}_0$ except that:

- Compute $(\mathsf{pp}, \mathsf{td}) \leftarrow \mathcal{O}.\mathsf{SimGen}(1^\lambda)$ and set $\mathsf{crs} = \mathsf{pp}$.
- Compute $\widetilde{C} \leftarrow \mathcal{O}.\mathsf{SimObf}(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k)$.

$\mathcal{H}_2$: Same as $\mathcal{H}_1$ except that:

- Compute $\widetilde{C} \leftarrow \mathcal{O}.\mathsf{SimObf}(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet} \| C_{\mathsf{null}})$.

$\mathcal{H}_3$: Same as $\mathcal{H}_2$ except that:

- Compute $|\phi\rangle = \mathsf{CSA.Enc}_k(|0\rangle)$.

$\mathcal{H}_4$: Simulated protocol: $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda)$. $\pi \leftarrow \mathsf{Sim}_1(\mathsf{crs}, \mathsf{td}, H)$. $b \leftarrow \mathcal{D}(\mathsf{crs}, H, \pi)$.

$\mathcal{H}_0$ and $\mathcal{H}_1$ are $\mathsf{negl}(\lambda)$ indistinguishable by the honest-to-simulated indistinguishability property of provably-correct obfuscation (Theorem 7.1). $\mathcal{H}_2$ and $\mathcal{H}_3$ are $\mathsf{negl}(\lambda)$ indistinguishable by the Hiding property of CSA scheme (Theorem 5.3). $\mathcal{H}_3$ and $\mathcal{H}_4$ are identical by definition of $(\mathsf{Sim}_0, \mathsf{Sim}_1)$.

All that remains to prove is that $\mathcal{H}_1$ and $\mathcal{H}_2$ are $\mathsf{negl}(\lambda)$ indistinguishable. We will show this for fixed randomness via a series of hybrids, then combine them using the evasive composability property of provably-correct obfuscation (Theorem 7.1).

Claim 9.4. Let $\mathcal{S}$ prepare $(H, |\psi\rangle)$, sample $k \leftarrow \mathsf{CSA.KeyGen}(1^{d(\lambda^2,m)}, 1^m)$, compute $|\phi\rangle \leftarrow \mathsf{CSA.Enc}_k(|\psi\rangle)$, and output $(|\phi\rangle, \{\mathcal{M}_k(r, \bullet)\}_{r \in \{0,1\}^\lambda})$ for $\mathcal{M}$ defined in item 2. For any $r^* \in \{0,1\}^\lambda$, any predicate φ, and any QPT adversary $\mathcal{A}$,

$$\left| \Pr_{\substack{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda) \\ (|\phi\rangle, \{\mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k(r,\bullet)\}_r) \leftarrow \mathcal{S}}} \left[\mathcal{A}\left(|\phi\rangle, \mathcal{O}.\mathsf{SimObf}\left(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k(r^*, \bullet)\right)\right) = 1 \right] \right.$$

$$\left. - \Pr_{\substack{(\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{Sim}_0(1^\lambda) \\ (|\phi\rangle, \{\mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k(r,\bullet)\}_r) \leftarrow \mathcal{S}}} \left[\mathcal{A}\left(|\phi\rangle, \mathcal{O}.\mathsf{SimObf}\left(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet} \| C_{\mathsf{null}}\right)\right) = 1 \right] \right| \leq \frac{\mathsf{negl}(\lambda)}{2^\lambda}.$$

Proof. Let $r^* \in \{0,1\}^\lambda$, φ, and $\mathcal{A}$ be given. We construct the following series of hybrids:

$\mathcal{H}_{1,0}$: Same as $\mathcal{H}_1$ above for fixed $r^* \in \{0,1\}^\lambda$:

- $(|\phi\rangle, \{\mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k(r, \bullet)\}_r) \leftarrow \mathcal{S}$.
- $\widetilde{C} \leftarrow \mathcal{O}.\mathsf{SimObf}\left(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k(r^*, \bullet)\right)$.
- $b \leftarrow \mathcal{A}(|\phi\rangle, \widetilde{C})$.

$\mathcal{H}_{1,1}$: Same as $\mathcal{H}_{1,0}$ except:

- Replace
$$|\psi\rangle \to |\psi'\rangle := \frac{M[\theta_{H,i}, f_{H,i}]\,|\psi\rangle}{\|M[\theta_{H,i}, f_{H,i}]\,|\psi\rangle\,\|},$$
where $i = G(r^*)$.

$\mathcal{H}_{1,2}$: Same as $\mathcal{H}_{1,1}$ except:

- Replace $\overline{f}$ in $\mathcal{M}_k(r^*, \bullet)$ with the zero function $f^* = 0$.

$\mathcal{H}_{1,3}$: Same as $\mathcal{H}_{1,2}$ except:

- Replace $\mathcal{M}_k(r^*, \bullet)$ from $\mathcal{H}_{1,2}$ with C_{null}.

$\mathcal{H}_{1,4}$: Same as $\mathcal{H}_{1,3}$ except:

- Replace $|\psi'\rangle$ defined in $\mathcal{H}_{1,1}$ with $|0^m\rangle$.

$\mathcal{H}_{1,5}$: Same as $\mathcal{H}_2$ above for fixed $r^* \in \{0,1\}^\lambda$:

- $(|\phi\rangle\,, \{\mathsf{CSA.Ver}_{k,\bullet} \| \mathcal{M}_k(r, \bullet)\}_r) \leftarrow \mathcal{S}$.
- $\widetilde{C} \leftarrow \mathcal{O}.\mathsf{SimObf}\left(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet} \| C_{\mathsf{null}}\right)$.
- $b \leftarrow \mathcal{A}(|\phi\rangle\,, \widetilde{C})$.

By the strong completeness of the ZX verifier (Theorem 4.2), Gentle Measurement (Theorem 3.4), and the fact that $n \geq \lambda$, we have that $\mathcal{H}_{1,0}$ and $\mathcal{H}_{1,1}$ are $2^{-3\lambda/2}$-indistinguishable = $(\mathsf{negl}(\lambda)/2^\lambda)$-indistinguishable.

By the Measurement Indistinguishability property of CSA (Theorem 5.5), we have that $\mathcal{H}_{1,1}$ and $\mathcal{H}_{1,2}$ are $2^{-\Omega(\lambda^2)}$-indistinguishable = $(\mathsf{negl}(\lambda)/2^\lambda)$-indistinguishable.

By the Simulated-Circuit Indistinguishability of the provable-obfuscation (Theorem 7.1), we have that $\mathcal{H}_{1,2}$ and $\mathcal{H}_{1,3}$ are $2^{-\lambda^2}$-indistinguishable = $(\mathsf{negl}(\lambda)/2^\lambda)$-indistinguishable.

By the Hiding property of CSA (Theorem 5.3), we have that $\mathcal{H}_{1,3}$ and $\mathcal{H}_{1,4}$ are $2^{-\Omega(\lambda^2)}$-indistinguishable = $(\mathsf{negl}(\lambda)/2^\lambda)$-indistinguishable.

$\mathcal{H}_{1,4}$ and $\mathcal{H}_{1,5}$ are identical by definition of $\mathcal{S}$. □

By Theorem 9.4 and the evasive composability property of the provable-obfuscation (Theorem 7.1), we have that for $\mathcal{S}$ (defined in Theorem 9.4), for any predicate φ, and for any QPT adversary $\mathcal{A}$,

there exists a negligible function $\mathsf{negl}(\cdot)$ such that

$$\left| \Pr_{\substack{(\mathsf{pp},\mathsf{td})\leftarrow\mathsf{Sim}_0(1^\lambda)\\ (|\phi\rangle,\{\mathsf{CSA.Ver}_{k,\bullet}\|\mathcal{M}_k(r,\bullet)\}_r)\leftarrow\mathcal{S}}} \left[\mathcal{A}\left(|\phi\rangle, \mathcal{O}.\mathsf{SimObf}\left(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet}\|\mathcal{M}_k\right)\right)=1\right] \right.$$

$$\left. - \Pr_{\substack{(\mathsf{pp},\mathsf{td})\leftarrow\mathsf{Sim}_0(1^\lambda)\\ (|\phi\rangle,\{\mathsf{CSA.Ver}_{k,\bullet}\|\mathcal{M}_k(r,\bullet)\}_r)\leftarrow\mathcal{S}}} \left[\mathcal{A}\left(|\phi\rangle, \mathcal{O}.\mathsf{SimObf}\left(1^{q(\lambda^2,m)}, \mathsf{pp}, \mathsf{td}, \varphi, \mathsf{CSA.Ver}_{k,\bullet}\|C_{\mathsf{null}}\right)\right)=1\right] \right| \leq \mathsf{negl}(\lambda).$$

This implies that $\mathcal{H}_1$ and $\mathcal{H}_2$ are $\mathsf{negl}(\lambda)$ indistinguishable, thus concluding the proof. □

Due to Theorem 7.6, we have the following corollaries.

Corollary 9.5. Assuming functional encryption with subexponential security (Theorem 3.15) and post-quantum NIZK arguments of knowledge for NP with a URS setup, there exists a NIZK argument of knowledge for QMA with a URS setup in the QPrO model (Theorem 3.23).

Corollary 9.6. Assuming indistinguishability obfuscation satisfies $\mathcal{S}$-evasive composability (Theorem 7.1) for $\mathcal{S}$ defined in Theorem 9.4 and post-quantum NIZK proof of knowledge for NP with a CRS setup, there exists a NIZK proof of knowledge for QMA with a CRS setup.

Acknowledgements. RJ and KT were supported in part by AFOSR, NSF 2112890, NSF CNS-2247727 and a Google Research Scholar award. This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-23-1-0543. We thank Dakshita Khurana for her assistance in the early stages of the project.

References

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

[AJ15] Prabhanjan Ananth and Abhishek Jain. Indistinguishability obfuscation from compact functional encryption. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 308–326. Springer, Berlin, Heidelberg, August 2015.

[ALL+21] Scott Aaronson, Jiahui Liu, Qipeng Liu, Mark Zhandry, and Ruizhe Zhang. New approaches for quantum copy-protection. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 526–555, Virtual Event, August 2021. Springer, Cham.

[BBV24] James Bartusek, Zvika Brakerski, and Vinod Vaikuntanathan. Quantum state obfuscation from classical oracles. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*, pages 1009–1017. ACM, 2024.

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

[BFM88] Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 103–112, New York, NY, USA, 1988. Association for Computing Machinery.

[BGI+12] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. *J. ACM*, 59(2), May 2012.

[BKNY23] James Bartusek, Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Obfuscation of pseudo-deterministic quantum circuits. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, page 1567–1578, New York, NY, USA, 2023. Association for Computing Machinery.

[BKS23] James Bartusek, Dakshita Khurana, and Akshayaram Srinivasan. Secure computation with shared EPR pairs (or: How to teleport in zero-knowledge). In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part V*, volume 14085 of *LNCS*, pages 224–257. Springer, Cham, August 2023.

[BL08] Jacob D. Biamonte and Peter J. Love. Realizable hamiltonians for universal adiabatic quantum computers. *Phys. Rev. A*, 78:012352, Jul 2008.

[BM22] James Bartusek and Giulio Malavolta. Indistinguishability Obfuscation of Null Quantum Circuits and Applications. In Mark Braverman, editor, *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, volume 215 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 15:1–15:13, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[BV15] Nir Bitansky and Vinod Vaikuntanathan. Indistinguishability obfuscation from functional encryption. In Venkatesan Guruswami, editor, *56th FOCS*, pages 171–190. IEEE Computer Society Press, October 2015.

[ÇGS25] Alper Çakan, Vipul Goyal, and Omri Shmueli. Public-key quantum fire and key-fire from classical oracles. *IACR Cryptol. ePrint Arch.*, page 726, 2025.

[CL01] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques: Advances in Cryptology*, EUROCRYPT '01, page 93–118, Berlin, Heidelberg, 2001. Springer-Verlag.

[CM16] Toby S. Cubitt and Ashley Montanaro. Complexity classification of local hamiltonian problems. *SIAM J. Comput.*, 45(2):268–316, 2016.

[CvH91] David Chaum and Eugène van Heyst. Group signatures. In Donald W. Davies, editor, *Advances in Cryptology — EUROCRYPT '91*, pages 257–265, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

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

[Fis05] Marc Fischlin. Communication-efficient non-interactive proofs of knowledge with online extractors. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, pages 152–168, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[FLS99] Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple noninteractive zero knowledge proofs under general assumptions. *SIAM Journal on Computing*, 29(1):1–28, September 1999.

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

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

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect non-interactive zero knowledge for np. In Serge Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006*, pages 339–358, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[HT25] Mi-Ying (Miryam) Huang and Er-Cheng Tang. Obfuscation of unitary quantum programs. In *66th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2025)*. IEEE, 2025. To appear.

[JJ21] Abhishek Jain and Zhengzhong Jin. Non-interactive zero knowledge from sub-exponential DDH. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I*, volume 12696 of *Lecture Notes in Computer Science*, pages 3–32. Springer, 2021.

[JLLW23] Aayush Jain, Huijia Lin, Ji Luo, and Daniel Wichs. The pseudorandom oracle model and ideal obfuscation. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part IV*, volume 14084 of *Lecture Notes in Computer Science*, pages 233–262. Springer, 2023.

[MNS16] Tomoyuki Morimae, Daniel Nagaj, and Norbert Schuch. Quantum proofs can be verified using only single-qubit measurements. *Physical Review A*, 93(2), February 2016.

[MY22] Tomoyuki Morimae and Takashi Yamakawa. Classically verifiable NIZK for QMA with preprocessing. In Shweta Agrawal and Dongdai Lin, editors, *ASIACRYPT 2022, Part IV*, volume 13794 of *LNCS*, pages 599–627. Springer, Cham, December 2022.

[PS19] Chris Peikert and Sina Shiehian. Noninteractive zero knowledge for NP from (plain) learning with errors. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part I*, volume 11692 of *LNCS*, pages 89–114. Springer, Cham, August 2019.

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

[SW14] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '14, page 475–484, New York, NY, USA, 2014. Association for Computing Machinery.

[Tro15] Joel A. Tropp. An introduction to matrix concentration inequalities. *Found. Trends Mach. Learn.*, 8(1-2):1–230, 2015.

[Unr17] Dominique Unruh. Post-quantum security of Fiat-Shamir. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 65–95. Springer, Cham, December 2017.

[Wat24] Brent Waters. A new approach for non-interactive zero-knowledge from learning with errors. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*, pages 399–410. ACM, 2024.

[Win99] Andreas J. Winter. Coding theorem and strong converse for quantum channels. *IEEE Trans. Inf. Theory*, 45(7):2481–2485, 1999.

[WWW25] Brent Waters, Hoeteck Wee, and David J. Wu. New techniques for preimage sampling: Improved nizks and more from LWE. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology - EUROCRYPT 2025 - 44th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Madrid, Spain, May 4-8, 2025, Proceedings, Part IV*, volume 15604 of *Lecture Notes in Computer Science*, pages 3–33. Springer, 2025.

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

[Zha20] Mark Zhandry. Schrödinger's pirate: How to trace a quantum decoder. In Rafael Pass and Krzysztof Pietrzak, editors, *TCC 2020, Part III*, volume 12552 of *LNCS*, pages 61–91. Springer, Cham, November 2020.

[Zha21] Mark Zhandry. Quantum lightning never strikes the same state twice. or: Quantum money from cryptographic assumptions. *J. Cryptol.*, 34(1), January 2021.

[Zha25] Mark Zhandry. A note on quantum-secure prps. *Quantum*, 9:1696, 2025.