Classical Verification of Quantum Computations with Efficient Verifier

Nai-Hui Chia[1,2], Kai-Min Chung[3], and Takashi Yamakawa[4]

[1] Joint Center for Quantum Information and Computer Science, University of Maryland nchia@umd.edu
[2] Department of Computer Science, University of Texas at Austin
[3] Institute of Information Science, Academia Sinica kmchung@iis.sinica.edu.tw
[4] NTT Secure Platform Laboratories takashi.yamakawa.ga@hco.ntt.co.jp

Abstract. In this paper, we extend the protocol of classical verification of quantum computations (CVQC) recently proposed by Mahadev to make the verification efficient. Our result is obtained in the following three steps:

- We show that parallel repetition of Mahadev's protocol has negligible soundness error. This gives the first constant round CVQC protocol with negligible soundness error. In this part, we only assume the quantum hardness of the learning with error (LWE) problem similar to Mahadev's work.
- We construct a two-round CVQC protocol in the quantum random oracle model (QROM) where a cryptographic hash function is idealized to be a random function. This is obtained by applying the Fiat-Shamir transform to the parallel repetition version of Mahadev's protocol.
- We construct a two-round CVQC protocol with an efficient verifier in the CRS+QRO model where both prover and verifier can access a (classical) common reference string generated by a trusted third party in addition to quantum access to QRO. Specifically, the verifier can verify a $\mathsf{QTIME}(T)$ computation in time $\mathsf{poly}(n, \log T)$ where n is the security parameter. For proving soundness, we assume that a standard model instantiation of our two-round protocol with a concrete hash function (say, SHA-3) is sound and the existence of post-quantum indistinguishability obfuscation and post-quantum fully homomorphic encryption in addition to the quantum hardness of the LWE problem.

1 Introduction

Quantum computers that outperform classical supercomputers have been realized recently [7] and may play a role similar to the super clusters in the foreseeable future. Indeed, this is happening now—IBM has provided an online platform for public users to run their computational tasks on IBM's quantum computing server [1]. Since quantum computers would be accessed by clients with only classical devices, verifying quantum computation by a classical computer has

become a major issue in this setting. To address this problem, there are several works toward reducing the verifier's quantum resource for verifying quantum computation [15,21,5,35]. However, it was unknown if the verifier could be purely classical until Mahadev [32] finally gave an affirmative solution. Specifically, she constructed an interactive protocol between an efficient classical verifier (a BPP machine) and an efficient quantum prover (a BQP machine) where the verifier can verify the result of the BQP computation. (In the following, we call such a protocol a CVQC protocol.[5]) Soundness of her protocol relies on a computational assumption that the learning with error (LWE) problem [40] is hard for an efficient quantum algorithm, which has been widely used in the field of cryptography. We refer to the extensive survey by Peikert [38] for details about LWE and its cryptographic applications.

Although the verifier in Mahadev's protocol is purely classical, it is not "efficient". In the classical cryptographic literature of delegating (classical) computation, efficient verifier that can verify a delegated time T computation in $o(T)$ time is a necessary requirement (as otherwise, the verifier performs the computation on its own). Indeed, many previous works suggested that the verifier's runtime can be $\mathsf{poly}\log(T)$ in the classical setting [30,34,28,29,25,42,14,9,26,16,27]. In contrast, in the literature of delegating quantum computation, the focus is mainly on reducing the required quantum power for the verifier, and all existing protocols with a single prover (e.g., in blind quantum computation [15] and Mahadev's protocol [32]) inherently requires the verifier to run in $poly(T)$ time to verify the delegated computation, even for verifiers with weak quantum power.

Therefore, whether a CVQC protocol with an efficient verifier (i.e., with runtime $o(T)$) exists is a natural and fundamental theoretical question. Also, from a technical perspective, classical efficient verifier protocols are closely related to PCP proofs, where many protocols are constructed based on PCP proofs, and a partial converse result is proven by Rothblum and Vadhan [43]. On the other hand, whether a quantum version of the PCP theorem holds is still an open question in quantum complexity theory [4]. Thus, the challenge of constructing a protocol with an efficient verifier is potentially related to the challenge of constructing quantum PCP proofs. While our construction relies on several strong and non-standard assumptions, our protocol provides the first feasibility result (in any reasonable models) that answers this question of efficient verifier CVQC protocol affirmatively.

1.1 Our Results

In this paper, our main result is a CVQC protocol with an efficient verifier, and we have also reached two milestones on the path to the final result. We summarize them as follows:

Parallel repetition of Mahadev's protocol We first show that parallel repetition version of Mahadev's protocol has negligible soundness error. Note that Mahadev's protocol has soundness error 3/4, which means that a cheating prover

[5] "CVQC" stands for "Classical Verification of Quantum Computations"

may convince the verifier even if it does not correctly computes the BQP computation with probability at most 3/4. Though we can exponentially reduce the soundness error by sequential repetition, we need super-constant rounds to reduce the soundness error to be negligible. If parallel repetition works to reduce the soundness error, then we need not increase the number of round. However, parallel repetition may not reduce soundness error for computationally sound protocols in general [11,39]. Thus, it was open to construct constant round protocol with negligible soundness error. We manage to answer this question by giving the first constant round CVQC protocol with negligible soundness error.

Two-round CVQC protocol Based on the parallel repetition version of Mahadev's protocol with negligible soundness, we then construct a two-round CVQC protocol in the quantum random oracle model (QROM) [12] where a cryptographic hash function is idealized to be a random function that is only accessible as a quantum oracle. This is obtained by applying the Fiat-Shamir transform [20,31,19] to the parallel repetition version of Mahadev's protocol.

CVQC protocol with an efficient verifier Finally, we construct a two-round CVQC protocol with logarithmic-time verifier in the CRS+QRO model where both prover and verifier can access to a (classical) common reference string generated by a trusted third party in addition to quantum access to QRO. For proving soundness, we assume that a standard model instantiation of our two-round protocol with a concrete hash function (say, SHA-3) is sound and the existence of post-quantum indistinguishability obfuscation [10,22] and (post-quantum) fully homomorphic encryption (FHE) [23] in addition to the quantum hardness of the LWE problem.

1.2 Technical Overview

Overview of Mahadev's protocol. First, we recall the high-level structure of Mahadev's 4-round CVQC protocol.[6] On input a common input x, a quantum prover and classical verifier proceeds as below to prove and verify that x belongs to a BQP language L.

First Message: The verifier generates a pair of "key" k and a "trapdoor" td, sends k to the prover, and keeps td as its internal state.

Second Message: The prover is given the key k, generates a classical "commitment" y along with a quantum state $|\mathsf{st}_P\rangle$, sends y to the verifier, and keeps $|\mathsf{st}_P\rangle$ as its internal state.

Third Message: The verifier randomly picks a "challenge" $c \xleftarrow{\$} \{0,1\}$ and sends c to the prover. Following the terminology in [32], we call the case of $c = 0$ the "test round" and the case of $c = 1$ the "Hadamard round".

Fourth Message: The prover is given a challenge c, generates a classical "answer" a by using the state $|\mathsf{st}_P\rangle$, and sends a to the verifier.

[6] See Sec. 3.1 for more details.

Final Verification: Finally, the verifier returns $\top$ indicating acceptance or $\bot$ indicating rejection. In case $c = 0$, the verification can be done publicly, that is, the final verification algorithm need not use td.

Mahadev showed that the protocol achieves negligible completeness error and constant soundness error against computationally bounded cheating provers. More precisely, she showed that if $x \in L$, then the verifier accepts with probability $1-\mathsf{negl}(n)$ where n is the security parameter, and if $x \notin L$, then any quantum polynomial time cheating prover can let the verifier accept with probability at most $3/4$. For proving this, she first showed the following lemma:[7]

Lemma 1 (informal). *For any $x \notin L$, if a quantum polynomial time cheating prover passes the test round with probability $1 - \mathsf{negl}(n)$, then it passes the Hadamard with probability $\mathsf{negl}(n)$ assuming the quantum hardness of the LWE problem.*

Given the above lemma, it is easy to prove the soundness of the protocol. Roughly speaking, we consider a decomposition of the Hilbert space $\mathcal{H}_P$ for the prover's internal state $|\psi_P\rangle$ into two subspaces S_0 and S_1 so that S_0 (resp. S_1) consists of quantum states that lead to rejection (resp. acceptance) in the test round. That is, we define these subspaces so that if the cheating prover's internal state after sending the second message is $|s_0\rangle \in S_0$ (resp. $|s_1\rangle \in S_1$), then the verifier returns rejection (acceptance) in the test round (i.e., the case of $c = 0$). Here, we note that the decomposition is well-defined since we can assume that a cheating prover just applies a fixed unitary on its internal space and measures some registers for generating the fourth message in the test round without loss of generality. Let Π_b be the projection onto S_b and $|\psi_b\rangle := \Pi_b |\psi_P\rangle$ for $b \in \{0, 1\}$. Then $|\psi_0\rangle$ leads to rejection in the test round (with probability 1), so if the verifier uniformly chooses $c \xleftarrow{\$} \{0, 1\}$, then $|\psi_0\rangle$ leads to acceptance with probability at most $1/2$. On the other hand, since $|\psi_1\rangle$ leads to the acceptance in the test round (with probability 1), by Lemma 1, $|\psi_1\rangle$ leads to the acceptance in the Hadamard round with only negligible probability. Therefore, the verifier uniformly chooses $c \xleftarrow{\$} \{0, 1\}$, then $|\psi_1\rangle$ leads to acceptance with probability at most $1/2 + \mathsf{negl}(n)$. Therefore, intuitively speaking, $|\psi_P\rangle = |\psi_0\rangle + |\psi_1\rangle$ leads to acceptance with probability at most $1/2 + \mathsf{negl}(n)$, which completes the proof of soundness. We remark that here is a small gap since measurements are not linear and thus we cannot simply conclude that $|\psi_P\rangle$ leads to acceptance with probability at most $1/2 + \mathsf{negl}(n)$ even though the same property holds for both $|\psi_0\rangle$ and $|\psi_1\rangle$. Indeed, Mahadev just showed that the soundness error is at most $3/4$ instead of $1/2 + \mathsf{negl}(n)$ to deal with this issue. A concurrent work by Alagic et al. [6] proved that the Mahadev's protocol actually achieves soundness error $1/2 + \mathsf{negl}(n)$ with more careful analysis.

[7] Strictly speaking, she just proved a similar property for what is called a "measurement protocol" instead of CVQC protocol. But this easily implies a similar statement for CVQC protocol since CVQC protocol can be obtained by combining a measurement protocol and the (amplified version of) Morimae-Fitzsimons protocol [35] without affecting the soundness error as is done in [32, Section 8].

Parallel repetition. Now, we turn our attention to parallel repetition version of Mahadev's protocol. Our goal is to prove that the probability that the verifier accepts on $x \notin L$ is negligible if the verifier and prover run the Mahadev's protocol m-times parallelly for sufficiently large m and the verifier accepts if and only if it accepts on the all coordinates.

Our first step is to consider a decomposition of the prover's space $\mathcal{H}_P$ into two subspaces $S_{i,0}$ and $S_{i,1}$ for each $i \in [m]$ similarly to the stand-alone case. Specifically, we want to define these subspaces so that $S_{i,0}$ (resp. $S_{i,1}$) consists of quantum states that lead to rejection (resp. acceptance) in the test round on the i-th coordinate. However, such subspaces are not well-defined since a cheating prover's behavior in the fourth round depends on challenges $c = c_1....c_m \in \{0,1\}^m$ on all coordinates. Thus, even if we focus on the test round on the i-th coordinate, all other challenges $c_{-i} = c_1...c_{i-1}c_{i+1}...c_m$ still have flexibility, and a different choice of c_{-i} leads to a different prover's behavior. In other words, the prover's strategy should be described as a unitary over $\mathcal{H}_{\mathbf{C}} \otimes \mathcal{H}_P$ where $\mathcal{H}_{\mathbf{C}}$ is a Hilbert space to store a challenge. Therefore $S_{i,0}$ and $S_{i,1}$ cannot be well-defined as a decomposition of $\mathcal{H}_P$ if we define them as above.

Therefore, we need to define these subspaces in a little different way. Specifically, our idea is to define them as subspaces that "know" and "do not know" an answer for the test round on i-th coordinate. More precisely, for any fixed noticeable "threshold" $\gamma = 1/\mathsf{poly}(n)$, we ideally require the followings:

1. ($S_{i,0}$ **"does not know" an answer.**) If the fourth message generation algorithm of the cheating prover runs with an internal state $|\psi_{i,0}\rangle \in S_{i,0}$, then it passes the test round on i-th coordinate with probability at most γ when the challenge c is uniformly chosen from $\{0,1\}^m$ such that $c_i = 0$.
2. ($S_{i,1}$ **"knows" an answer.**) There is an efficient algorithm that is given any $|\psi_{i,1}\rangle \in S_{i,1}$ as input and outputs an accepting answer for the test round on i-th coordinate with overwhelming probability.
3. (**Efficient projection.**) A measurement described by $\{\Pi_{S_{i,0}}, \Pi_{S_{i,1}}\}$ can be performed efficiently where $\Pi_{S_{i,0}}$ and $\Pi_{S_{i,1}}$ denote projections to $S_{i,0}$ and $S_{i,1}$, respectively.

Unfortunately, we do not know how to achieve these requirements in the above clean form. Nonetheless, we can show that a "noisy" version of the above requirements can be achieved by using the techniques taken from works on an amplification theorem for QMA [33,36]. We will explain this in more detail in the next paragraph since this is the technical core of our proof. In the rest of this paragraph, we explain how to prove the soundness of the parallel repetition version of Mahadev's protocol assuming that the above requirements are satisfied in the clean form as above for simplicity. Here, we observe that for any $i \in [m]$ and $b \in \{0,1\}$, any efficiently generated $|\psi_{i,b}\rangle \in S_{i,b}$ leads to acceptance in the verification on i-th coordinate for any fixed c such that $c_i = b$ with probability at most $2^{m-1}\gamma + \mathsf{negl}(n)$. This can be seen by a similar argument to the stand-alone case: The case of $b = 0$ follows from the above requirement 1 considering that the number of $c \in \{0,1\}^m$ such that $c_i = 0$ is 2^{m-1}. The case of $b = 1$ follows

from the above requirement 2 combined with Lemma 1 assuming the quantum hardness of LWE.

Our next step is to sequentially apply projections onto $S_{i,0}$ and $S_{i,1}$ for $i = 1, ..., m$ to further decompose the prover's state $|\psi_P\rangle$. More precisely, for any fixed $c = c_1...c_m \in \{0,1\}^m$, we define

$$|\psi_0\rangle := \Pi_{S_{1,0}} |\psi_P\rangle, \qquad |\psi_1\rangle := \Pi_{S_{1,1}} |\psi_P\rangle$$

and

$$|\psi_{\bar{c}_1,...,\bar{c}_{i-1},0}\rangle := \Pi_{S_{i,0}} |\psi_{\bar{c}_1,...,\bar{c}_{i-1}}\rangle, \qquad |\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle := \Pi_{S_{i,1}} |\psi_{\bar{c}_1,...,\bar{c}_{i-1}}\rangle$$

for $i = 2, ..., m$ where $\bar{c}_i$ denotes $1 - c_i$. Then we have

$$|\psi\rangle = |\psi_{c_1}\rangle + |\psi_{\bar{c}_1,c_2}\rangle + \cdots + |\psi_{\bar{c}_1,...,\bar{c}_{m-1},c_m}\rangle + |\psi_{\bar{c}_1,...,\bar{c}_m}\rangle .$$

Here, for each $i \in [m]$, we have $|\psi_{\bar{c}_1,...,\bar{c}_{i-1},c_i}\rangle \in S_{i,c_i}$ by definition. Therefore, $|\psi_{\bar{c}_1,...,\bar{c}_{i-1},c_i}\rangle$ leads to acceptance on the verification on i-th coordinate with probability at most $2^{m-1}\gamma + \mathsf{negl}(n)$ when the challenge is c. Moreover, if we consider the above decomposition for a randomly chosen c, then we have $E_{c \xleftarrow{\$} \{0,1\}^m}[\| |\psi_{\bar{c}_1,...,\bar{c}_m}\rangle \|] \leq 2^{-m}$ since an expected norm is halved whenever we apply either of projections onto $S_{i,0}$ or $S_{i,1}$ randomly. Therefore, we can conclude that the verifier accepts on the all coordinates with probability at most $2^{m-1}\gamma + 2^{-m} + \mathsf{negl}(n)$. This is not negligible since we need to assume that γ is noticeable due to a technical reason. However, we can make $2^{m-1}\gamma + 2^{-m} + \mathsf{negl}(n)$ as small as any noticeable function by appropriately setting $m = O(\log n)$ and $\gamma = 1/\mathsf{poly}(n)$. This implies that a cheating adversary's winning probability is $\mathsf{negl}(n)$ if we set $m = \omega(\log n)$.

How to define $S_{i,0}$ and $S_{i,1}$. In this paragraph, we explain how to define subspaces $S_{i,0}$ and $S_{i,1}$ and achieve a noisy version of the requirements in the previous paragraph. For defining these subspaces, we borrow a lemma from [36], which was originally used for proving an amplification theorem for QMA. Since their lemma is a little complicated to state in a general form, we only explain what is ensured by their lemma in our context. In our context, their lemma ensures that there is an efficient operator Q over $\mathcal{H}_{\mathbf{C}} \times \mathcal{H}_P$ where $\mathcal{H}_{\mathbf{C}}$ is a register for storing a challenge $c \in \{0,1\}^m$ such that

1. (Eigenvectors span $\mathcal{H}_P$.) there is a orthonormal basis $\{|\hat{\alpha}_j\rangle\}_j$ of $\mathcal{H}_P$ such that $|0^m\rangle_{\mathbf{C}} |\hat{\alpha}_j\rangle_P$ is an eigenvector of Q with eigenvalue $e^{i\theta_j}$ for some θ_j,
2. (Eigenvalue corresponds to success probability.) if the fourth message generation algorithm of the cheating prover runs with an internal state $|\hat{\alpha}_j\rangle$, then it passes the test round on i-th coordinate with probability $p_j := \cos^2(\theta_j/2)$ when the challenge c is uniformly chosen from $\{0,1\}^m$ such that $c_i = 0$, and
3. (Extractable) there is an extraction algorithm that is given a state $|\hat{\alpha}_j\rangle$ and outputs an accepting answer for the test round on i-th coordinate with overwhelming probability in time $\mathsf{poly}(n, p_j^{-1})$.

Given this lemma, our rough idea is to define $S_{i,0}$ (resp. $S_{i,1}$) as a subspace spanned by $|\hat{\alpha}_j\rangle$ such that $p_j \leq \gamma$ (resp. $p_j > \gamma$). Then, it is easy to see that $S_{i,0}$ and $S_{i,1}$ satisfy the requirements 1 and 2 (i.e., $S_{i,0}$ "does not know" an answer and $S_{i,1}$ "knows" an answer). However, we do not know how to efficiently perform a projection onto $S_{i,0}$ or $S_{i,1}$ since there is no known efficient algorithm for phase estimation without an approximation error. On the other hand, we can efficiently approximate a phase with an approximation error $1/\mathsf{poly}(n)$ [36]. Then, our next idea is to introduce an inverse polynomial gap between thresholds for $S_{i,0}$ and $S_{i,1}$, i.e., we define $S_{i,0}$ (resp. $S_{i,1}$) as a subspace spanned by $|\hat{\alpha}_j\rangle$ such that $p_j \leq \gamma$ (resp. $p_j \geq \gamma + 1/\mathsf{poly}(n)$). Then, we can efficiently perform a projection to $S_{i,0}$ or $S_{i,1}$ by using the phase estimation algorithm with an approximation error $1/\mathsf{poly}(n)$ if the original state does not have a "grey area", which is a space spanned by $|\hat{\alpha}_j\rangle$ such that $p_j \in (\gamma, \gamma + 1/\mathsf{poly}(n))$. However, it may be the case that the original state is dominated by the grey area. To resolve this issue, we randomly set the threshold γ from T possible choices so that we can upper bound the expected norm of the grey area component by $O(1/T)$. In the main body, we formalize this "noisy" version of the decomposition and show that this suffices for proving the soundness of parallel repetition version of Mahadev's protocol.

Remark 1. We remark that parallel repetition of Mahadev's protocol is also analyzed in a concurrent work of Alagic et al. [6], who gave an elegant analysis. Their analysis starts from the same observation (Lemma 1) but is interestingly different from ours (see Section 1.3 for further discussion). An advantage of our analysis is that it is more constructive. Namely, we show that the ("noisy" version of) projection to $S_{i,0}$ and $S_{i,1}$ can be constructed efficiently. This is a useful feature that has found application in the work of [18], who constructed CVQC protocols for quantum sampling problems. They used the technique developed here to analyze parallel repetition of their protocol (while the analysis of [6] does not seem to generalize).

Two-round protocol via Fiat-Shamir transform. Here, we explain how to convert the parallel repetition version of Mahadev's protocol to a two-round protocol in the QROM. First, we observe that the third message of the Mahadev's protocol is public-coin, and thus the parallel repetition version also satisfies this property. Then by using the Fiat-Shamir transform [20], we can replace the third message with hash value of the transcript up to the second round. Though the Fiat-Shamir transform was originally proven sound only in the classical ROM, recent works [31,19] showed that it is also sound in the QROM. This enables us to apply the Fiat-Shamir transform to the parallel repetition version of Mahadev's protocol to obtain a two-round protocol in the QROM.

Making verification efficient. Finally, we explain how to make the verification efficient. Our idea is to delegate the verification procedure itself to the prover by using delegation algorithm for classical computation. Since the verification is classical, this seems to work at first glance. However, there are the following two problems:

1. There is not a succinct description of the verification procedure since the verification procedure is specified by the whole transcript whose size is $\mathsf{poly}(T)$ when verifying a language in $\mathsf{QTIME}(T)$. Then the verifier cannot specify the verification procedure to delegate within time $O(\log(T))$.
2. Since the CVQC protocol is not publicly verifiable (i.e., verification requires a secret information that is not given to the prover), the prover cannot know the description of the verification procedure, which is supposed to be delegated to the prover.

We solve the first problem by using a succinct randomized encoding, which enables one to generate a succinct encoding of a Turing machine M and an input x so that the encoding only reveals the information about $M(x)$ and not M or x. Then our idea is that instead of sending the original first message, the verifier just sends a succinct encoding of (V_1, s) where V_1 denotes the Turing machine that takes s as input and works as the first-message-generation algorithm of the CVQC protocol with randomness $PRG(s)$ where PRG is a pseudorandom number generator. This enables us to make the transcript of the protocol succinct (i.e., the description size is logarithmic in T) so that the verifier can specify the verification procedure succinctly. To be more precise, we have to use a strong output-compressing randomized encoding [8], where the encoding size is independent of the output length of the Turing machine. They construct a strong output-compressing randomized encoding based on iO and other mild assumptions in the common reference string. Therefore our CVQC protocol also needs the common reference string.

We solve the second problem by using FHE. Namely, the verifier sends an encryption of the trapdoor td by FHE, and the prover performs the verification procedure over the ciphertext and provides a proof that it honestly applied the homomorphic evaluation by SNARK. Then the verifier decrypts the resulting FHE ciphertext and accepts if the decryption result is "accept" and the SNARK proof is valid.

In the following, we describe (a simplified version of) our construction. Suppose that we have a 2-round CVQC that works as follows:

First message: Given an instance x, the verifier generates a pair (k, td) of a"key" and "trapdoor", sends k to P, and keeps td as its internal state.

Second message: Given x and k, the prover generates a response e and sends it to the verifier.

Verification: Given x, k, td, e, the verifier returns $\top$ indicating acceptance or $\bot$ indicating rejection.

Then we construct a CVQC protocol with efficient verification as follows.

Setup: It generates a CRS for a strong output-compressing randomized encoding.

First Message: Given a CRS and an instance x, the verifier picks a seed s for PRG and a public and secret keys $(\mathsf{pk}_{\mathsf{fhe}}, \mathsf{sk}_{\mathsf{fhe}})$ of FHE, computes $\mathsf{ct} \xleftarrow{\$} \mathsf{FHE.Enc}(\mathsf{pk}_{\mathsf{fhe}}, s)$ and generates a succinct encoding $\widehat{M_{\mathsf{inp}}}$ of $M(s)$ where M is a classical Turing machine that works as follows:

$M(s)$**:** Given a seed s for PRG, it generates (k, td) as in the building block CVQC protocol by using a randomness $PRG(s)$ and outputs k.

Then the verifier sends $(\widehat{M_{\mathsf{inp}}}, \mathsf{pk}_{\mathsf{fhe}}, \mathsf{ct})$ to the prover and keeps $\mathsf{sk}_{\mathsf{fhe}}$ as its internal state.

Second Message: The prover obtains k by decoding $\widehat{M_{\mathsf{inp}}}$, computes e as in the building block CVQC protocol, and homomorphically evaluates a classical circuit $C[x, e]$ on ct to generate ct' where $C[x, e]$ is a circuit that works as follows:

$C[x, e](s)$**:** Given a seed s for PRG, it generates (k, td) as in the building block CVQC protocol by using a randomness $PRG(s)$ and returns 1 if and only if e is an accepting answer in the building block CVQC w.r.t. x and (k, td).

Then the prover generates a SNARK proof π_{snark} that proves that there exists e' such that ct' is a result of a homomorphic evaluation of the circuit $C[x, e]$ on ct. Then it sends $(\mathsf{ct}', \pi_{\mathsf{snark}})$ to the verifier

Verification: The verifier accepts if the decryption result of ct' is 1 and π_{snark} passes the verification of SNARK.

Intuitively, the soundness of the above protocol can be proven by considering the following hybrids. In the first hybrid, the verifier extracts the witness e' from π_{snark} by using the extractability of SNARK and runs the original verification of the building block CVQC on the second message e' instead of checking if the decryption result of ct' is 1. This decreases the cheating prover's success probability by a factor of $\mathsf{poly}(n)$ since the extraction succeeds with probability $1/\mathsf{poly}(n)$ and if the extraction succeeds, the verifier's output should be the same. In the next hybrid, we change ct to an encryption of $0^{|s|}$ instaed of s. Since the verifier no longer uses $\mathsf{sk}_{\mathsf{fhe}}$, this hybrid is indistinguishable from the previous one by the CPA security of FHE. In the next hybrid, we generate $\widehat{M_{\mathsf{inp}}}$ by a simulation algorithm of the strong output-compressing randomized encoding from $M(s) = k$. This hybrid is indistinguishable from the previous one by the security of the strong output-compressing randomized encoding. In the next hybrid, we replace k that is used as an input of the simulation algorithm of the strong output-compressing randomized encoding with a one generated with a true randomness instead of $PRG(s)$. This hybrid is indistinguishable from the previous one by the security of PRG noting that s is no longer used for generating ct. In this final hybrid, a cheating prover is essentially only given k and has no information about td, and it wins if and only if the extraction algorithm of SNARK extracts an accepting second message e' of the building block CVQC. Thus, the winning probability in the final hybrid is negligible due to the soundness of the building block CVQC. Therfore the above efficient verification version is also sound.

Though the above proof sketch can be made rigorous if we assume adaptive extractability for SNARK, we want to instantiate SNARK in the QROM [17], which is only proven to have non-adaptive extractability. Specifically, it only ensures the extractability in the setting where the statement is chosen before making any query to the random oracle. To deal with this issue, we first expand

the protocol to the four-round protocol where the verifier randomly sends a "salt" z, which is a random string of a certain length, in the third round and the prover uses the "salted" random oracle $H(z, \cdot)$ for generating the SNARK proof. Since the statement to be proven by SNARK is determined up to the second round, and the salting essentially makes the random oracle "fresh", we can argue the soundness of the CVQC protocol even with the non-adaptive extractability of the SNARK. At this point, we obtain four-round CVQC protocol with efficient verification. Here, we observe that the third message is just a salt z, which is public-coin. Therefore we can just apply the Fiat-Shamir transform again to make the protocol two-round.

1.3 Related Works

Verification of Quantum Computation. There is a long line of researches on verification of quantum computation. Except for solutions relying on computational assumptions, there are two type of settings where verification of quantum computation is known to be possible. In the first setting, instead of considering purely classical verifier, we assume that a verifier can perform a certain kind of weak quantum computations [15,21,5,35]. In the second setting, we assume that a prover is splitted into two remote servers that share entanglement but do not communicate [41]. Though these works do not give a CVQC protocol in our sense, the advantage is that we need not assume any computational assumption for the proof of soundness, and thus they are incomparable to Mahadev's result and ours.

Subsequent to Mahadev's breakthrough result, Gheorghiu and Vidick [24] gave a CVQC protocol that also satisfies blindness, which ensures that a prover cannot learn what computation is delegated. We note that their protocol requires polynomial number of rounds.

Post-Quantum Indistinguishability Obfuscation. There are several candidates of post-quantum indistniguishability obfuscation [2,3,13,44]. Especially, the recent work by Brakerski et al. [13] gave a construction of indistniguishability obfuscation based on the LWE assumption and a certain type of circular security of LWE-based encryption schemes against subexponential time adversaries.

Concurrent Work. In a concurrent and independent work, Alagic et al. [6] also shows similar results to our first and second results, parallel repetition theorem for the Madadev's protocol and a two-round CVQC protocol by the Fiat-Shamir transform. We note that our third result, a two-round CVQC protocol with efficient verification, is unique in this paper. On the other hand, they also give a construction of non-interactive zero-knowledge arguments for QMA, which is not given in this paper.

We mention that we have learned the problem of parallel repetition for Mahadev's protocol from the authors of [6] on March 2019, but investigated the problem independently later as a stepping stone toward making the verifier efficient. Interestingly, the analyses of parallel repetition in the two works are quite

different. Briefly, the analysis in [6] relies on the observation that for any two different challenges $c_1 \neq c_2 \in \{0,1\}^m$, the projections of an efficient-generated prover's state on the accepting subspaces corresponding to c_1 and c_2 are almost orthogonal, which leads to an elegant proof of the parallel repetition theorem.

As mentioned, we additionally show that the projections can be approximated "efficiently" by constructing an efficient quantum procedure (Lemma 4). This is the main technical step in our proof, where we combine several tools such as Jordan's lemma, phase estimation, and random thresholding to construct the efficient projector. We then use this efficient projector iteratively to bound the success probability of the prover. Our construction of the efficient projection has found applications in a related context in [18].

2 Preliminaries

Notations. For a bit $b \in \{0,1\}$, $\bar{b}$ denotes $1-b$. For a finite set $\mathcal{X}$, $x \xleftarrow{\$} \mathcal{X}$ means that x is uniformly chosen from $\mathcal{X}$. For finite sets $\mathcal{X}$ and $\mathcal{Y}$, $\mathsf{Func}(\mathcal{X},\mathcal{Y})$ denotes the set of all functions with domain $\mathcal{X}$ and range $\mathcal{Y}$. A function $f : \mathbb{N} \to [0,1]$ is said to be negligible if for all polynomial p and sufficiently large $n \in \mathbb{N}$, we have $f(n) < 1/p(n)$ and said to be overwhelming if $1-f$ is negligible. We denote by poly an unspecified polynomial and by negl an unspecified negligible function. We say that a classical (resp. quantum) algorithm is efficient if it runs in probabilistic polynomial-time (resp. quantum polynominal time). For a quantum or randomized algorithm $\mathcal{A}$, $y \xleftarrow{\$} \mathcal{A}(x)$ means that $\mathcal{A}$ is run on input x and outputs y and $y := \mathcal{A}(x;r)$ means that $\mathcal{A}$ is run on input x and randomness r and outputs y. For an interactive protocol between a "prover" P and "verifier" V, $y \xleftarrow{\$} \langle P(x_P), V(x_V))\rangle(x)$ means an interaction between them with prover's private input x_P verifier's private input x_V, and common input x outputs y. For a quantum state $|\psi\rangle$, $M_{\mathbf{X}} \circ |\psi\rangle$ means a measurement in the computational basis on the register $\mathbf{X}$ of $|\psi\rangle$. We denote by $\mathsf{QTIME}(T)$ a class of languages decided by a quantum algorithm whose running time is at most T. We use n to denote the security parameter throughout the paper.

2.1 Learning with Error Problem

Roughly speaking, the learning with error (LWE) is a problem to solve system of noisy linear equations. Regev [40] proved that the hardness of LWE can be reduced to hardness of certain worst-case lattice problems via quantum reductions. We do not give a definition of LWE in this paper since we use the hardness of LWE only for ensuring the soundness of the Mahadev's protocol (Lemma 3), which is used as a black-box manner in the rest of the paper. Therefore, we use exactly the same assumption as that used in [32], to which we refer for detailed definitions and parameter settings for LWE.

2.2 Quantum Random Oracle Model

The quantum random oracle model (QROM) [12] is an idealized model where a real-world hash function is modeled as a quantum oracle that computes a random function. More precisely, in the QROM, a random function $H : \mathcal{X} \to \mathcal{Y}$ of a certain domain $\mathcal{X}$ and range $\mathcal{Y}$ is uniformly chosen from $\mathsf{Func}(\mathcal{X}, \mathcal{Y})$ at the beginning, and every party (including an adversary) can access to a quantum oracle O_H that maps $|x\rangle |y\rangle$ to $|x\rangle |y \oplus H(x)\rangle$. We often abuse notation to denote $\mathcal{A}^H$ to mean a quantum algorithm $\mathcal{A}$ is given oracle O_H.

2.3 Lemma

Here, we give a simple lemma, which is used in the proof of soundness of parallel repetition version of the Mahadev's protocol in Sec. 3.3.

Lemma 2. *Let $|\psi\rangle = \sum_{i=1}^{m} |\psi_i\rangle$ be a quantum state and M be a projective measurement. Then we have*

$$\Pr[M \circ |\psi\rangle = 1] \leq m \sum_{i=1}^{m} \| |\psi_i\rangle \|^2 \Pr\left[M \circ \frac{|\psi_i\rangle}{\| |\psi_i\rangle \|} = 1\right]$$

A proof can be found in the full version.

3 Parallel Repetition of Mahadev's Protocol

3.1 Overview of Mahadev's Protocol

Here, we recall Mahadev's protocol [32]. We only give a high-level description of the protocol and properties of it and omit the details since they are not needed to show our result.

The protocol is run between a quantum prover P and a classical verifier V on a common input x. The aim of the protocol is to enable a verifier to classically verify $x \in L$ for a BQP language L with the help of interactions with a quantum prover. The protocol is a 4-round protocol where the first message is sent from V to P. We denote the i-th message generation algorithm by V_i for $i \in \{1, 3\}$ or P_i for $i \in \{2, 4\}$ and denote the verifier's final decision algorithm by V_{out}. Then a high-level description of the protocol is given below.

V_1: On input the security parameter 1^n and x, it generates a pair (k, td) of a"key" and "trapdoor", sends k to P, and keeps td as its internal state.

P_2: On input x and k, it generates a classical "commitment" y along with a quantum state $|\mathsf{st}_P\rangle$, sends y to P, and keeps $|\mathsf{st}_P\rangle$ as its internal state.

V_3: It randomly picks a "challenge" $c \xleftarrow{\$} \{0, 1\}$ and sends c to P.[8] Following the terminology in [32], we call the case of $c = 0$ the "test round" and the case of $c = 1$ the "Hadamard round".

[8] The third message is just a public-coin, and does not depend on the transcript so far or x.

P_4**:** On input $|\mathsf{st}_P\rangle$ and c, it generates a classical "answer" a and sends a to P.
V_{out}**:** On input k, td, y, c, and a, it returns $\top$ indicating acceptance or $\bot$ indicating rejection. In case $c = 0$, the verification can be done publicly, that is, V_{out} need not take td as input.

For the protocol, we have the following properties:
Completeness: For all $x \in L$, we have $\Pr[\langle P, V\rangle(x)] = \bot] = \mathsf{negl}(n)$.
Soundness: If the LWE problem is hard for quantum polynomial-time algorithms, then for any $x \notin L$ and a quantum polynomial-time cheating prover P^*, we have $\Pr[\langle P^*, V\rangle(x)] = \bot] \leq 3/4$.

We need a slightly different form of soundness implicitly shown in [32], which roughly says that if a cheating prover can pass the "test round" (i.e., the case of $c = 0$) with overwhelming probability, then it can pass the "Hadamard round" (i.e., the case of $c = 1$) only with a negligible probability.

Lemma 3 (implicit in [32]). *If the LWE problem is hard for quantum polynomial-time algorithms, then for any $x \notin L$ and a quantum polynomial-time cheating prover P^* such that $\Pr[\langle P^*, V\rangle(x)] = \bot \mid c = 0] = \mathsf{negl}(n)$, we have $\Pr[\langle P^*, V\rangle(x)] = \top \mid c = 1] = \mathsf{negl}(n)$.*

We will also use the following simple fact:

Fact 1 *There exists an efficient prover that passes the test round with probability 1 (but passes the Hadamard round with probability 0) even if $x \notin L$.*

3.2 Parallel Repetition

Here, we prove that the parallel repetition of Mahadev's protocol decrease the soundness bound to be negligible. Let P^m and V^m be m-parallel repetitions of the honest prover P and verifier V in Mahadev's protocol. Then we have the following:

Theorem 1 (Completeness). *For all $m = \Omega(\log^2(n))$, for all $x \in L$, we have $\Pr[\langle P^m, V^m\rangle(x)] = \bot] = \mathsf{negl}(n)$.*

Theorem 2 (Soundness). *For all $m = \Omega(\log^2(n))$, if the LWE problem is hard for quantum polynomial-time algorithms, then for any $x \notin L$ and a quantum polynomial-time cheating prover P^*, we have $\Pr[\langle P^*, V^m\rangle(x)] = \top] \leq \mathsf{negl}(n)$.*

The completeness (Theorem 1) easily follows from the completeness of Mahadev's protocol. In the next subsection, we prove the soundness (Theorem 2).

3.3 Proof of Soundness

First, we remark that it suffices to show that for any $\mu = 1/\mathsf{poly}(n)$, there exists $m = O(\log(n))$ such that the success probability of the cheating prover is at most μ. This is because we are considering $\omega(\log(n))$-parallel repetition, in which case

the number of repetitions is larger than any $m = O(\log(n))$ for sufficiently large n, and thus we can just focus on the first m coordinates ignoring the rest of the coordinates. Thus, we prove the above claim in this section.

Characterization of cheating prover. Any cheating prover can be characterized by a tuple (U_0, U) of unitaries over Hilbert space $\mathcal{H}_{\mathbf{C}} \otimes \mathcal{H}_{\mathbf{X}} \otimes \mathcal{H}_{\mathbf{Z}} \otimes \mathcal{H}_{\mathbf{Y}} \otimes \mathcal{H}_{\mathbf{K}}$.[9] A prover characterized by (U_0, U) works as follows.[10]

Second Message: Upon receiving $k = (k_1, ..., k_m)$, it applies U_0 to the state $|0\rangle_{\mathbf{X}} \otimes |0\rangle_{\mathbf{Z}} \otimes |0\rangle_{\mathbf{Y}} \otimes |k\rangle_{\mathbf{K}}$, and then measures the Y register to obtain $y = (y_1, ..., y_m)$. Then it sends $\mathbf{y}$ to V and keeps the resulting state $|\psi(k, y)\rangle_{\mathbf{X},\mathbf{Z}}$ over $\mathcal{H}_{\mathbf{X},\mathbf{Z}}$.

fourth Message: Upon receiving $c \in \{0,1\}^m$, it applies U to $|c\rangle_{\mathbf{C}} |\psi(k, y)\rangle_{\mathbf{X},\mathbf{Z}}$ and then measures the $\mathbf{X}$ register in computational basis to obtain $a = (a_1, ..., a_m)$. We denote the designated register for a_i by $\mathbf{X}_i$.

For each $i \in [m]$, we denote by Acc_{k_i,y_i} the set of a_i such that the verifier accepts a_i in the test round on the i-th coordinate when the first and second messages are k_i and y_i, respectively. Note that one can efficiently check if $a_i \in \mathsf{Acc}_{k_i,y_i}$ without knowing the trapdoor behind k_i since verification in the test round can be done publicly as explained in Sec. 3.1.

We first give ideas about Lemma 4 that is the main lemma for this section. For each coordinate $i \in [m]$, we would like to decompose the space $\mathcal{H}_{\mathbf{X},\mathbf{Z}}$ into a subspace $S_{i,0}$ that "does not know" $a_i \in \mathsf{Acc}_{k_i,y_i}$ and a subspace $S_{i,1}$ that "knows" $a_i \in \mathsf{Acc}_{k_i,y_i}$. Ideally, we want to prove the following statement: For any $i \in [m]$ and $|\psi\rangle \in \mathcal{H}_{\mathbf{X},\mathbf{Z}}$, if we decompose it as

$$|\psi\rangle = |\psi_0\rangle + |\psi_1\rangle$$

where $|\psi_0\rangle \in S_{i,0}$ and $|\psi_1\rangle \in S_{i,1}$, then we have the followings:[11]

1. ($|\psi_0\rangle$ **"does not know"** $a_i \in \mathsf{Acc}_{k_i,y_i}$.) If we apply U to $|c\rangle_{\mathbf{C}} |\psi_0\rangle_{\mathbf{X},\mathbf{Z}}$ for $c \xleftarrow{\$} \{0,1\}^m$ such that $c_i = 0$ and measures the $\mathbf{X}_i$ register in computational basis to obtain a_i, then $a_i \in \mathsf{Acc}_{k_i,y_i}$ with "small" probability.[12]
2. ($|\psi_1\rangle$ **"knows"** $a_i \in \mathsf{Acc}_{k_i,y_i}$.) There is an efficient algorithm that is given $|\psi_1\rangle$ as input and outputs $a_i \in \mathsf{Acc}_{k_i,y_i}$ with overwhelming probability.
3. (**Efficient projection.**) A measurement described by $\{\Pi_{S_{i,0}}, \Pi_{S_{i,1}}\}$ can be performed efficiently where $\Pi_{S_{i,0}}$ and $\Pi_{S_{i,1}}$ denote projections to $S_{i,0}$ and $S_{i,1}$, respectively.

If this is true, then the rest of the proof would be easy following the outline described in Section 1.2. However, we do not know how to prove it in the above clean form. Therefore we prove a noisy version of the above claim where

[9] $\mathcal{H}_{\mathbf{X}} \otimes \mathcal{H}_{\mathbf{Z}}$ corresponds to $\mathcal{H}_P$ in Section 1.2.

[10] Here, we hardwire into the cheating prover the instance $x \notin L$ on which it will cheat instead of giving it as an input.

[11] $|\psi_0\rangle$ and $|\psi_1\rangle$ correspond to $|\psi_{i,0}\rangle$ and $|\psi_{i,1}\rangle$ in Section 1.2, respectively.

[12] The threshold for "small" can be set to be any noticeable function.

1. the way of decomposition is randomized,
2. there is an error term, i.e., we decompose $|\psi\rangle$ as

$$|\psi\rangle = |\psi_0\rangle + |\psi_1\rangle + |\psi_{err}\rangle$$

   by using a state $|\psi_{err}\rangle$ whose norm is "small" on average, and
3. we have $\| |\psi_0\rangle \|^2 + \| |\psi_1\rangle \|^2 \leq \| |\psi\rangle \|^2$. We note that this condition automatically follows if $|\psi_0\rangle$ and $|\psi_1\rangle$ are orthogonal as in the above clean version, but they may not be orthogonal in our case.

Specifically, our lemma is stated as follows:

Lemma 4. *Let (U_0, U) be any prover's strategy. Let $m = O(\log n)$, $i \in [m]$, $\gamma_0 \in [0,1]$, and $T \in \mathbb{N}$ such that $\frac{\gamma_0}{T} = 1/\mathsf{poly}(n)$. Let γ be sampled uniformly randomly from $[\frac{\gamma_0}{T}, \frac{2\gamma_0}{T}, \ldots, \frac{T\gamma_0}{T}]$. Then, there exists an efficient quantum procedure $G_{i,\gamma}$ such that for any (possibly sub-normalized) quantum state $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$,*

$$\begin{aligned} G_{i,\gamma} |0^m\rangle_{\mathbf{C}} |\psi\rangle_{\mathbf{X},\mathbf{Z}} |0^t\rangle_{ph} |0\rangle_{th} |0\rangle_{in} =& z_0 |0^m\rangle_{\mathbf{C}} |\psi_0\rangle_{\mathbf{X},\mathbf{Z}} |0^t 01\rangle_{ph,th,in} \\ &+ z_1 |0^m\rangle_{\mathbf{C}} |\psi_1\rangle_{\mathbf{X},\mathbf{Z}} |0^t 11\rangle_{ph,th,in} + |\psi'_{err}\rangle \end{aligned}$$

where t is the number of qubits in the register ph, $z_0, z_1 \in \mathbb{C}$ such that $|z_0| = |z_1| = 1$, and z_0, z_1, $|\psi_0\rangle_{\mathbf{X},\mathbf{Z}}$, $|\psi_1\rangle_{\mathbf{X},\mathbf{Z}}$, and $|\psi'_{err}\rangle$ may depend on γ.

Furthermore, the following properties are satisfied.

1. ***(Error is Small.)** If we define $|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}} := |\psi\rangle_{\mathbf{X},\mathbf{Z}} - |\psi_0\rangle_{\mathbf{X},\mathbf{Z}} - |\psi_1\rangle_{\mathbf{X},\mathbf{Z}}$, then we have $E_\gamma[\| |\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}} \|^2] \leq \frac{6}{T} + \mathsf{negl}(n)$.*
2. ***(Efficient projection.)** For any fixed γ, $\Pr[M_{ph,th,in} \circ |\psi'_{err}\rangle \in \{0^t 01, 0^t 11\}] = 0$. This implies that if we apply the measurement $M_{ph,th,in}$ on $\frac{G_{i,\gamma}|0^m\rangle_{\mathbf{C}}|\psi\rangle_{\mathbf{X},\mathbf{Z}}|0^t\rangle_{ph}|0\rangle_{th}|0\rangle_{in}}{\||\psi\rangle_{\mathbf{X},\mathbf{Z}}\|}$, then the outcome is $0^t b 1$ with probability $\| |\psi_b\rangle_{\mathbf{X},\mathbf{Z}} \|^2$ and the resulting state in the register $(\mathbf{X},\mathbf{Z})$ is $\frac{|\psi_b\rangle_{\mathbf{X},\mathbf{Z}}}{\||\psi_b\rangle_{\mathbf{X},\mathbf{Z}}\|}$ ignoring a global phase factor.*
3. ***(Projection halves the squared norm.)** For any fixed γ, $E_{b\in\{0,1\}}[\| |\psi_b\rangle_{\mathbf{X},\mathbf{Z}} \|^2] \leq \frac{1}{2} \| |\psi\rangle_{\mathbf{X},\mathbf{Z}} \|^2$.*
4. ***($|\psi_0\rangle$ "does not know" $a_i \in \mathsf{Acc}_{k_i,y_i}$.)** For any fixed γ and $c \in \{0,1\}^m$ such that $c_i = 0$, we have*

$$\Pr\left[M_{\mathbf{X}_i} \circ U \frac{|c\rangle_{\mathbf{C}} |\psi_0\rangle_{\mathbf{X},\mathbf{Z}}}{\| |\psi_0\rangle_{\mathbf{X},\mathbf{Z}} \|} \in \mathsf{Acc}_{k_i,y_i}\right] \leq 2^{m-1}\gamma + \mathsf{negl}(n).$$

5. ***($|\psi_1\rangle$ "knows" $a_i \in \mathsf{Acc}_{k_i,y_i}$.)** For any fixed γ, there exists an efficient quantum algorithm Ext_i such that*

$$\Pr\left[\mathsf{Ext}_i\left(\frac{|0^m\rangle_{\mathbf{C}} |\psi_1\rangle_{\mathbf{X},\mathbf{Z}}}{\| |\psi_1\rangle_{\mathbf{X},\mathbf{Z}} \|}\right) \in \mathsf{Acc}_{k_i,y_i}\right] = 1 - \mathsf{negl}(n).$$

Procedure 1 $G_{i,\gamma}$

1. Do quantum phase estimation U_{est} on $Q = (2\Pi_{in} - I)(2\Pi_{i,out} - I)$ with input state $|0^m\rangle_{\mathbf{C}} |\psi\rangle_{\mathbf{X},\mathbf{Z}}$ and τ-bit precision and failure probability 2^{-n} where the parameter τ will be specified later, i.e.,
$$U_{est} |u\rangle_{\mathbf{C},\mathbf{X},\mathbf{Z}} |0^t\rangle_{ph} \to \sum_{\theta \in (-\pi,\pi]} \alpha_\theta |u\rangle_{\mathbf{C},\mathbf{X},\mathbf{Z}} |\theta\rangle_{ph}.$$
such that $\sum_{\theta \notin \bar{\theta} \pm 2^{-\tau}} |\alpha_\theta|^2 \leq 2^{-n}$ for any eigenvector $|u\rangle_{\mathbf{C},\mathbf{X},\mathbf{Z}}$ of Q with eigenvalue $e^{i\bar{\theta}}$.
2. Apply $U_{th} : |u\rangle_{\mathbf{C},\mathbf{X},\mathbf{Z}} |\theta\rangle_{ph} |0\rangle_{th} \xrightarrow{U_{th}} |u\rangle_{\mathbf{C},\mathbf{X},\mathbf{Z}} |\theta\rangle_{ph} |b\rangle_{th}$, where $b = 1$ if $\cos^2(\theta/2) \geq \gamma - \delta$.
3. Apply $U_{est}^\dagger$.
4. Apply $U_{in} : |c\rangle_{\mathbf{C}} |0\rangle_{in} \xrightarrow{U_{in}} |c\rangle_{\mathbf{C}} |b'\rangle_{in}$, where $b' = 1$ if $c = 0^m$.

We prove that the algorithm $G_{i,\gamma}$ given in Figure 1 satisfies the above conditions. The proof is based on a lemma about two projectors shown by Nagaj, Wocjan, and Zhang [36], which in turn is based on the Jordan's lemma. See the full version for a proof.

In Lemma 4, we showed that by fixing any $i \in [m]$, we can partition any prover's state $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$ into $|\psi_0\rangle_{\mathbf{X},\mathbf{Z}}$, $|\psi_1\rangle_{\mathbf{X},\mathbf{Z}}$, and $|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}$ with certain properties. In the following, we sequentially apply Lemma 4 for each $i \in [m]$ to further decompose the prover's state.

Lemma 5. *Let m, γ_0, T be as in Lemma 4, and let $\gamma_i \xleftarrow{\$} [\frac{\gamma_0}{T}, \frac{2\gamma_0}{T}, \ldots, \frac{T\gamma_0}{T}]$ for each $i \in [m]$. For any $c \in \{0,1\}^m$, a state $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$ can be partitioned as follows.*

$$|\psi\rangle_{\mathbf{X},\mathbf{Z}} = |\psi_{c_1}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,c_2}\rangle_{\mathbf{X},\mathbf{Z}} + \cdots + |\psi_{\bar{c}_1,\ldots,\bar{c}_{m-1},c_m}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}$$

where the way of partition may depend on the choice of $\hat{\gamma} = \gamma_1 \ldots \gamma_m$. Further, the following properties are satisfied.

1. *For any fixed $\hat{\gamma}$ and any c, $i \in [m]$ such that $c_i = 0$, we have*
$$\Pr\left[M_{\mathbf{X}_i} \circ U \frac{|0^m\rangle_{\mathbf{C}} |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},0}\rangle_{\mathbf{X},\mathbf{Z}}}{\| |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},0}\rangle_{\mathbf{X},\mathbf{Z}} \|} \in \mathsf{Acc}_{k_i,y_i}\right] \leq 2^{m-1}\gamma_0 + \mathsf{negl}(n).$$
2. *For any fixed $\hat{\gamma}$ and any c, $i \in [m]$ such that $c_i = 1$, there exists an efficient algorithm Ext_i such that*
$$\Pr\left[\mathsf{Ext}_i\left(\frac{|0^m\rangle_{\mathbf{C}} |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}}{\| |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}} \|}\right) \in \mathsf{Acc}_{k_i,y_i}\right] = 1 - \mathsf{negl}(n).$$
3. *For any fixed $\hat{\gamma}$, we have $E_c[\| |\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}} \|^2] \leq 2^{-m}$.*

4. *For any fixed c, we have $E_{\hat{\gamma}}[\| |\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}} \|^2] \leq \frac{6m^2}{T} + \mathsf{negl}(n)$.*
5. *For any fixed $\hat{\gamma}$ and c there exists an efficient quantum algorithm $H_{\hat{\gamma},c}$ that is given $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$ as input and produces $\frac{|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}}{\||\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}\|}$ with probability $\| |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}} \|^2$ ignoring a global phase factor.*

Proof. We inductively define $|\psi_{c_1}\rangle_{\mathbf{X},\mathbf{Z}},\ldots,|\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}}$ as follows.

First, we apply Lemma 4 for the state $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$ with $\gamma = \gamma_1$ to give a decomposition

$$|\psi\rangle_{\mathbf{X},\mathbf{Z}} = |\psi_0\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_1\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{err,1}\rangle_{\mathbf{X},\mathbf{Z}}$$

where $|\psi_{err,1}\rangle_{\mathbf{X},\mathbf{Z}}$ corresponds to $|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}$ in Lemma 4.

For each $i = 2, \ldots, m$, we apply Lemma 4 for the state $|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1}}\rangle_{\mathbf{X},\mathbf{Z}}$ with $\gamma = \gamma_i$ to give a decomposition

$$|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1}}\rangle_{\mathbf{X},\mathbf{Z}} = |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},0}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{err,i}\rangle_{\mathbf{X},\mathbf{Z}}$$

where $|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},0}\rangle_{\mathbf{X},\mathbf{Z}}$, $|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}$, and $|\psi_{err,i}\rangle_{\mathbf{X},\mathbf{Z}}$ corresponds to $|\psi_0\rangle_{\mathbf{X},\mathbf{Z}}$, $|\psi_1\rangle_{\mathbf{X},\mathbf{Z}}$, and $|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}$ in Lemma 4, respectively.

Then it is easy to see that we have

$$|\psi\rangle_{\mathbf{X},\mathbf{Z}} = |\psi_{c_1}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,c_2}\rangle_{\mathbf{X},\mathbf{Z}} + \cdots + |\psi_{\bar{c}_1,\ldots,\bar{c}_{m-1},c_m}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}$$

where we define $|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}} := \sum_{i=1}^{m} |\psi_{err,i}\rangle_{\mathbf{X},\mathbf{Z}}$.

The first and second claims immediately follow from the fourth and fifth claims of Lemma 4 and $\gamma_i \leq \gamma_0$ for each $i \in [m]$.

By the third claim of Lemma 4, we have $E_{c_1\ldots c_i}[\| |\psi_{\bar{c}_1,\ldots,\bar{c}_i}\rangle_{\mathbf{X},\mathbf{Z}} \|] \leq \frac{1}{2} E_{c_1\ldots c_{i-1}}[\| |\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1}}\rangle_{\mathbf{X},\mathbf{Z}} \|]$. Ths implies the third claim.

By the first claim of Lemma 4, we have $E_{\gamma_i}[\| |\psi_{err,i}\rangle_{\mathbf{X},\mathbf{Z}} \|^2] \leq \frac{6}{T} + \mathsf{negl}(n)$. The fourth claim follows from this and the Cauchy-Schwarz inequality.

Finally, for proving the fifth claim, we define the procedure $H_{\hat{\gamma},c}$ as described in Procedure 2 We can easily see that $H_{\hat{\gamma},c}$ satisfies the desired property by the second claim of Lemma 4.

Given Lemma 5, we can start proving Theorem 2.

Proof (Proof of Theorem 2).

First, we recall how a cheating prover characterized by (U_0, U) works. When the first message k is given, it first applies

$$U_0 |0\rangle_{\mathbf{X},\mathbf{Z}} |0\rangle_{\mathbf{Y}} |k\rangle_{\mathbf{K}} \xrightarrow{\text{measure } \mathbf{Y}} |\psi(k,y)\rangle_{\mathbf{X},\mathbf{Z}} |k\rangle_{\mathbf{K}}.$$

to generate the second message y and $|\psi(k,y)\rangle_{\mathbf{X},\mathbf{Z}}$. Then after receiving the third message c, it applies U on $|c\rangle_{\mathbf{C}} |\psi(k,y)\rangle_{\mathbf{X},\mathbf{Z}}$ and measures the register $\mathbf{X}$ in the computational basis to obtain the fourth message a. In the following, we just write $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$ to mean $|\psi(k,y)\rangle_{\mathbf{X},\mathbf{Z}}$ for notational simplicity. Let $M_{i,k_i,\mathsf{td}_i,y_i,c_i}$ be

Procedure 2 $H_{\hat{\gamma},c}$

On input $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$, it works as follows:
For each $i = 1, ..., m$, it applies

1. Prepare registers $\mathbf{C}$, (ph_1, th_1, in_1),..., (ph_m, th_m, in_m) all of which are initialized to be $|0\rangle$.
2. For each $i = 1, ..., m$, do the following:
   (a) Apply G_{i,γ_i} on the quantum state in the registers $(\mathbf{C}, \mathbf{X}, \mathbf{Z}, ph_i, th_i, in_i)$.
   (b) Measure the registers (ph_i, th_i, in_i) in the computational basis.
   (c) If the outcome is $0^t c_i 1$, then it halts and returns the state in the register $(\mathbf{X}, \mathbf{Z})$. If the outcome is $0^t \bar{c}_i 1$, continue to run. Otherwise, immediately halt and abort.

the measurement that outputs the verification result of the value in the register $\mathbf{X}_i$ w.r.t. $k_i, \mathsf{td}_i, y_i, c_i$, and let $M_{k,\mathsf{td},y,c}$ be the measurement that returns $\top$ if and only if $M_{i,k_i,\mathsf{td}_i,y_i,c_i}$ returns $\top$ for all $i \in [m]$ where $k = (k_1, ..., k_m)$, $\mathsf{td} = (\mathsf{td}_1, ..., \mathsf{td}_m)$, $y = (y_1, ..., y_m)$ and $c = (c_1, ..., c_m)$. With this notation, a cheating prover's success probability can be written as

$$\Pr_{k,\mathsf{td},y,c}[M_{k,\mathsf{td},y,c} \circ U |c\rangle_{\mathbf{C}} |\psi\rangle_{\mathbf{X},\mathbf{Z}} = \top].$$

Let γ_0, $\hat{\gamma}$, and T be as in Lemma 5. According to Lemma 5, for any fixed $\hat{\gamma}$ and $c \in \{0,1\}^m$, we can decompose $|\psi\rangle_{\mathbf{X},\mathbf{Z}}$ as

$$|\psi\rangle_{\mathbf{X},\mathbf{Z}} = |\psi_{c_1}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,c_2}\rangle_{\mathbf{X},\mathbf{Z}} + \cdots + |\psi_{\bar{c}_1,...,\bar{c}_{m-1},c_m}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,...,\bar{c}_{m-1},\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}.$$

To prove the theorem, we show the following two inequalities. First, for any fixed $\hat{\gamma}$, $i \in [m]$, $c \in \{0,1\}^m$ such that $c_i = 0$, k_i, td_i, and y_i, we have

$$\Pr\left[M_{i,k_i,\mathsf{td}_i,y_i,0} \circ \frac{U |c\rangle_{\mathbf{C}} |\psi_{\bar{c}_1,...,\bar{c}_{i-1},0}\rangle_{\mathbf{X},\mathbf{Z}}}{\| |\psi_{\bar{c}_1,...,\bar{c}_{i-1},0}\rangle_{\mathbf{X},\mathbf{Z}} \|} = \top\right] \leq 2^{m-1}\gamma_0 + \mathsf{negl}(n). \quad (1)$$

This easily follows from the first claim of Lemma 5

Second, for any fixed $\hat{\gamma}$, $i \in [m]$, and $c \in \{0,1\}^m$ such that $c_i = 1$, we have

$$\underset{k,\mathsf{td},y}{E}\left[\| |\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}} \|^2 \Pr\left[M_{i,k_i,\mathsf{td}_i,y_i,1} \circ U \frac{|c\rangle_{\mathbf{C}} |\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}}{\| |\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}} \|} = \top\right]\right] = \mathsf{negl}(n) \quad (2)$$

assuming the quantum hardness of LWE problem.

For proving Eq. 2, we consider a cheating prover against the original Mahadev's protocol on the i-th corrdinate described below:

1. Given k_i, it picks $k_{-i} = k_1...k_{i-1}, k_{i+1}, ..., k_m$ as in the protocol and computes $U_0 |0\rangle_{\mathbf{X},\mathbf{Z}} |0\rangle_{\mathbf{Y}} |k\rangle_{\mathbf{K}}$ and measure the register $\mathbf{Y}$ to obtain $y = (y_1, ..., y_m)$ along with the corresponding state $|\psi\rangle_{\mathbf{X},\mathbf{Z}} = |\psi(k, y)\rangle_{\mathbf{X},\mathbf{Z}}$.

2. Apply $H_{\hat{\gamma},c}$ (which is defined in the fifth claim of Lemma 5) to generate the state $\frac{|\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}}{\||\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}\|}$, which succeeds with probability $\| |\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}} \|^2$ (ignoring a global phase factor). We denote by Succ the event that it succeeds in generating the state. If it fails to generate the state, then it overrides y_i by picking it in a way such that it can pass the test round with probability 1, which can be done according to Fact 1. Then it sends y_i to the verifier.

3. Given a challenge c'_i, it works as follows:

   - When $c'_i = 0$ (i.e., Test round), if Succ occurred, then it runs Ext_i in the second claim of Lemma 5 on input $\frac{|0^m\rangle_{\mathbf{C}}|\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}}{\||\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}\|}$ to generate a fourth message accepted with probability $1 - \mathsf{negl}(n)$. If Succ did not occur, then it returns a fourth message accepted with probability 1, which is possible by Fact 1.

   - When $c'_i = 1$ (i.e., Hadamard round), if Succ occurred, then it computes $U\frac{|c\rangle_{\mathbf{C}}|\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}}{\||\psi_{\bar{c}_1,...,\bar{c}_{i-1},1}\rangle_{\mathbf{X},\mathbf{Z}}\|}$ and measure the register $\mathbf{X}_i$ to obtain the fourth message a_i. If Succ did not occur, it just aborts.

Then we can see that this cheating adversary passes the test round with overwhelming probability and passes the Hadamard round with the probability equal to the LHS of Eq. 2. Therefore, Eq. 2 follows from Lemma 3 assuming the quantum hardness of LWE problem.

Now, we are ready to prove the soundness of the parallel repetition version of Mahadev's protocol (Theorem 2). As remarked at the beginning of Sec. 3.3, it suffices to show that for any $\mu = 1/\mathsf{poly}(n)$, there exists $m = O(\log(n))$ such that the success probability of the cheating prover is at most μ. Here we set $m = \log \frac{1}{\mu^2}$, $\gamma_0 = 2^{-2m}$, and $T = 2^m$. Note that this parameter setting satisfies the requirement for Lemma 5 since $m = \log \frac{1}{\mu^2} = \log(\mathsf{poly}(n)) = O(\log n)$ and

$\frac{\gamma_0}{T} = 2^{-3m} = \mu^6 = 1/\mathsf{poly}(n)$. Then we have

$$
\begin{aligned}
&\Pr_{k,\mathsf{td},y,c}\left[M_{k,\mathsf{td},y,c} \circ U\,|c\rangle_{\mathbf{C}}\,|\psi\rangle_{\mathbf{X},\mathbf{Z}} = \top\right] \\
&= \Pr_{k,\mathsf{td},y,c,\hat{\gamma}}\left[M_{k,\mathsf{td},y,c} \circ U\,|c\rangle_{\mathbf{C}}\left(\sum_{i=1}^{m}|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}} + |\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}\right) = \top\right] \\
&\leq (m+2)\,E_{k,\mathsf{td},y,c,\hat{\gamma}}\left[\sum_{i=1}^{m}\|\,|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}\,\|^2 \Pr\left[M_{k,\mathsf{td},y,c} \circ U\frac{|c\rangle_{\mathbf{C}}\,|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}}{\|\,|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}\,\|} = \top\right]\right. \\
&+ \|\,|\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}}\,\|^2 \Pr\left[M_{k,\mathsf{td},y,c} \circ U\frac{|c\rangle_{\mathbf{C}}\,|\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}}}{\|\,|\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}}\,\|} = \top\right] \\
&\left.+ \|\,|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}\,\|^2 \Pr\left[M_{k,\mathsf{td},y,c} \circ U\frac{|c\rangle_{\mathbf{C}}\,|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}}{\|\,|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}\,\|} = \top\right]\right] \\
&\leq (m+2)\,E_{k,\mathsf{td},y,c,\hat{\gamma}}\left[\sum_{i=1}^{m}\|\,|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}\,\|^2 \Pr\left[M_{i,k_i,\mathsf{td}_i,y_i,c_i} \circ U\frac{|c\rangle_{\mathbf{C}}\,|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}}{\|\,|\psi_{\bar{c}_1,\ldots,\bar{c}_{i-1},c_i}\rangle_{\mathbf{X},\mathbf{Z}}\,\|} = \top\right]\right. \\
&\left.+ \|\,|\psi_{\bar{c}_1,\ldots,\bar{c}_m}\rangle_{\mathbf{X},\mathbf{Z}}\,\|^2 + \|\,|\psi_{err}\rangle_{\mathbf{X},\mathbf{Z}}\,\|^2\right] \\
&\leq (m+2)(m(2^{m-1}\gamma_0 + \mathsf{negl}(n)) + 2^{-m} + \frac{6m^2}{T} + \mathsf{negl}(n)) \\
&\leq \mathsf{poly}(\log \mu^{-1})\mu^2 + \mathsf{negl}(n).
\end{aligned}
$$

The first equation follows from Lemma 5. The first inequality follows from Lemma 2. The second inequality holds since considering the verification on a particular coordinate just increases the acceptance probability and probabilities are at most 1. The third inequality follows from Eq. 1 and 2, which give an upper bound of the first term and Lemma 5, which gives upper bounds of the second and third terms. The last inequality follows from our choices of γ_0, T, and m. For sufficiently large n, this can be upper bounded by μ. Since $\Pr_{k,\mathsf{td},y,c}[M_{k,\mathsf{td},y,c} \circ U\,|c\rangle_{\mathbf{C}}\,|\psi\rangle_{\mathbf{X},\mathbf{Z}} = \top]$ is the success probability of a cheating prover, the above inequality means that for any $\mu = 1/\mathsf{poly}(n)$, there exists $m = O(\log(n))$ such that the success probability of the cheating prover is at most μ. As remarked at the beginning of Sec. 3.3, this suffices for proving that a chearing prover's success probability is negligible when $m = \omega(\log n)$.

4 Two-Round Protocol via Fiat-Shamir Transform

In this section, we show that if we apply the Fiat-Shamir transform to m-parallel version of the Mahadev's protocol, then we obtain two-round protocol in the QROM. That is, we prove the following theorem.

Theorem 3. *Assuming LWE assumption, there exists a two-round CVQC protocol with overwhelming completeness and negligible soundness error in the QROM.*

Proof. Let $m > n$ be a sufficiently large integer so that m-parallel version of the Mahadev's protocol has negligible soundness. For notational simplicity, we abuse the notation to simply use V_i, P_i, and V_{out} to mean the m-parallel repetitions of them. Let $H : \mathcal{Y} \to \{0,1\}^m$ be a hash function idealized as a quantum random oracle where $\mathcal{X}$ is the space of the second message y and $\mathcal{Y} = \{0,1\}^m$. Our two-round protocol is described below:

First Message: The verifier runs V_1 to generate (k, td). Then it sends k to the prover and keeps td as its state.

Second Message: The prover runs P_2 on input k to generate y along with the prover's state $|\mathsf{st}_P\rangle$. Then set $c := H(y)$, and runs P_4 on input $|\mathsf{st}_P\rangle$ and y to generate a. Finally, it returns (y, a) to the verifier.

Verification: The verifier computes $c = H(y)$, runs $V_{\mathsf{out}}(k, \mathsf{td}, y, c, a)$, and outputs as V_{out} outputs.

It is clear that the completeness is preserved given that H is a random oracle. We can reduce the soundness of this protocol to the soundness of m-parallel version of the Mahadev's protocol by using the result of [19], which shows that Fiat-Shamir transform preserves soundness in the QROM. See the full version for details.

5 Making Verifier Efficient

In this section, we construct a CVQC protocol with efficient verification in the CRS+QRO model where a classical common reference string is available for both prover and verifier in addition to quantum access to QRO. Our main theorem in this section is stated as follows:

Theorem 4. *Assuming LWE assumption and existence of post-quantum iO, post-quantum FHE, and two-round CVQC protocol in the standard model, there exists a two-round CVQC protocol for* $\mathsf{QTIME}(T)$ *with verification complexity* $\mathsf{poly}(n, \log T)$ *in the CRS+QRO model.*

Remark 2. One may think that the underlying two-round CVQC protocol can be in the QROM instead of in the standard model since we rely on the QROM anyway. However, this is not the case since we need to use the underlying two-round CVQC in a non-black box way, which cannot be done if that is in the QROM. Since our two-round protocol given in Sec. 4 is only proven secure in the QROM, we do not know any two-round CVQC protocol provably secure in the standard model. On the other hand, it is widely used heuristic in cryptography that a scheme proven secure in the QROM is also secure in the standard model if the QRO is instantiated by a well-designed cryptographic hash function. For example, many candidates for the NIST post-quantum standardization [37] give security proofs in the QROM and claim their security in the real world. Therefore, we believe that it is reasonable to assume that a standard model instantiation of the scheme in Sec. 4 with a concrete hash function is sound.

Remark 3. One may think we need not assume CRS in addition to QRO since CRS may be replaced with an output of QRO. This can be done if CRS is just a uniformly random string. However, in our construction, CRS is non-uniform and has a certain structure. Therefore we cannot implement CRS by QRO.

5.1 Four-Round Protocol

First, we construct a four-round scheme with efficient verification, which is transformed into two-round protocol in the next subsection. Our construction is based on the following building blocks. Definitions of them can be found in the full version.

- A two-round CVQC protocol $\Pi = (P = P_2, V = (V_1, V_{\mathsf{out}}))$ in the standard model, which works as follows:
  V_1**:** On input the security parameter 1^n and x, it generates a pair (k, td) of a"key" and "trapdoor", sends k to P, and keeps td as its internal state.
  P_2**:** On input x and k, it generates a response e and sends it to V.
  V_{out}**:** On input x, k, td, e, it returns $\top$ indicating acceptance or $\bot$ indicating rejection.
- A post-quantum PRG $\mathsf{PRG} : \{0,1\}^{\ell_s} \to \{0,1\}^{\ell_r}$ where ℓ_r is the length of randomness for V_1.
- An FHE scheme $\Pi_{\mathsf{FHE}} = (\mathsf{FHE.KeyGen}, \mathsf{FHE.Enc}, \mathsf{FHE.Eval}, \mathsf{FHE.Dec})$ with post-quantum CPA security.
- A strong output compressing randomized encoding scheme $\Pi_{\mathsf{RE}} = (\mathsf{RE.Setup}, \mathsf{RE.Enc}, \mathsf{RE.Dec})$ with post-quantum security. We denote the simulator for Π_{RE} by $\mathcal{S}_{\mathsf{re}}$.
- A SNARK $\Pi_{\mathsf{SNARK}} = (P_{\mathsf{snark}}, V_{\mathsf{snark}})$ in the QROM for an NP language L_{snark} defined below:
  We have $(x, \mathsf{pk}_{\mathsf{fhe}}, \mathsf{ct}, \mathsf{ct}') \in L_{\mathsf{snark}}$ if and only if there exists e such that $\mathsf{ct}' = \mathsf{FHE.Eval}(\mathsf{pk}_{\mathsf{fhe}}, C[x,e], \mathsf{ct})$ where $C[x,e]$ is a circuit that works as follows:
  $C[x,e](s)$**:** Given input s, it computes $(k, \mathsf{td}) \xleftarrow{\$} V_1(1^n, x; PRG(s))$, and returns 1 if and only if $V_{\mathsf{out}}(x, k, \mathsf{td}, e) = \top$ and 0 otherwise.

Let L be a BPP language decided by a quantum Turing machine QTM (i.e., for any $x \in \{0,1\}^*$, $x \in L$ if and only if QTM accepts x), and for any T, L_T denotes the set consisting of $x \in L$ such that QTM accepts x in T steps. Then we construct a 4-round CVQC protocol $(\mathsf{Setup}_{\mathsf{eff}}, P_{\mathsf{eff}} = (P_{\mathsf{eff},2}, P_{\mathsf{eff},4}), V_{\mathsf{eff}} = (V_{\mathsf{eff},1}, V_{\mathsf{eff},3}, V_{\mathsf{eff},\mathsf{out}}))$ for L_T in the CRS+QRO model where the verifier's efficiency only logarithmically depends on T. Let $H : \{0,1\}^{2n} \times \{0,1\}^{2n} \to \{0,1\}^n$ be a quantum random oracle.

$\mathsf{Setup}_{\mathsf{eff}}(1^n)$**:** The setup algorithm takes the security parameter 1^n as input, generates $\mathsf{crs}_{\mathsf{re}} \xleftarrow{\$} \{0,1\}^\ell$ and computes $\mathsf{ek}_{\mathsf{re}} \xleftarrow{\$} \mathsf{RE.Setup}(1^n, 1^\ell, \mathsf{crs}_{\mathsf{re}})$ where ℓ is a parameter specified later. Then it outputs a CRS for verifier $\mathsf{crs}_{V_{\mathsf{eff}}} := \mathsf{ek}_{\mathsf{re}}$ and a CRS for prover $\mathsf{crs}_{P_{\mathsf{eff}}} := \mathsf{crs}_{\mathsf{re}}$.[13]

[13] We note that we divide the CRS into $\mathsf{crs}_{V_{\mathsf{eff}}}$ and $\mathsf{crs}_{P_{\mathsf{eff}}}$ just for the verifier efficiency and soundness still holds even if a cheating prover sees $\mathsf{crs}_{V_{\mathsf{eff}}}$.

$V^H_{\mathsf{eff},1}$**:** Given $\mathsf{crs}_{V_{\mathsf{eff}}} = \mathsf{ek}_{\mathsf{re}}$ and x, it generates $s \xleftarrow{\$} \{0,1\}^{\ell_s}$ and $(\mathsf{pk}_{\mathsf{fhe}}, \mathsf{sk}_{\mathsf{fhe}}) \xleftarrow{\$} \mathsf{FHE.KeyGen}(1^n)$, computes $\mathsf{ct} \xleftarrow{\$} \mathsf{FHE.Enc}(\mathsf{pk}_{\mathsf{fhe}}, s)$ and $\widehat{M_{\mathsf{inp}}} \xleftarrow{\$} \mathsf{RE.Enc}(\mathsf{ek}_{\mathsf{re}}, M, s, T')$ where M is a Turing machine that works as follows:
$M(s)$**:** Given an input $s \in \{0,1\}^{\ell_s}$, it computes $(k, \mathsf{td}) \xleftarrow{\$} V_1(1^n, x; PRG(s))$ and outputs k
and T' is specified later. Then it sends $(\widehat{M_{\mathsf{inp}}}, \mathsf{pk}_{\mathsf{fhe}}, \mathsf{ct})$ to P_{eff} and keeps $\mathsf{sk}_{\mathsf{fhe}}$ as its internal state.

$P^H_{\mathsf{eff},2}$**:** Given $\mathsf{crs}_{P_{\mathsf{eff}}} = \mathsf{crs}_{\mathsf{re}}$, x and the message $(\widehat{M_{\mathsf{inp}}}, \mathsf{pk}_{\mathsf{fhe}}, \mathsf{ct})$ from the verifier, it computes $k \leftarrow \mathsf{RE.Dec}(\mathsf{crs}_{\mathsf{re}}, \widehat{M_{\mathsf{inp}}})$, $e \xleftarrow{\$} P_2(x,k)$, and $\mathsf{ct}' \leftarrow \mathsf{FHE.Eval}(\mathsf{pk}_{\mathsf{fhe}}, C[x,e], \mathsf{ct})$ where $C[x,e]$ is a classical circuit defined above. Then it sends ct' to V_{eff} and keeps $(\mathsf{pk}_{\mathsf{fhe}}, \mathsf{ct}, \mathsf{ct}', e)$ as its state.

$V^H_{\mathsf{eff},3}$ Upon receiving ct', it randomly picks $z \xleftarrow{\$} \{0,1\}^{2n}$ and sends z to P_{eff}.

$P^H_{\mathsf{eff},4}$ Upon receiving z, it computes $\pi_{\mathsf{snark}} \xleftarrow{\$} P^{H(z,\cdot)}_{\mathsf{snark}}((x, \mathsf{pk}_{\mathsf{fhe}}, \mathsf{ct}, \mathsf{ct}'), e)$ and sends π_{snark} to V_{eff}.

$V^H_{\mathsf{eff},\mathsf{out}}$**:** It returns $\top$ if $V^{H(z,\cdot)}_{\mathsf{snark}}((x, \mathsf{pk}_{\mathsf{fhe}}, \mathsf{ct}, \mathsf{ct}'), \pi_{\mathsf{snark}}) = \top$ and $1 \leftarrow \mathsf{FHE.Dec}(\mathsf{sk}_{\mathsf{fhe}}, \mathsf{ct}')$ and $\bot$ otherwise.

Choice of parameters.

- We set ℓ to be an upper bound of the length of k where $(k, \mathsf{td}) \xleftarrow{\$} V_1(1^n, x)$ for $x \in L_T$. We note that we have $\ell = \mathsf{poly}(n, T)$.
- We set T' to be an upperbound of the running time of M on input $s \in \{0,1\}^{\ell_s}$ when $x \in L_T$. We note that we have $T' = \mathsf{poly}(n, T)$.

Verification Efficiency. By encoding efficiency of Π_{RE} and verification efficiency of Π_{SNARK}, V_{eff} runs in time $\mathsf{poly}(n, |x|, \log T)$.

Remark 4. We note that the running time of the setup algorithm is $\mathsf{poly}(T)$. This can be done by a trusted party that has a strong (classical) computational power. Alternatively, as in the classical delegating computation literature, we can consider an offline/online setting where the verifier can spend a one-time cost of $\mathsf{poly}(T)$ to setup the CRS in the offline stage, and use it to delegate multiple quantum computation efficiently in the online stage.

Theorem 5 (Completeness). *For any $x \in L_T$,*

$$\Pr\left[\langle P^H_{\mathsf{eff}}(\mathsf{crs}_{P_{\mathsf{eff}}}), V^H_{\mathsf{eff}}(\mathsf{crs}_{V_{\mathsf{eff}}})\rangle(x) = \bot\right] = \mathsf{negl}(n)$$

where $(\mathsf{crs}_{P_{\mathsf{eff}}}, \mathsf{crs}_{V_{\mathsf{eff}}}) \xleftarrow{\$} \mathsf{Setup}_{\mathsf{eff}}(1^n)$.

Proof. This easily follows from completeness and correctness of the underlying primitives.

Theorem 6 (Soundness). *For any $x \notin L_T$ any efficient quantum cheating prover $\mathcal{A}$,*

$$\Pr\left[\langle \mathcal{A}^H(\mathsf{crs}_{P_{\mathsf{eff}}}, \mathsf{crs}_{V_{\mathsf{eff}}}), V^H_{\mathsf{eff}}(\mathsf{crs}_{V_{\mathsf{eff}}})\rangle(x) = \top\right] = \mathsf{negl}(n)$$

where $(\mathsf{crs}_{P_{\mathsf{eff}}}, \mathsf{crs}_{V_{\mathsf{eff}}}) \xleftarrow{\$} \mathsf{Setup}_{\mathsf{eff}}(1^n)$.

A proof can be found in the full version.

5.2 Reducing to Two-Round via Fiat-Shamir

Since the third message is public-coin in the four-round protocol in the previous section, we can apply the Fiat-Shamir transform similarly to Sec.4. Then we obtain the two-round CVQC protocol in the QROM, which completes the proof of Theorem 4. Details can be found in the full version.

Acknowledgement

Kai-Min Chung is partially supported by the Academia Sinica Career Development Award under Grant no. 23-17, and MOST QC project under Grant no. MOST 108-2627-E-002-001-.

Nai-Hui Chia were supported by Scott Aaronson's Vannevar Bush Faculty Fellowship.

References

1. Ibm quantum experience. https://quantum-computing.ibm.com/docs/. Accessed: 2020-05-22.
2. Shweta Agrawal. Indistinguishability obfuscation without multilinear maps: New methods for bootstrapping and instantiation. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part I*, volume 11476 of *LNCS*, pages 191–225. Springer, Heidelberg, May 2019.
3. Shweta Agrawal and Alice Pellet-Mary. Indistinguishability obfuscation without maps: Attacks and fixes for noisy linear FE. In Vincent Rijmen and Yuval Ishai, editors, *EUROCRYPT 2020, Part I*, LNCS, pages 110–140. Springer, Heidelberg, May 2020.
4. Dorit Aharonov, Itai Arad, and Thomas Vidick. Guest column: The quantum pcp conjecture. *SIGACT News*, 44(2):47–79, June 2013.
5. Dorit Aharonov, Michael Ben-Or, Elad Eban, and Urmila Mahadev. Interactive proofs for quantum computations. *arXiv*, 1704.04487, 2017.
6. Gorjan Alagic, Andrew M. Childs, Alex B. Grilo, and Shih-Han Hung. Non-interactive classical verification of quantum computation. In *TCC 2020*, 2020. To appear.
7. Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan,

Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

8. Saikrishna Badrinarayanan, Rex Fernando, Venkata Koppula, Amit Sahai, and Brent Waters. Output compression, MPC, and iO for turing machines. In *ASIACRYPT 2019, Part I*, LNCS, pages 342–370. Springer, Heidelberg, December 2019.
9. Saikrishna Badrinarayanan, Yael Tauman Kalai, Dakshita Khurana, Amit Sahai, and Daniel Wichs. Succinct delegation for low-space non-deterministic computation. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, *50th ACM STOC*, pages 709–721. ACM Press, June 2018.
10. Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. *J. ACM*, 59(2):6:1–6:48, 2012.
11. Mihir Bellare, Russell Impagliazzo, and Moni Naor. Does parallel repetition lower the error in computationally sound protocols? In *38th FOCS*, pages 374–383. IEEE Computer Society Press, October 1997.
12. Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Heidelberg, December 2011.
13. Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Factoring and pairings are not necessary for io: Circular-secure lwe suffices. *IACR Cryptol. ePrint Arch.*, 2020:1024, 2020.
14. Zvika Brakerski, Justin Holmgren, and Yael Tauman Kalai. Non-interactive delegation and batch NP verification from standard computational assumptions. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *49th ACM STOC*, pages 474–482. ACM Press, June 2017.
15. Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. Universal blind quantum computation. In *50th FOCS*, pages 517–526. IEEE Computer Society Press, October 2009.
16. Ran Canetti, Yilei Chen, Justin Holmgren, Alex Lombardi, Guy N. Rothblum, Ron D. Rothblum, and Daniel Wichs. Fiat-Shamir: from practice to theory. In Moses Charikar and Edith Cohen, editors, *51st ACM STOC*, pages 1082–1090. ACM Press, June 2019.
17. Alessandro Chiesa, Peter Manohar, and Nicholas Spooner. Succinct arguments in the quantum random oracle model. In *TCC 2019, Part II*, LNCS, pages 1–29. Springer, Heidelberg, March 2019.
18. Kai-Min Chung, Yi Lee, Han-Hsuan Lin, and Xiaodi Wu. Constant-round blind classical verification of quantum sampling. In submission, 2020.
19. Jelle Don, Serge Fehr, Christian Majenz, and Christian Schaffner. Security of the Fiat-Shamir transformation in the quantum random-oracle model. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 356–383. Springer, Heidelberg, August 2019.
20. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194. Springer, Heidelberg, August 1987.
21. Joseph F Fitzsimons and Elham Kashef. Unconditionally verifiable blind quantum computation. *Physical Review A*, 96(1):012303, 2017.

22. Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. *SIAM J. Comput.*, 45(3):882–929, 2016.
23. Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 169–178. ACM Press, May / June 2009.
24. Alexandru Gheorghiu and Thomas Vidick. Computationally-secure and composable remote state preparation. In *60th FOCS*, pages 1024–1033. IEEE Computer Society Press, 2019.
25. Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: Interactive proofs for muggles. *J. ACM*, 62(4):27:1–27:64, 2015.
26. Justin Holmgren and Ron Rothblum. Delegating computations with (almost) minimal time and space overhead. In Mikkel Thorup, editor, *59th FOCS*, pages 124–135. IEEE Computer Society Press, October 2018.
27. Yael Tauman Kalai, Omer Paneth, and Lisa Yang. How to delegate computations publicly. In Moses Charikar and Edith Cohen, editors, *51st ACM STOC*, pages 1115–1124. ACM Press, June 2019.
28. Yael Tauman Kalai, Ran Raz, and Ron D. Rothblum. Delegation for bounded space. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 565–574. ACM Press, June 2013.
29. Yael Tauman Kalai, Ran Raz, and Ron D. Rothblum. How to delegate computations: the power of no-signaling proofs. In David B. Shmoys, editor, *46th ACM STOC*, pages 485–494. ACM Press, May / June 2014.
30. Joe Kilian. A note on efficient zero-knowledge proofs and arguments (extended abstract). In *24th ACM STOC*, pages 723–732. ACM Press, May 1992.
31. Qipeng Liu and Mark Zhandry. Revisiting post-quantum Fiat-Shamir. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 326–355. Springer, Heidelberg, August 2019.
32. Urmila Mahadev. Classical verification of quantum computations. In Mikkel Thorup, editor, *59th FOCS*, pages 259–267. IEEE Computer Society Press, October 2018.
33. Chris Marriott and John Watrous. Quantum arthur—merlin games. *Comput. Complex.*, 14(2):122–152, June 2005.
34. Silvio Micali. Computationally sound proofs. *SIAM J. Comput.*, 30(4):1253–1298, 2000.
35. Tomoyuki Morimae and Joseph F. Fitzsimons. Post hoc verification with a single prover. *Physical Review Letters*, 120:040501, 2018.
36. Daniel Nagaj, Pawel Wocjan, and Yong Zhang. Fast amplification of qma. *arXiv*, 0904.1549, 2009.
37. NIST. Post-quantum cryptography standardization. https://csrc.nist.gov/Projects/Post-Quantum-Cryptography. Accessed: 2020-09-21.
38. Chris Peikert. A decade of lattice cryptography. *Foundations and Trends in Theoretical Computer Science*, 10(4):283–424, 2016.
39. Krzysztof Pietrzak and Douglas Wikström. Parallel repetition of computationally sound protocols revisited. In Salil P. Vadhan, editor, *TCC 2007*, volume 4392 of *LNCS*, pages 86–102. Springer, Heidelberg, February 2007.
40. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):34:1–34:40, 2009.
41. Ben W Reichardt, Falk Unger, and Umesh Vazirani. Classical command of quantum systems. *Nature*, 496(7746):456, 2013.

42. Omer Reingold, Guy N. Rothblum, and Ron D. Rothblum. Constant-round interactive proofs for delegating computation. In Daniel Wichs and Yishay Mansour, editors, *48th ACM STOC*, pages 49–62. ACM Press, June 2016.
43. Guy N. Rothblum and Salil P. Vadhan. Are pcps inherent in efficient arguments? *Comput. Complex.*, 19(2):265–304, 2010.
44. Hoeteck Wee and Daniel Wichs. Candidate obfuscation via oblivious lwe sampling. *IACR Cryptol. ePrint Arch.*, 2020:1042, 2020.