2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)

Computationally-secure and composable remote state preparation

(Extended abstract)

Alexandru Gheorghiu and Thomas Vidick
Department of Computing and Mathematical Sciences
California Institute of Technology, USA
Email: {andrugh, vidick}@caltech.edu

***Abstract*—We introduce a protocol between a classical polynomial-time verifier and a quantum polynomial-time prover that allows the verifier to securely delegate to the prover the preparation of certain single-qubit quantum states. The prover is unaware of which state he received and moreover, the verifier can check with high confidence whether the preparation was successful. The delegated preparation of single-qubit states is an elementary building block in many quantum cryptographic protocols. We expect our implementation of "random remote state preparation with verification", a functionality first defined in (Dunjko and Kashefi 2014), to be useful for removing the need for quantum communication in such protocols while keeping functionality. The main application that we detail is to a protocol for blind and verifiable delegated quantum computation (DQC) that builds on the work of (Fitzsimons and Kashefi 2018), who provided such a protocol with quantum communication. Recently, both blind and verifiable DQC were shown to be possible, under computational assumptions, with a classical polynomial-time client (Mahadev 2017, Mahadev 2018). Compared to the work of Mahadev, our protocol is more modular, applies to the measurement-based model of computation (instead of the Hamiltonian model) and is composable. Our proof of security builds on ideas introduced in (Brakerski et al. 2018).**

***Keywords*-verifiable quantum computation; composable security; learning with errors**

For a full version of the paper, we refer to [GV19].

I. Introduction

In the problem of delegated computation a user (often referred to as *client* or *verifier*) is provided as input a pair (C, x) of a circuit C and an input x for the circuit. The verifier's task is to evaluate $C(x)$ as efficiently as possible. For this the verifier may delegate some or all of the computation to a powerful but untrusted server (often referred to as the *prover*). Let n be the length of x and T the size of the circuit C. Ideally, the runtime of the verifier is (quasi-)linear in n and poly-logarithmic in T, while the runtime of the prover is quasi-linear in T. (Reducing space usage, for both the verifier and the prover, is also of interest, but for simplicity we focus on time.)

A productive line of research in complexity and cryptography has led to protocols for delegated computation with increasing efficiency and whose soundness can be *information-theoretic* [GKR15] or based on *cryptographic assumptions* [Kil92], [KRR14]. The latter type include protocols utilizing public-key cryptography and making standard cryptographic assumptions, such as [HR18], as well as non-interactive protocols based on more non-standard assumptions, such as [GGPR13]. In addition to the natural applications in cloud and distributed computing, research in delegated computation is motivated by cryptographic applications (such as short zero-knowledge proofs [Gro10], [BSCG$^+$13]) and connections to complexity theory (such as the theory of multiprover interactive proof systems [KRR14] and probabilistically checkable proofs [GKR15]).

In this paper we are concerned with the problem of *delegating quantum computations* (DQC). Here the verifier is provided as input the classical description of a quantum circuit C, as well as a classical input x for the circuit, and its goal is to obtain the result of a measurement of the output qubit of C in the computational basis, when it is executed on x.[1] In this context the main question is the following: What security guarantees can DQC protocols achieve, and at what cost?

To gain an understanding of the current landscape around this question we briefly discuss the most relevant known results, referring to [GKK17] for a more extensive treatment. First we note that DQC protocols come with two related but seemingly independent types of security guarantee: *blindness* and *verifiability*. A

[1] For simplicity we restrict to circuits that take classical inputs and return a single classical output bit obtained as the result of a measurement that is promised to return a particular value, 0 or 1, with probability at least $\frac{2}{3}$. This setting corresponds to delegating *decision problems*, i.e. problems in which the output is a single bit. Our results also apply to the setting of *relational* or *sampling* problems for which the output consists of multiple bits.

2575-8454/19/$31.00 ©2019 IEEE
DOI 10.1109/FOCS.2019.00066

DQC protocol is said to be blind if throughout the interaction the prover does not learn anything about the delegated computation except for an upper bound on its size. A DQC protocol is said to be verifiable if it is unlikely for the prover to succeed in convincing the verifier to accept a false statement. The question of blind delegation of quantum computation was first considered by Childs [Chi05], who gave such a protocol with quantum communication. Verifiable delegation of quantum computation was formalized in [ABE10], [BFK09] (see also [ABOEM17], [FK17]); the authors gave protocols for verifiable DQC, and just like Childs' protocol, these protocols also require quantum communication.

Next we consider the question of efficiency of DQC protocols, focusing on the amount of quantum communication required as a measure of the verifier's "quantum effort". A first class of protocols, such as those from [ABE10], [BFK09], are known as *prepare-and-send* protocols. This is because the verifier is required to prepare a number of small quantum states and send them to the prover. In [ABE10] the size of these quantum states (i.e. the number of qubits) depends on the protocol's soundness (the probability that the verifier accepts an incorrect outcome). In [FK17] the verifier is only required to prepare a number of single-qubit states that depends on the protocol's soundness. A second class of protocols is *receive-and-measure* protocols such as [HM15], [FHM18], in which the verifier receives single qubits from the prover and is required to measure them in one of a small number of possible bases. The protocol that requires the least quantum capability from the verifier is the one from [FHM18]; in their protocol, the verifier only needs to measure the single qubits it receives one at a time in one of two bases, computational and Hadamard. The most communication-efficient protocols fall in the prepare-and-send category and require a total amount of quantum communication that scales as $O(T\log(1/\delta))$ where δ is the soundness error [KW17]; the most efficient protocols in the second category have a cubic dependence on T.

All the aforementioned protocols provide information-theoretic security (for either blindness or verifiability), and all require some limited but nonzero quantum capability for the verifier. In a recent breakthrough Mahadev introduced the first entirely classical protocol for DQC [Mah18b]. The protocol operates in the *Hamiltonian model* of quantum computation, in which instead of directly performing the computation C the prover encodes the outcome of C in the smallest eigenvalue of a local Hamiltonian H_C.[2] The goal of the protocol is for the prover to provide evidence that it has prepared an eigenstate $|\psi\rangle$ of H_C with associated eigenvalue strictly smaller than a. At the heart of Mahadev's result is a commitment procedure that allows the prover to commit to individual qubits of $|\psi\rangle$, and subsequently reveal a measurement outcome for a basis of the verifier's choice, using classical communication alone.

The fact that the verifier in Mahadev's protocol is entirely classical marks a major departure from previous works, yet it comes at a cost in terms of security and efficiency. The security of the protocol is computational and rests on the post-quantum security of the learning with errors problem (LWE); moreover, the protocol is not blind, as the circuit has to be communicated to the prover so that it can determine H_C and prepare an eigenstate.[3] In terms of efficiency, the transformation from circuit to Hamiltonian results in an eigenvalue estimation problem that needs to be solved with accuracy at least $b - a = O(1/T^2)$ for the best constructions known [BC18]. As a result the prover has to prepare $\Omega(T^2)$ copies of the ground state, which implies that at least $\Omega(nT^2)$ single qubits have to be sent by the prover. Moreover, preparation of a smallest eigenvalue eigenstate $|\psi\rangle$ of H_C requires a circuit whose depth scales linearly with T, rather than with the depth of C. This induces a large overhead on the prover's side when the circuit C has low depth but high width[4].

Finally, and arguably most importantly, the protocol is monolithic and not obviously composable: while it solves the desired task of verification of quantum computation, it is not at first clear how or even if the protocol can be simplified to solve more elementary problems (e.g. verifying the preparation of a single qubit state or verifying the application of an elementary quantum operation) or combined with other cryptographic primitives (e.g. to remove or reduce quantum communication in a larger protocol).

Our work is motivated by the following question: does there exist a delegation protocol for quantum computation that combines the appealing feature of having an entirely classical verifier while maintaining

[2] If the circuit returns 0 with probability at least $\frac{2}{3}$, the smallest eigenvalue is smaller than a threshold a, and if it returns 0 with probability less than $\frac{1}{3}$, the smallest eigenvalue is larger than a threshold $b > a$ (this is generally referred to as the "Kitaev circuit-to-Hamiltonian construction" [KSV02]).

[3] The protocol can in principle be made blind by combining it with a scheme for *quantum homomorphic encryption* [Mah18a] but this introduces yet another layer of complexity.

[4] Such circuits are highly parallelizable and one might hope for the complexity of delegating one to scale with depth rather than with total circuit size.

the relative efficiency (small polynomial overhead), simplicity (prover's computation is as close as possible to direct computation of delegated circuit), and security guarantees (verifiability, blindness, composability) of protocols with quantum communication?

II. Overview of results and proof techniques

We answer the question in the affirmative by providing an efficient, composable classical protocol for blind and verifiable DQC. The honest prover in our protocol only needs to implement the desired computation, expressed as a computation in the measurement-based model of computation, together with a sequential preprocessing phase consisting of a number of rounds that depends on the circuit size but such that the complexity of implementing each round scales only with the security parameter. The protocol combines the benefits of the best prepare-and-send quantum-verifier protocols for DQC but requires only classical communication; the downside is that our protocol is computationally sound.

Our DQC protocol is based on a basic quantum functionality that we develop and that we believe has wider applicability than the specific application to DQC. More precisely, we provide a computationally sound and composable protocol for the following two-party task, termed *random remote state preparation* (RSP): Alice (whom we will later identify with the verifier) receives either a uniformly random bit $b \in \{0,1\}$ or a uniformly random value $\theta \in \Theta = \{0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4}\}$ and Bob (whom we will later identify with the prover) receives the single-qubit state $|b\rangle$, in the case when Alice gets b, or the state $|+_\theta\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$, in the case when Alice gets θ. Informally, this amounts to Alice having the ability to "steer" a random state $|+_\theta\rangle$ (or $|b\rangle$) within Bob's workspace, using classical communication only, and such that Bob does not learn the value of θ (or b, respectively).

The idea for RSP was introduced by Dunjko and Kashefi [DK16]. The main functionality they consider is a weaker variant of RSP termed *random remote state preparation with blindness*, or RSP_B. Intuitively, the latter functionality ensures that Bob learns no information about θ, but it allows him to receive a state that is different from $|+_\theta\rangle$. The authors show that RSP_B (and variants of it) can be composed with a prepare-and-send protocol due to [BFK09] to achieve blind (but not verifiable) delegated computation. In [CCKW18] a candidate implementation of RSP_B is given and shown secure against a limited class of adversaries referred to as "honest-but-curious" adversaries. The authors of [DK16] also discuss a stronger form of their primitive, called RSP_S (for *strong*), and observe that it can be used to achieve blind and verifiable DQC by composing it with the protocol of [FK17]. The authors do not, however, provide any instantiation of RSP_S (other than the direct one, using quantum communication).

Our main contribution is to define an ideal functionality, denoted RSP_V (random remote state preparation with verification),[5] and show that it can be implemented using a protocol having computational security and classical communication:

Theorem II.1 (Informal)**.** *Assuming the learning with errors problem is computationally intractable for efficient quantum algorithms, for each $\varepsilon > 0$ there exists a protocol with classical communication that implements the functionality* RSP_V *within distance ε and which has $O(1/\varepsilon^3)$ communication complexity.*

Here, by "implements within distance ε", we mean that any efficient quantum procedure has advantage at most ε in distinguishing the real remote state preparation protocol from the ideal functionality RSP_V.

To show this result we introduce a protocol for remote state preparation and show that it is secure based on the learning with errors problem. This is achieved by building on ideas from [BCM$^+$18], [Mah18b] as well as from the literature on rigidity, self-testing, and quantum random access codes. The protocol consists of a sequence of simple tests that play a similar role to the Bell test in multi-prover entanglement-based protocols for DQC [RUV13]. For completeness it is sufficient to succeed in a constant (depending on the desired distance ε) fraction of rounds, so that a partially faulty device may in principle be used to implement the protocol successfully.

We view RSP_V as a fundamental resource for the construction of interactive protocols that involve classical communication between classical and quantum parties. For our result to be as widely applicable as possible, we establish security of our protocol in the *abstract cryptography* (AC) framework [MR11]. This allows one to use the primitive as a building block in other protocols.

As a specific example of the versatility of RSP we obtain a new protocol for DQC that only requires classical communication. The most natural protocol to which our construction applies is the delegated computation protocol from [FK17]. As already observed

[5] It is not hard to verify that RSP_V is functionally equivalent to RSP_S, in the sense that either functionality can be used to implement the other using a simple protocol. Since the definitions are syntactically different, we use a different name to avoid confusion.

in [DK16], having a remote state preparation functionality immediately yields a blind and verifiable protocol for DQC with classical communication and computational soundness. The resulting protocol is more "direct" than the Mahadev protocol, in the sense that in our construction the operations that the prover has to perform are closer to the quantum computation that the verifier is delegating. (The protocol from [FK17] operates in the measurement-based quantum computing model,[6] but we expect that protocols in the circuit model such as [Bro18] can also be implemented from RSP_V.)

If one assumes that RSP_V can be implemented at unit cost then the protocol we obtain is also more efficient than Mahadev's: for fixed soundness error, δ, the number of operations performed by the prover scales linearly in the size of the delegated circuit and polynomially in the security parameter of the protocol. Unfortunately, our current version of RSP does not have unit cost. Furthemore, the number of uses of RSP required is linear in the circuit size, T. This implies that each use must be implemented with error $O(\delta/T)$. With our current analysis, assuming we take δ to be a constant, this results in a total communication that scales as $O(T^4)$. This is not as good as the quasi-linear complexity of prepare-and-measure protocols that use quantum communication. It is important to note, however, that the added overhead of the protocol stems from RSP_V. Thus, any improvement in the complexity of doing the state preparation will lead to an improvement in the complexity of the resulting DQC protocol. We believe reducing the overhead of RSP_V is possible and mention a potential way of achieving this in Section V below.

Before proceeding with more details of our approach it may be useful to briefly address the following question: can one use the protocol from [Mah18b] directly to implement RSP? Specifically, couldn't one enforce that the prover prepares a small-eigenvalue eigenstate of the Hamiltonian $H_\theta = -|+_\theta\rangle\langle+_\theta|$? In fact it is not at all straightforward to do this. The reasons are related to aspects of the Mahadev protocol discussed earlier. First, the committment procedure results in a state that can be measured in one of two possible bases, but it is not clear if any other form of computation besides a direct measurement can be performed on the committed qubit. Second, the guarantee provided is only that the state "exists" (i.e. the Hamiltonian has a small-eigenvalue eigenstate), but not that the state has actually been prepared by the prover. Finally, the information that the prover may have about the state it prepared is not explicitly limited (in the protocol from [Mah18b] the prover learns a classical description of the Hamiltonian, hence, in this case, the value of θ); forcing the prover to prepare an unknown state may require adding an additional layer of (quantum) homomorphic encryption to the protocol.

III. Remote state preparation: ideal resource

We formulate our variant of RSP as a *resource* in the abstract cryptography framework [MR11]. Abstract cryptography (AC), similar to universal composability (UC) [Can01], is a framework for proving the security of cryptographic protocols in a way that ensures that the protocols can be securely composed in arbitrary ways. Informally, the idea is to argue that a given protocol, which we refer to as the *real protocol*, is indistinguishable from an *ideal functionality* (or *resource*) that captures precisely what honest or dishonest parties should be able to achieve in the protocol. This involves proving two things: *correctness*, meaning that any efficient family of circuits (known as a *distinguisher*) that interacts either with an honest run of the real protocol or with the ideal functionality has a negligible advantage in deciding which it is interacting with; *security*, meaning that any attack that a malicious party could perform in the real protocol can be mapped to an attack on the ideal functionality. This latter property is formalized by saying that there exists an efficient family of (quantum) circuits, known as a *simulator*, such that any distinguisher interacting with the ideal functionality and the simulator, or with the real protocol involving only the honest parties, has negligible advantage in deciding which it is interacting with. Showing that such a simulator exists is usually the main difficulty in proving security in AC. Since the existing results on the composability of DQC protocols are expressed in the AC framework, we also present our results in AC. For more details on the framework we refer to [MR11], [DFPR14]. For the purposes of this introduction we assume basic familiarity with the framework.

We denote our variant of the ideal RSP by RSP_V, for *random Remote State Preparation with Verification*. The name is chosen in direct analogy to the resource RSP_B of *random Remote State Preparation with Blindness* introduced in [DK16]. The resource RSP_V is represented schematically in Figure 1. In the resource, Alice inputs a bit $W \in \{X, Z\}$ that denotes a measurement basis, computational ($W = Z$) or Hadamard ($W = X$). Bob inputs a bit $c \in \{0, 1\}$ that denotes honest ($c = 0$) or malicious ($c = 1$) behavior. If $c = 0$ then in the case

[6] It should be noted that the translation from the circuit model to MBQC incurs only a linear increase in overhead and this is also true for the protocol from [FK17], as explained in [KW17].

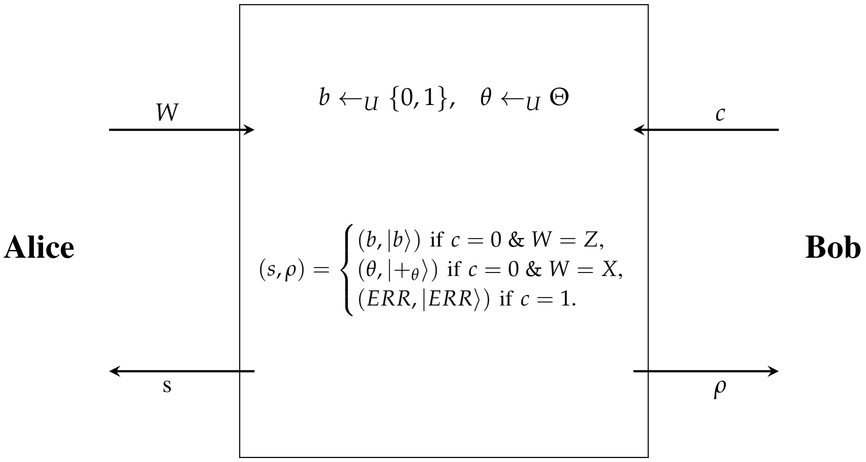
Alice
Bob
W
c
$b \leftarrow_U \{0,1\}, \quad \theta \leftarrow_U \Theta$
$(s,\rho) = \begin{cases} (b, |b\rangle) \text{ if } c = 0 \ \& \ W = Z, \\ (\theta, |+_\theta\rangle) \text{ if } c = 0 \ \& \ W = X, \\ (ERR, |ERR\rangle) \text{ if } c = 1. \end{cases}$
s
ρ

Figure 1: The resource RSP_V. It chooses b uniformly at random from $\{0,1\}$ and θ uniformly at random from $\Theta = \{0, \frac{\pi}{4}, \dots, \frac{7\pi}{4}\}$. It takes $W \in \{X, Z\}$ as input from Alice and $c \in \{0,1\}$ as input from Bob. When $c = 0$ it outputs either b to Alice and $|b\rangle$ to Bob, if $W = Z$; or θ to Alice and $|+_\theta\rangle$ to Bob, if $W = X$. When $c = 1$ it outputs ERR to Alice and $|ERR\rangle$ to Bob.

when $W = Z$ Alice receives a uniformly random bit $b \in \{0,1\}$ and Bob receives the state $|b\rangle$; in the case when $W = X$ Alice receives a uniformly random value $\theta \in \Theta = \{0, \frac{\pi}{4}, \dots, \frac{7\pi}{4}\}$ and Bob receives the state $|+_\theta\rangle$. If $c = 1$ both Alice and Bob receive an ERR message, indicating abort.

Note that the resource RSP_V can almost be understood as a communication channel from Alice to Bob that would allow Alice to select one of 10 possible single-qubit states $|0\rangle, |1\rangle$, or $|+_\theta\rangle$ for $\theta \in \Theta$ and send it to Bob. There are two differences: first, Alice does not choose the state, but instead the functionality chooses it uniformly at random and tells Alice what it is. Second, Bob may decide to block the channel, in which case both parties receive an error message. This in contrast with the weaker resource of RSP_B, also introduced in [DK16] and for which [CCKW18] give a real protocol with security against "honest-but-curious" adversaries, in which Bob is allowed to select the family of states $\{\rho_\theta\}$ that it receives (by explicitly specifying them to the resource).[7] The resource RSP_V allows less flexibility to a dishonest user, making it more useful as a building block. In particular, the rigidity of Bob's output state is essential to obtain a protocol that is verifiable.

IV. Remote state preparation: real protocol

In the previous section we defined the ideal functionality for remote state preparation with verifiability, RSP_V. In this section we describe a protocol that we prove is computationally indistinguishable from the ideal functionality. The protocol builds on ideas from [BCM$^+$18] and [Mah18b]. The main difficulty in the implementation of RSP_V is to obtain *verifiability*, i.e. the guarantee that an *arbitrary* (computationally bounded) prover successfully interacting with the verifier *must* have prepared locally the correct state, and yet have obtained no more information (computationally) about the state itself than could be gained had the state been sent directly by the verifier (or the ideal resource). To achieve this we significantly strengthen the rigidity argument from [BCM$^+$18] by giving more control, and freedom, to the verifier in the kinds of states that are prepared.

In the real protocol, that we call the *buffered remote state preparation protocol* (BRSP), Alice and Bob interact through two communication resources: a classical channel as well as a *measurement buffer*. The measurement buffer takes as input a classical message M from Alice, and from Bob a specification (as a quantum circuit) of a measurement for each of the possible messages of Alice, as well as a state on which the measurement is to be performed (as a quantum state). The buffer then performs the measurement associated with Alice's message, forwards the outcome to Alice, and returns the post-measurement state to Bob.

The necessity of relying on a measurement buffer to obtain a secure protocol is a consequence of the use of rigidity to obtain verifiability. Rigidity arguments require the assumption that, in an execution of the real

[7] The ρ_θ should satisfy the consistency condition $\rho_\theta + \rho_{\pi+\theta} = \mathrm{Id}$, which says that it is possible to generate the state ρ_θ by performing a θ-dependent measurement on a fixed state ρ; we refer to [DK16] for details.

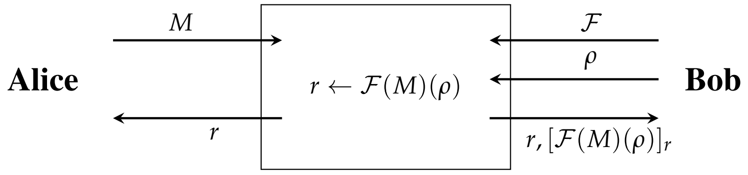
Alice
M
r
$r \leftarrow \mathcal{F}(M)(\rho)$
$\mathcal{F}$
ρ
$r, [\mathcal{F}(M)(\rho)]_r$
Bob

Figure 2: The measurement buffer. Alice inputs a message M. Bob inputs a specification $\mathcal{F}$ which takes as input Alice's message and returns a measurement $\mathcal{F}(M)$. Bob also inputs a state ρ. The buffer measures ρ with $\mathcal{F}(M)$ producing classical outcome r and the post-measurement state denoted $[\mathcal{F}(M)(\rho)]_r$. Both Alice and Bob receive r and Bob also receives $[\mathcal{F}(M)(\rho)]_r$.

protocol, the measurements implemented by Bob are "local"; in other words, that the simulator constructed in the security proof can interact directly with those measurements. In the AC framework, in general, a malicious Bob may "delegate" any measurements that it wishes to make to the environment[8], which would render them inaccessible to the simulator. By constructing the protocol from a measurement buffer resource we explicitly prevent such behavior from Bob. (Note that the use of the buffer does *not* prevent Bob from sharing entanglement with the environmnent, or from exchanging quantum states with it in-between any two uses of the measurement buffer.) While the measurement buffer is necessary to obtain composable security, the use of this resource can be omitted when considering stand-alone security only (since in that case, there is no environment). Finally, note that the measurement buffer is not a "physical" resource of the protocol; in an actual run of the protocol Alice and Bob interact only classically.

We proceed with an informal description of the protocol and its analysis. Our starting point is the work [BCM$^+$18], in which the authors give a classical protocol between a verifier and prover such that provided the prover is accepted with non-negligible probability in the protocol, it is guaranteed that a subset of the values returned by the prover contain information-theoretic randomness. This guarantee holds as long as the prover is computationally bounded, and more specifically that it does not have the ability to break the learning with errors (LWE) problem while the protocol is being executed.

We observe that the proof of [BCM$^+$18] explicitly establishes a stronger rigidity statement whereby the prover is guaranteed, up to a local rotation on its workspace, to have prepared a $|+\rangle$ state and measured it in the computational basis (hence the randomness). Formulated differently, the protocol from [BCM$^+$18] implements a weak variant of RSP$_V$ in which only the option $W = Z$ is available to Alice. This is not sufficient for delegated computation, but it is a starting point.

To generate the other states needed for RSP$_V$ we need to go deeper in the protocol from [BCM$^+$18]. At a high level, the idea is to engineer the preparation of a state of the form

$$\frac{1}{\sqrt{2}}\big(\,|0\rangle\,|x_0\rangle + |1\rangle\,|x_1\rangle\,\big)\,, \tag{1}$$

where $x_0, x_1 \in \{0,1\}^w$ are bitstrings defined as the unique preimages of an element y, provided by the prover to the verifier, under a claw-free pair of functions $f_0, f_1 : \{0,1\}^w \to \mathcal{Y}$, where $\mathcal{Y}$ is some finite range set. For the purposes of this discussion it is not important how the state (1) is obtained, as long as we can guarantee that the prover prepares such a state.

In [BCM$^+$18] the next step is to ask the prover to measure the second register in the Hadamard basis (i.e. implement the Fourier transform over $\mathbb{Z}_2^w$ and then measure in the computational basis). Labeling the outcome as $d \in \{0,1\}^w$, the first qubit is projected to the state $\frac{1}{\sqrt{2}}(-1)^{d\cdot x_0}(|0\rangle + (-1)^{d\cdot(x_0 \oplus x_1)}\,|1\rangle)$ that provides the basis for the randomness generation described earlier.

Consider the following simple modification: by thinking of x_0, x_1 as elements of $\mathbb{Z}_8^{w/3}$ (assuming w is a multiple of 3) instead of $\{0,1\}^w$, we can ask the prover to implement the Fourier transform over $\mathbb{Z}_8$, yielding an outcome $d \in \mathbb{Z}_8^{w/3}$ and a post-measurement state

$$|\psi_\theta\rangle \;=\; \frac{1}{\sqrt{2}}\omega^{d\cdot x_0}(|0\rangle + \omega^{d\cdot(x_0+x_1)}\,|1\rangle)\,, \tag{2}$$

where $\omega = e^{\frac{2i\pi}{8}}$ and the addition and inner product are taken modulo 8. Up to a global phase this is precisely the state $|+_\theta\rangle$, for $\theta = \frac{\pi}{4}\, d \cdot (x_0 + x_1)$.

So far the argument establishes completeness: if Alice and Bob follow the protocol, Alice obtains an angle

[8] In AC and UC, the environment represents anything that is external to the protocols under consideration [Can01], [MR11]. This can include other protocols, other parties etc.

θ and Bob obtains the state $|+_\theta\rangle$. Moreover, using a slight extension of the adaptive hardcore bit statement from [BCM$^+$18] it is not hard to show that the value of θ is computationally indistinguishable from uniform from Bob's perspective. The main difficulty is to argue that the prover *must* have created *precisely* the state $|\psi_\theta\rangle$ in (2), and not for instance a related state such as $|\psi_{3\theta}\rangle$. Note that this would be allowed in RSP_B, but it is not in RSP_V.

In order to show that the prover must have a state that is equal, up to an isometry, to a state of the form $|\psi_\theta\rangle$ we combine rigidity arguments similar to those employed in [BCM$^+$18] with a new idea: we introduce a test that asks the prover to demonstrate that the state it has prepared implements a near-optimal $2 \mapsto 1$ quantum random access code (QRAC). A $2 \mapsto 1$ QRAC is a procedure that encodes two classical bits into a single qubit, in a way that maximizes the success probability of the following task: given a request for either the first or the second bit (chosen with equal probability), perform a measurement on the single qubit that returns the value of that bit with the highest possible probability. As shown in [ALMO08] the optimum success probability of this task is $\frac{1}{2} + \frac{1}{2\sqrt{2}}$, and is achieved by encoding the two bits in one of the four single-qubit states $|+_0\rangle, |+_{\frac{\pi}{2}}\rangle, |+_\pi\rangle$ and $|+_{\frac{3\pi}{2}}\rangle$. More specifically, if the input bits are denoted b_1, b_2, then the QRAC state is $|+_{b_1\pi+b_2\frac{\pi}{2}}\rangle$. Moreover, the optimal measurement for predicting one bit or the other is a measurement in the basis $\{|+_{\frac{\pi}{4}}\rangle, |+_{\frac{5\pi}{4}}\rangle\}$, if b_1 is requested, or $\{|+_{\frac{3\pi}{4}}\rangle, |+_{\frac{7\pi}{4}}\rangle\}$, if b_2 is requested.

We extend the optimality proof from [ALMO08] to show that even a near-optimal family of states and measurements must be close, up to a global rotation, to the ones described above (see also [TKV$^+$18], [FK19] for similar results). Next we enforce that the prover's states and measurements implement a near-optimal $2 \mapsto 1$ QRAC by asking that the prover successfully predict certain bits of θ, given partial information about it. For example, the verifier can reveal to the prover that $\theta \in \{\frac{\pi}{2}, \frac{3\pi}{2}\}$ and ask which is the case; the prover should be able to answer with probability 1 by performing the appropriate measurement. Or the verifier can reveal that $\theta \in \{|+_0\rangle, |+_{\frac{\pi}{2}}\rangle, |+_\pi\rangle, |+_{\frac{3\pi}{2}}\rangle\}$ and ask the prover to guess one additional bit of θ; the prover should be able to succeed with probability $\frac{1}{2} + \frac{1}{2\sqrt{2}}$.

Making use of the rigidity argument to establish composable security requires the simulator to have access to Bob's measurement operators. For this reason, while most communication steps of the protocol can be implemented using a classical communication channel, in the last step of the protocol, described in the previous paragraph, the communication takes place through a measurement buffer: Alice inputs partial information about θ, and Bob inputs a description of the measurement that he would have performed on each of Alice's possible questions, together with the quantum state on which the measurement is to be performed.

We introduce a sequential protocol that consists of a number N of tests, followed by a random stopping time. We show that any behavior of the prover that has non-negligible probability of passing a fraction of tests that is within a small enough constant of the optimal fraction is such that the following property holds: at the end of the protocol, the state of the prover is unitarily equivalent to a state that is computationally indistinguishable (up to a small computational error that depends on N and other parameters of the protocol) from a state of the form $|\psi_\theta\rangle$ together with some θ-independent side information.

V. Delegated quantum computation

Having defined the ideal RSP_V functionality as well as the real protocol that implements this functionality from classical channels, we now discuss applications. As mentioned, the most natural application of RSP is to verifiable delegated quantum computation. Intuitively, the idea is the following: suppose Alice wishes to delegate $C(x)$ to Bob, for some quantum circuit C having T gates. Using the measurement-based protocol from [FK17], if Alice were to send Bob $O(T\log(1/\delta))$ randomly chosen states, from the ten possible choices mentioned earlier (the $|+_\theta\rangle$ states, with $\theta \in \Theta$, and the $|0\rangle$, $|1\rangle$ states), she would be able to delegate $C(x)$ to Bob and the protocol would have soundness error at most δ. The RSP_V functionality allows her to do exactly this, using classical communication alone. Of course, unlike the protocol of [FK17], the security of this construction would be computational, rather than information-theoretic. To summarize, in the delegation protocol Alice first executes RSP_V a certain number of times with Bob in order to prepare the required resource states in Bob's quantum memory. She then engages in the protocol of [FK17] as if she had sent the random states to Bob.

How many times does Alice need to execute RSP_V? To delegate the circuit of size T and achieve soundness error δ, the number of executions must clearly be at least $\Omega(T\log(1/\delta))$. If the real protocol used to implement RSP_V prepared the intended states *exactly*, then we would have exactly that many runs. Of course, this is not the case, and we need to account for the failure probability of the real protocol, which we denote as ε. It was shown in [DFPR14], [GKW15] that the protocol of [FK17] is robust to deviations in the collective state of the resource qubits. If there are M such qubits, and the error per state is ε then by the triangle inequality it follows that the deviation of the whole state is at most $M\varepsilon$. We therefore need to choose $\varepsilon = O(\delta/M)$ and since $M = \Omega(T \log(1/\delta))$, this means that $\varepsilon = O\left(\frac{\delta}{T \log(1/\delta)}\right)$. To achieve error at most ε, the real protocol associated to RSP_V must have a running time[9] of $O(1/\varepsilon^3)$. Putting everything together, this leads to a total number of operations that scales as $O((T^4/\delta^3) \log^4(1/\delta))$.

Ideally, one may hope for an implementation whose communication is linear in T. Some potential avenues for achieving this are as follows. First, it may be possible to have a single-use parallel version of our protocol, whereby all states would be generated in a single iteration. The verifier would send all challenges for multiple remote state preparations to the prover at once, rather than sequentially. Of course, while this would reduce the number of rounds of interaction it would not reduce the total amount of communication. An alternative is to modify the QRAC procedure. Instead of using a $2 \mapsto 1$ QRAC to verify the preparation of single-qubit states, one could use an $m \mapsto 1$ QRAC, for some sufficiently large m, to verify the preparation of multiple qubits at the same time. A second avenue for reducing communication is to develop a protocol for verifiable delegated quantum computation in which the verifier sends single-qubit states to the prover and such that the protocol is robust to constant deviations in the fidelity of these qubits. If such a protocol, with linear communication complexity, were to exist, it would be possible to use RSP_V with a constant ε and hence achieve linear classical communication. A version of such a protocol was proposed in [GKW15], though that protocol only works for specific deviations (depolarizing noise). All of these approaches seem to be technically challenging, and we leave the possibility of a more communication efficient protocol open for future work.

In the language of AC, the ideal functionality for verifiable DQC has already been defined in [DK16]. What we show is that this functionality is computationally indistinguishable from the real protocol described earlier. To do this we first adapt the definitions of DQC resources to the setting of computational security. We then show that the results pertaining to those resources in the information-theoretic case also hold in the case of computational security. Finally, we show that the RSP_V functionality can be used to implement the computational DQC functionalities. It follows that the real protocol we described is computationally indistinguishable from the ideal DQC resource.

As already mentioned one of the main advantages to proving the security of RSP_V in the AC framework is that one can directly plug this primitive into other existing protocols. Aside from DQC, a related application is to *multi-party quantum computation* (MPQC). In [KP17] the authors define AC functionalities for multi-party quantum computation. Their protocol consists of a number of clients, each having its own input, that wish to delegate a computation on their collective inputs to a quantum server. Its security, as defined in [KP17], is guaranteed in the settings where either the server is malicious (but the clients are not), or a subset of clients is malicious (but the server behaves honestly). The protocol works by having the clients perform a remote state preparation protocol, in which the clients send quantum states to the server. It then proceeds in a manner similar to the single-client DQC protocols. In principle, remote state preparation could be replaced with our RSP_V primitive, leading to an MPQC protocol in which the clients and the server use only classical communication. We leave the formalization of this intuition to future work.

VI. Acknowledgment

We thank Rotem Arnon-Friedman, Vedran Dunjko, Urmila Mahadev and Christopher Portmann for useful discussions. Alexandru Gheorghiu and Thomas Vidick are supported by MURI Grant FA9550-18-1-0161 and the IQIM, an NSF Physics Frontiers Center (NSF Grant PHY-1125565) with support of the Gordon and Betty Moore Foundation (GBMF-12500028). Thomas Vidick is also supported by NSF CAREER Grant CCF-1553477, AFOSR YIP award number FA9550-16-1-0495, and a CIFAR Azrieli Global Scholar award.

References

[ABE10] Dorit Aharonov, Michael Ben-Or, and Elad Eban. Interactive proofs for quantum computations. In *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 5-7, 2010. Proceedings*, pages 453–469, 2010.

[ABOEM17] Dorit Aharonov, Michael Ben-Or, Elad Eban, and Urmila Mahadev. Interactive Proofs for Quantum Computations. *Arxiv preprint arXiv:1704.04487*, 2017.

[9] It should be noted that our analysis for the dependency on ε is not optimal and could be improved, thereby reducing the running time of $O(1/\varepsilon^3)$.

[ALMO08] Andris Ambainis, Debbie Leung, Laura Mancinska, and Maris Ozols. Quantum random access codes with shared randomness. *arXiv preprint arXiv:0810.2937*, 2008.

[BC18] Johannes Bausch and Elizabeth Crosson. Analysis and limitations of modified circuit-to-Hamiltonian constructions. *Quantum*, 2:94, September 2018.

[BCM$^+$18] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 320–331. IEEE, 2018.

[BFK09] Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. Universal blind quantum computation. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 517–526. IEEE, 2009.

[Bro18] Anne Broadbent. How to verify a quantum computation. *Theory of Computing*, 14(1):1–37, 2018.

[BSCG$^+$13] Eli Ben-Sasson, Alessandro Chiesa, Daniel Genkin, Eran Tromer, and Madars Virza. Snarks for c: Verifying program executions succinctly and in zero knowledge. In *Advances in Cryptology–CRYPTO 2013*, pages 90–108. Springer, 2013.

[Can01] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Proceedings 2001 IEEE International Conference on Cluster Computing*, pages 136–145. IEEE, 2001.

[CCKW18] Alexandru Cojocaru, Léo Colisson, Elham Kashefi, and Petros Wallden. Delegated pseudo-secret random qubit generator. *arXiv preprint arXiv:1802.08759*, 2018.

[Chi05] Andrew M Childs. Secure assisted quantum computation. *Quantum Information & Computation*, 5(6):456–466, 2005.

[DFPR14] Vedran Dunjko, Joseph F Fitzsimons, Christopher Portmann, and Renato Renner. Composable security of delegated quantum computation. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 406–425. Springer, 2014.

[DK16] Vedran Dunjko and Elham Kashefi. Blind quantum computing with two almost identical states. *arXiv preprint arXiv:1604.01586*, 2016.

[FHM18] Joseph F Fitzsimons, Michal Hajdušek, and Tomoyuki Morimae. Post hoc verification of quantum computation. *Physical review letters*, 120(4):040501, 2018.

[FK17] Joseph F Fitzsimons and Elham Kashefi. Unconditionally verifiable blind quantum computation. *Physical Review A*, 96(1):012303, 2017.

[FK19] Máté Farkas and Jędrzej Kaniewski. Self-testing mutually unbiased bases in the prepare-and-measure scenario. *Physical Review A*, 99(3):032316, 2019.

[GGPR13] Rosario Gennaro, Craig Gentry, Bryan Parno, and Mariana Raykova. Quadratic span programs and succinct nizks without pcps. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 626–645. Springer, 2013.

[GKK17] Alexandru Gheorghiu, Theodoros Kapourniotis, and Elham Kashefi. Verification of quantum computation: An overview of existing approaches. *Theory of computing systems*, pages 1–94, 2017. Arxiv preprint arXiv:1709.06984.

[GKR15] Shafi Goldwasser, Yael Tauman Kalai, and Guy N Rothblum. Delegating computation: interactive proofs for muggles. *Journal of the ACM (JACM)*, 62(4):27, 2015.

[GKW15] Alexandru Gheorghiu, Elham Kashefi, and Petros Wallden. Robustness and device independence of verifiable blind quantum computing. *New Journal of Physics*, 17(8):083040, 2015.

[Gro10] Jens Groth. Short non-interactive zero-knowledge proofs. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 341–358. Springer, 2010.

[GV19] Alexandru Gheorghiu and Thomas Vidick. Computationally-secure and composable remote state preparation. *arXiv preprint arXiv:1904.06320*, 2019.

[HM15] Masahito Hayashi and Tomoyuki Morimae. Verifiable measurement-only blind quantum computing with stabilizer testing. *Physical review letters*, 115(22):220502, 2015.

[HR18] Justin Holmgren and Ron Rothblum. Delegating computations with (almost) minimal time and space overhead. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 124–135. IEEE, 2018.

[Kil92] Joe Kilian. A note on efficient zero-knowledge proofs and arguments. In *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*, pages 723–732. ACM, 1992.

[KP17] Elham Kashefi and Anna Pappa. Multiparty delegated quantum computing. *Cryptography*, 1(2):12, 2017.

[KRR14] Yael Tauman Kalai, Ran Raz, and Ron D Rothblum. How to delegate computations: the power of no-signaling proofs. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 485–494. ACM, 2014.

[KSV02] Alexei Yu Kitaev, Alexander Shen, and Mikhail N Vyalyi. *Classical and quantum computation*. Number 47. American Mathematical Soc., 2002.

[KW17] Elham Kashefi and Petros Wallden. Optimised resource construction for verifiable quantum computation. *Journal of Physics A: Mathematical and Theoretical*, 50(14):145306, 2017.

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

[Mah18b] Urmila Mahadev. Classical verification of quantum computations. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 259–267. IEEE, 2018.

[MR11] Ueli Maurer and Renato Renner. Abstract cryptography. In *In Innovations in Computer Science*. Citeseer, 2011.

[RUV13] Ben W Reichardt, Falk Unger, and Umesh Vazirani. Classical command of quantum systems. *Nature*, 496(7446):456, 2013.

[TKV$^+$18] Armin Tavakoli, Jędrzej Kaniewski, Tamás Vértesi, Denis Rosset, and Nicolas Brunner. Self-testing quantum states and measurements in the prepare-and-measure scenario. *Physical Review A*, 98(6):062307, 2018.