Computationally-secure and composable remote state preparation

Alexandru Gheorghiu*[1] and Thomas Vidick†[1]

[1] *Department of Computing and Mathematical Sciences, California Institute of Technology*

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 protocol realizes the following functionality, with computational security: the verifier chooses one of the observables Z, X, Y, $(X+Y)/\sqrt{2}$, $(X-Y)/\sqrt{2}$; the prover receives a uniformly random eigenstate of the observable chosen by the verifier; the verifier receives a classical description of that state. 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" (RSP$_V$), 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 an 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).

*Email: andrugh@caltech.edu

†Email: vidick@cms.caltech.edu

Contents

1 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 [GKK] 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 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 [Chi01], who gave such a protocol with quantum communication. Verifiable delegation of quantum computation was formalized in [ABOE08, 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 [ABOE08, 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 [ABOE08] 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

[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.

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 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?

1.1 Our results

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,

[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" [KSVV02]).

[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.

expressed as a computation in the measurement-based model of computation, together with a sequential pre-processing 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 $\theta \in \Theta = \{0, \frac{\pi}{4}, \dots, \frac{7\pi}{4}\}$ 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 actual functionality is slightly more complicated; see Section 1.2 and Figure 1.)

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 trivial 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 (see Theorem 4.2 for a formal statement and Section 1.2 for a definition of RSP_V):

Theorem 1.1 (Informal)**.** *Assuming the learning with errors problem is computationally intractable for efficient quantum algorithms, there exists a protocol with classical communication that implements the functionality* RSP_V *within distance* $\varepsilon > 0$ *and which has* $O(1/\varepsilon^3)$ *communication complexity.*

Here, by "implements within distance ε", we mean that any efficient quantum circuit 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$^+$18b, Mah18b] as well as from the literature on rigidity, self-testing, and quantum random access codes. Since this is our main technical contribution, we explain the protocol in more detail in Section 1.3 below.

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

[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.

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 in [DK16], having a remote state preparation functionality immediately yields a blind and verifiable protocol for DQC with classical communication and computational soundness (we explain this in more detail in Section 1.4). 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; see Section 1.4 for a discussion.)

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 $O(T^4)$ (see Section 5 for a more fine-grained analysis). 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 1.4 below. We also note that our 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]. The protocol allows for a constant fraction of failed tests, so that a partially faulty device may in principle be used to implement the protocol successfully.

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 $H_\theta = -\left|+_\theta\right\rangle\left\langle+_\theta\right|$? 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.

1.2 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

[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].

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 Section 2.4 and [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 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 $\theta \in \Theta = \{0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4}\}$ and Bob receives the state $|+_\theta\rangle$. If $c = 1$ both Alice and Bob receive an ERR message, indicating abort.

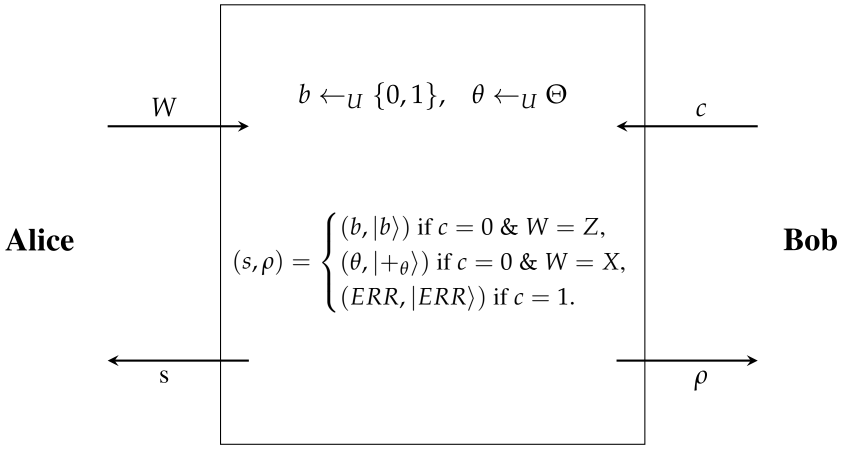
$b \leftarrow_U \{0,1\}, \quad \theta \leftarrow_U \Theta$
W
c
Alice
Bob
$(s, \rho) = \begin{cases} (b, |b\rangle) \text{ if } c = 0 \,\&\, W = Z, \\ (\theta, |+_\theta\rangle) \text{ if } c = 0 \,\&\, W = X, \\ (ERR, |ERR\rangle) \text{ if } c = 1. \end{cases}$
s
ρ

Figure 1: The resource RSP_V. It chooses b uniformly at random from $\{0, 1\}$ and θ uniformly at random from $\Theta = \{0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4}\}$. 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.

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.

1.3 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+18b] 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+18b] 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.

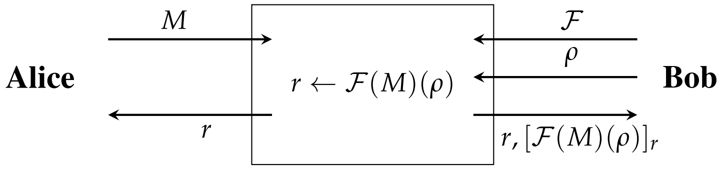
M
Alice
r
$r \leftarrow \mathcal{F}(M)(\rho)$
$\mathcal{F}$
ρ
Bob
$r, [\mathcal{F}(M)(\rho)]_r$

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 $[\mathcal{F}(M)(\rho)]_r$.

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 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

[7] The ρ_θ should satisfy the consistency condition shown in (6), 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.

[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.

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. The buffer is discussed in more detail in Section 4.

We proceed with an informal description of the protocol and its analysis. Our starting point is the work [BCM$^+$18b], 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$^+$18b] 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$^+$18b] 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$^+$18b]. 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$^+$18b] 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 θ and Bob obtains the state $|+_\theta\rangle$. Moreover, using a slight extension of the adaptive hardcore bit statement from [BCM$^+$18b] 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+18b] 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. 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.

The complete argument is given in Section 3. 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.

1.4 Application: delegated 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 $\varepsilon = O\left(\frac{\delta}{T\,\log(1/\delta)}\right)$. As shown in Section 3, to achieve error at most ε, the real protocol associated to RSP_V must have a running time 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. It may be possible to do this by considering a single-use parallel version of our protocol, whereby all states would be generated in a single iteration. Achieving this is likely to be technically challenging, and we leave the possibility 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. This is done in Section 5. 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.

Upon completion of this work we became aware of the independent work “QFactory: classically-instructed remote secret qubits preparation” by Cojocaru, Colisson, Kashefi and Wallden. Using our terminology, their main result is the design of a protocol for RSP_B, the blind variant of RSP, that they prove computationally secure.

Outline. We start with Section 2, which contains the preliminaries for this work. Most notably, in this section we recast some of the definitions pertaining to composability of DQC protocols, expressed in the AC framework, in the setting of computational, rather than information-theoretic security. Then, in Section 3, we describe the remote state preparation protocol and prove the rigidity statement about its functionality. In other words we show that, provided the verifier accepts with non-negligible probability, the prover's state is close (up to an isometry) to the ideal random state that the verifier receives a description of. In the proof, we make use of an extended noisy trapdoor claw-free function family, for which we provide the relevant definitions in Appendix A, as well as present the properties of these functions that we require. Next, in Section 4 we describe the ideal RSP_V functionality and prove, in the AC framework, that the protocol from Section 3 implements this functionality from classical channels, under computational assumptions. Having done this, we end in Section 5 by showing that the ideal RSP_V functionality can be used to implement the functionality for blind and verifiable delegated quantum computation. From the previous results, this implies that one can have a computationally secure DQC protocol by using our RSP primitive to prepare the quantum states used by that protocol. The specific DQC protocol we consider is the one from [FK17].

Acknowledgments. 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.

2 Preliminaries

2.1 Notation

We write $\mathcal{H}$ for a finite-dimensional Hilbert space, using indices $\mathcal{H}_A$, $\mathcal{H}_B$ to specify distinct spaces. $\mathrm{L}(\mathcal{H})$ is the set of linear operators on $\mathcal{H}$. We write $\mathrm{Id}_A \in \mathrm{L}(\mathcal{H}_A)$ for the identity operator, $\mathrm{Tr}(\cdot)$ for the trace, and $\mathrm{Tr}_B : \mathrm{L}(\mathcal{H}_A \otimes \mathcal{H}_B) \to \mathrm{L}(\mathcal{H}_A)$ for the partial trace. $\mathrm{Pos}(\mathcal{H})$ is the set of positive semidefinite operators and $\mathrm{D}(\mathcal{H}) = \{X \in \mathrm{Pos}(\mathcal{H}) : \mathrm{Tr}(X) = 1\}$ the set of density matrices (also called states).

Given $A \in \mathrm{L}(\mathcal{H})$, $\|A\|_1 = \mathrm{Tr}\sqrt{A^\dagger A}$ is the Schatten 1-norm and $TD(A) = \frac{1}{2}\|A\|_1$ the trace distance.

Given $X, Z \in \mathrm{L}(\mathcal{H})$ we write $\{X, Z\} = XZ + ZX$ for the anticommutator and $[X, Z] = XZ - ZX$ for the commutator. $\sigma_X, \sigma_Y, \sigma_Z \in \mathrm{L}(\mathbb{C}^2)$ are the single-qubit Pauli matrices. For an angle θ we let $\sigma_{X,\theta} = \cos\theta\,\sigma_X + \sin\theta\,\sigma_Y$.

A completely positive trace-preserving (CPTP) map $\mathcal{F} : \mathcal{H}_A \to \mathcal{H}_B$ is a linear map such that for any $\mathcal{H}_C$ and $\rho \in \mathrm{Pos}(\mathcal{H}_A \otimes \mathcal{H}_C)$ it holds that $(\mathcal{F} \otimes \mathrm{Id}_C)(\rho) \in \mathrm{Pos}(\mathcal{H}_B \otimes \mathcal{H}_C)$ and $\mathrm{Tr}(\mathcal{F} \otimes \mathrm{Id}_C(\rho)) = \mathrm{Tr}(\rho)$.

We let $\Theta = \{0, \frac{\pi}{4}, \frac{2\pi}{4}, \ldots, \frac{7\pi}{4}\}$. For $\theta \in \Theta$, $|+_\theta\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$. We often identify elements of $\mathbb{Z}_8$ with $\{0, 1, 2, \ldots, 7\}$. For a finite set S we write $x \leftarrow_U S$ to mean that x is chosen uniformly at random from S. A negligible function is a function $\delta : \mathbb{N} \to \mathbb{R}$ that goes to 0 faster than any inverse polynomial, i.e. $p(\lambda)\delta(\lambda) \to_{\lambda\to\infty} 0$ for any polynomial p.

2.2 Efficient states and operations

Definition 2.1. We say that a family of states $\{\rho_\lambda \in \mathrm{D}(\mathcal{H}_{A_\lambda})\}_{\lambda\in\mathbb{N}}$ is efficiently preparable (or just "efficient") if there exists a polynomial-time uniformly generated[9] family of circuits $\{C_\lambda\}_{\lambda\in\mathbb{N}}$ acting on $\mathcal{H}_{A_\lambda} \otimes \mathcal{H}_{B_\lambda}$ such that

$$\forall\lambda\,, \qquad \mathrm{Tr}_{B_\lambda}\big(C_\lambda(|0\rangle\langle 0|_{A_\lambda} \otimes |0\rangle\langle 0|_{B_\lambda})\big) = \rho_\lambda\ .$$

[9] By "polynomial-time uniformly generated we mean that there exists a Turing machine T that on input 1^λ returns a description of the circuit C_λ using some fixed finite universal gate set.

Definition 2.2. We say that a family of CPTP maps $\{\mathcal{F}_\lambda : \mathrm{L}(\mathcal{H}_{A_\lambda}) \to \mathrm{L}(\mathcal{H}_{B_\lambda})\}_{\lambda\in\mathbb{N}}$ is efficient if there exists a polynomial-time uniformly generated family of circuits $\{C_\lambda\}$ acting on $\mathcal{H}_{A_\lambda} \otimes \mathcal{H}_{B_\lambda} \otimes \mathcal{H}_{C_\lambda}$ such that

$$\forall\lambda\,, \forall\rho \in \mathrm{D}(\mathcal{H}_{A_\lambda})\,, \qquad \mathrm{Tr}_{A_\lambda C_\lambda}\big(C_\lambda(\rho \otimes |0\rangle\langle 0|_{B_\lambda C_\lambda})\big) = \mathcal{F}_\lambda(\rho_\lambda)\,.$$

2.3 Computational distinguishability

Definition 2.3. Given two families of (not necessarily normalized) density operators $\{\rho_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\sigma_\lambda\}_{\lambda\in\mathbb{N}}$ we say that ρ and σ are *computationally distinguishable with advantage at most* $\delta(\lambda)$, and write $\rho \approx_{c,\delta} \sigma$, if for any polynomial-time uniformly generated family of circuits $\{D_\lambda\}_{\lambda\in\mathbb{N}}$, known as a *distinguisher*, there is a $\lambda_0 \in \mathbb{N}$ such that

$$\forall\lambda \geq \lambda_0\,, \qquad \frac{1}{2}\big|\mathrm{Tr}\big(D_\lambda^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda\rho_\lambda\big) - \mathrm{Tr}\big(D_\lambda^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda\sigma_\lambda\big)\big| \leq \delta(\lambda)\,. \tag{3}$$

The best $\delta(\lambda)$ in Definition 2.3 implicitly depends on the specific polynomial bound that is placed on the size of the distinguisher. In this paper it will always be the case that $\delta(\lambda) = \delta + \mathrm{negl}(\lambda)$, for some constant δ and a negligible function of λ. The size of the distinguisher will affect the negligible function; the statement should be interpreted as saying that for any polynomial size bound on the distinguisher there is a negligible function of λ such that (3) holds.

Lemma 2.4. *For any density operators $\{\rho_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\sigma_\lambda\}_{\lambda\in\mathbb{N}}$, $\{\rho_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\sigma_\lambda\}_{\lambda\in\mathbb{N}}$ are computationally distinguishable with advantage at most $\|\rho_\lambda - \sigma_\lambda\|_1$.*

Proof. For any ρ, σ and $0 \leq D \leq \mathrm{Id}$ it holds that:

$$\frac{1}{2}\big|\mathrm{Tr}\big(D_\lambda^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda\rho_\lambda\big) - \mathrm{Tr}\big(D_\lambda^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda\sigma_\lambda\big)\big| \leq \|\rho_\lambda - \sigma_\lambda\|_1 \tag{4}$$

$\square$

Lemma 2.5. *For $b \in \{0,1\}$ let $\{\rho_\lambda^b\}_{\lambda\in\mathbb{N}}$ and $\{\sigma_\lambda^b\}_{\lambda\in\mathbb{N}}$ be two families of density operators. For all λ, let $\rho_\lambda = \sum_b |b\rangle\langle b| \otimes \rho_\lambda^b$ and $\sigma_\lambda = \sum_b |b\rangle\langle b| \otimes \sigma_\lambda^b$. Suppose that $\{\rho_\lambda\}$ and $\{\sigma_\lambda\}$ are distinguishable with advantage at most $\delta(\lambda)$. Then for $b \in \{0,1\}$, $\{\rho_\lambda^b\}$ and $\{\sigma_\lambda^b\}$ are distinguishable with advantage at most $\delta_b(\lambda)$ where $\delta_0(\lambda), \delta_1(\lambda)$ are such that $|\delta_0(\lambda) + \delta_1(\lambda) - \delta(\lambda)| = \mathrm{negl}(\lambda)$.*

Proof. For $b \in \{0,1\}$ fix a family of efficient distinguishers $\{D_\lambda^b\}$ for $\{\rho_\lambda^b\}$ and $\{\sigma_\lambda^b\}$ with advantage $\delta_b(\lambda)$. Then the distinguisher $D = |0\rangle\langle 0| \otimes D^0 + |1\rangle\langle 1| \otimes D^1$ is efficient and has distinguishing advantage $\delta_0(\lambda) + \delta_1(\lambda)$ for $\{\rho_\lambda\}$ and $\{\sigma_\lambda\}$. This shows that $\delta_0(\lambda) + \delta_1(\lambda) \leq \delta(\lambda)$. Conversely, let $\{D_\lambda\}$ be an efficient distinguisher for $\{\rho_\lambda\}$ and $\{\sigma_\lambda\}$ with advantage $\delta(\lambda)$. Then

$$\begin{aligned}&\big|\mathrm{Tr}(D_\lambda^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda\rho_\lambda) - \mathrm{Tr}(D_\lambda^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda\sigma_\lambda)\big| \\ &\qquad\qquad \leq \sum_b \big|\mathrm{Tr}((D_\lambda^b)^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda^b\rho_\lambda^b) - \mathrm{Tr}((D_\lambda^b)^\dagger(|0\rangle\langle 0| \otimes \mathrm{Id})D_\lambda^b\sigma_\lambda^b)\big|\,,\end{aligned}$$

where D_λ^b is the efficient distinguisher that initializes an ancilla qubit to state $|b\rangle\langle b|$ and then runs D_λ. $\square$

2.4 Composable security

Abstract cryptography (AC) is a framework for proving the security of protocols under composition. For example if protocols π_1 and π_2 are shown to be secure in the AC framework then their sequential composition $\pi_1 \circ \pi_2$ or parallel composition $\pi_1 | \pi_2$ is automatically secure as well. For an in-depth introduction to the framework of abstract cryptography specialized to the present context of two-party quantum protocols we refer to [DFPR13]. Here we briefly recall the key notions and terminology.

The actions of the two players, generally called Alice and Bob, in a two-party protocol π are specified by a sequence of CPTP maps $\pi_A = \{\mathcal{E}_i : \mathrm{L}(\mathcal{H}_{AC}) \to \mathrm{L}(\mathcal{H}_{AC})\}_i$ and $\pi_B = \{\mathcal{F}_i : \mathrm{L}(\mathcal{H}_{CB}) \to \mathrm{L}(\mathcal{H}_{CB})\}_i$, where A and B are Alice and Bob's private registers respectively, and C represents a communication channel. In AC the channel C is modeled as a resource $\mathcal{R}$, where in general a resource is itself represented as a sequence of completely positive trace-preserving (CPTP) maps with internal memory.

In the AC framework a protocol can be thought of as a process that constructs a resource, $\mathcal{S}$, from some other resource, $\mathcal{R}$. For instance, a protocol $\pi_{AB} = (\pi_A, \pi_B)$ can construct an ideal resource for delegated quantum computation from a resource consisting of classical and quantum channels. The resource $\pi_A\mathcal{R}$ obtained by plugging in one player's strategy into the resource is another resource, itself modeled as a sequence of CPTP maps, that can be thought of as a quantum strategy as defined in [GW07]. When both Alice and Bob follow the protocol while interacting with the resource $\mathcal{R}$ we write $\pi_{AB}\mathcal{R}$, or $\pi_A\mathcal{R}\pi_B$.[10] Note that $\pi_A\mathcal{R}\pi_B$ is again itself a resource, having input and output interfaces for both Alice and Bob.

Since the goal in the AC framework is to show that certain resources are indistinguishable from each other, we need a notion of distinguishability of resources. Informally, two resources $\mathcal{R}_1$ and $\mathcal{R}_2$, each modeled as a sequence of CPTP maps with input and output spaces of compatible dimension, are *computationally distinguishable with advantage at most* ε if no efficient distinguisher $\mathcal{D}$ (itself represented as a family of efficient CPTP maps) can distinguish an interaction with $\mathcal{R}_1$ from an interaction with $\mathcal{R}_2$. Here, the distinguisher is allowed to create an initial state as input to the resource (the state can be entangled with a reference system kept by the distinguisher); then, upon having received the output of the first map, it can modify it in an arbitrary (efficient) way and input it to the second map, etc., until it is required to make an (efficient) measurement on the output of the last map (and its own reference system) in order to return a guess for the resource with which it was interacting. We write the composition of $\mathcal{D}$ and $\mathcal{R}_i$, for $i \in \{1,2\}$, as $\mathcal{D}\mathcal{R}_i$; this is a resource that takes no input and outputs a single bit.

Definition 2.6. Let $\varepsilon = \varepsilon(\lambda) \in [0,1]$ be a function of a security parameter $\lambda \in \mathbb{N}$, and let $\mathcal{R}_1$ and $\mathcal{R}_2$ be two resources having input and output spaces of the same dimension. We say that $\mathcal{R}_1$ and $\mathcal{R}_2$ have distinguishing advantage ε if for all efficient distinguishers $\mathcal{D}$ it holds that $|\Pr(\mathcal{D}\mathcal{R}_1 = 1) - \Pr(\mathcal{D}\mathcal{R}_2 = 1)| \leq \varepsilon$. We write this as:

$$\mathcal{R}_1 \approx_{c,\varepsilon} \mathcal{R}_2 \,. \tag{5}$$

With this definition we have the following.

Definition 2.7. Let $\varepsilon = \varepsilon(\lambda) \in [0,1]$ be a function of a security parameter $\lambda \in \mathbb{N}$. We say that a protocol $\pi = (\pi_A, \pi_B)$ constructs a resource $\mathcal{S}$ from a resource $\mathcal{R}$ with (computational) error (or distance) ε if:

- **Correctness:** $\pi_{AB}\mathcal{R} \approx_{c,\varepsilon} \mathcal{S}$.
- **Security:** There exists an efficient simulator σ such that $\pi_A\mathcal{R} \approx_{c,\varepsilon} \mathcal{S}\sigma$.

(Here π_{AB}, R, S and σ may all implicitly depend on λ.)

[10] The ordering of the protocols, π_A and π_B, has no special significance.

The first condition expresses the fact that if Alice and Bob follow the instructions of the protocol, the resulting resource behaves as the ideal one. The second condition expresses the fact that if Bob does not follow the protocol, any attack he performs on the real protocol can be mapped to an attack on the ideal protocol. This mapping is referred to as a simulator. Note that Definition 2.7 implicitly assumes that Alice always behaves honestly; this need not be the case in general but always holds in the context of this paper.

2.5 Rigidity

Definition 2.8. Let finite-dimensional Hilbert spaces $\mathcal{H}_{\mathsf{A}}$ and $\mathcal{H}_{\mathsf{A}'}$ and operators $R \in \mathrm{L}(\mathcal{H}_{\mathsf{A}})$ and $S \in \mathrm{L}(\mathcal{H}_{\mathsf{A}'})$ be functions of a parameter $\delta > 0$ (the dependence on δ is left implicit in the notation). We say that R and S are δ-isometric with respect to $|\psi\rangle \in \mathcal{H}_{\mathsf{A}} \otimes \mathcal{H}_{\mathsf{B}}$, and write $R \simeq_\delta S$, if there exists an isometry $V : \mathcal{H}_{\mathsf{A}} \to \mathcal{H}_{\mathsf{A}'}$ such that

$$\left\| (R - V^\dagger S V) \otimes \mathrm{Id}_{\mathsf{B}} |\psi\rangle \right\|^2 = O(\delta).$$

We sometimes write the isometry as a CPTP map $\Phi(R) = VRV^\dagger$ for $R \in \mathrm{L}(\mathcal{H}_A)$, and also write $\Phi(|\phi\rangle)$ for $V|\phi\rangle$, $\Phi(\sigma)$ for $V\sigma V^\dagger$. If V is the identity, then we further say that R and S are δ-equivalent, and write $R \approx_\delta S$ for $\|(R - S) \otimes \mathrm{Id}_{\mathsf{B}} |\psi\rangle\|^2 = O(\delta)$.

The following can be shown by a standard application of Jordan's lemma. Furthermore, the isometry V can be implemented using the "swap" isometry as in [MYS12].

Lemma 2.9. *Let $|\psi\rangle \in \mathcal{H}_{\mathsf{A}} \otimes \mathcal{H}_{\mathsf{B}}$ and Z, X, X' observables on $\mathcal{H}_{\mathsf{A}}$ such that $\{Z, X\} \approx_\delta 0$ and $\{Z, X'\} \approx_\delta 0$. Then there exist $\delta' = O(\sqrt{\delta})$, an isometry $V : \mathcal{H}_{\mathsf{A}} \to \mathbb{C}^2 \otimes \mathcal{H}_{\mathsf{A}'}$, and Hermitian commuting A_X, A_Y on $\mathcal{H}_{\mathsf{A}'}$ such that $A_X^2 + A_Y^2 = \mathrm{Id}$ and*

$$Z \simeq_{\delta'} \sigma_Z \otimes \mathrm{Id}, \quad X \simeq_{\delta'} \sigma_X \otimes \mathrm{Id}, \quad \text{and} \quad X' \simeq_{\delta'} \sigma_X \otimes A_X + \sigma_Y \otimes A_Y .$$

Furthermore, there exists a polynomial-time algorithm that given explicit circuits implementing Z, X and X' as input returns an explicit circuit that implements the isometry V.

2.6 Delegated quantum computation

2.6.1 Ideal functionalities

We recall the ideal resources for blind and verifiable delegated quantum computation (DQC), as defined in [DFPR13]. We start with blindness.

Definition 2.10 (Definition 4.1 in [DFPR13])**.** The ideal DQC resource $\mathcal{S}^{blind}$ which provides both correctness and blindness takes an input ψ_A at Alice's interface, but no honest input at Bob's interface. Bob's filtered interface has a control bit b, set by default to 0, which he can flip to activate the other filtered functionalities. The resource $\mathcal{S}^{blind}$ then outputs the permitted leak ℓ^{ψ_A} at Bob's interface, and accepts two further inputs, a state ψ_B and a map description $|\mathcal{E}\rangle\langle\mathcal{E}|$. If $b = 0$, it outputs the correct result $\mathcal{U}(\psi_A)$ at Alice's interface; otherwise it outputs Bob's choice $\mathcal{E}(\psi_{AB})$.

Next we give the definition for blindness and verifiability. The main difference is that the ideal functionality is no longer allowed to return an output of Bob's choice at Alice's interface.

Definition 2.11 (Definition 4.2 in [DFPR13])**.** The ideal DQC resource $\mathcal{S}^{blind}_{verif}$ which provides correctness, blindness and verifiability takes an input ψ_A at Alice's interface, and two filtered control bits b and c (set by default to 0). If $b = 0$, it outputs the correct result $\mathcal{U}(\psi_A)$ at Alice's interface. If $b = 1$, it outputs the permitted leak ℓ^{ψ_A} at Bob's interface, then reads the bit c, and conditioned on its value, it either outputs $\mathcal{U}(\psi_A)$ or $|ERR\rangle$ at Alice's interface.

We provide the definitions of the ideal resources for two variants of random remote state preparation introduced in [DK16]:

Definition 2.12 (Definition 11 in [DK16])**.** The ideal resource called the strong random remote state preparation (RSP_S) has two interfaces A, and B, standing for Alice and Bob. The resource first selects an angle θ (from the set of 8 states) chosen uniformly at random. Bob's interface has a filtered functionality comprising a bit c which Bob can pre-set to zero or one, depending on whether he will behave maliciously. If Bob pre-sets $c = 0$, the resource outputs the state $|+_\theta\rangle\langle+_\theta|$ on Bob's interface. If Bob pre-sets $c = 1$, it awaits a description of a CPTP map $\mathcal{E}$ from Bob. Once the set is received, the functionality outputs $\mathcal{E}(|+_\theta\rangle\langle+_\theta|)$ at Bob's interface. In both cases, the resource outputs the angle θ at Alice's interface.

Definition 2.13 (Definition 8 in [DK16])**.** The ideal resource called the *random remote blind state preparation for blindness* RSP_B has two interfaces A, and B, standing for Alice and Bob. The resource first selects a θ chosen uniformly at random. Bob's interface has a filtered functionality comprising a bit c which Bob can pre-set to zero or one, depending on whether he will behave maliciously. If Bob pre-sets $c = 0$, the resource outputs the state $|+_\theta\rangle\langle+_\theta|$ on Bob's interface. If Bob pre-sets $c = 1$, it awaits the set $\{(\theta, [\rho^\theta])\}_\theta$ from Bob, where $[\rho^\theta]$ denotes the classical description of a quantum state, with the property that $\rho^\theta + \rho^{\theta+\pi} = \rho^{\theta'} + \rho^{\theta'+\pi}, \forall\theta, \theta'$. If the states Bob inputs do not satisfy the property above, the ideal functionality ignores the set Bob has input and awaits a new valid set. Once the set is received, the functionality outputs ρ^θ at Bob's interface. In both cases, the resource outputs the angle θ at Alice's interface.

Briefly, the difference between RSP_S and RSP_B is as follows. In RSP_S if Alice accepts she receives a random angle θ, and Bob receives $\mathcal{E}(|+_\theta\rangle)$, for some CPTP map $\mathcal{E}$, that is *independent of* θ. In the weaker RSP_B, the possible states, $\{\rho_\theta\}_{\theta\in\Theta}$, that Bob receives should satisfy

$$\rho_\theta + \rho_{\pi+\theta} = \mathrm{Id} \ . \tag{6}$$

Importantly, there is no requirement that Bob's state is the correct $|+_\theta\rangle$ state, up to to the action of an independent CPTP map. It is precisely the possibility that the deviation map can depend on θ that makes RSP_B unsuitable for verifiability (though it does provide blindness).

2.6.2 Local criteria

Dunjko et al. [DFPR13] give "local" criteria, δ-local-blindness and independent δ-local-verifiability, that can be used to establish the security of a protocol for delegated quantum computation in the AC framework. Their definitions are geared to showing information-theoretic security. We adapt them to the setting of computational security, as follows.

Definition 2.14. A DQC protocol provides δ-local-blindness if for all efficient adversaries $\{\mathcal{F}_i : \mathrm{L}(\mathcal{H}_{CB}) \to \mathrm{L}(\mathcal{H}_{CB})\}$ in the protocol there is an efficient CPTP map $\mathcal{F} : \mathrm{L}(\mathcal{H}_B) \to \mathrm{L}(\mathcal{H}_B)$ such that for all efficiently preparable ψ_{ABR},

$$\mathrm{Tr}_A \circ \mathcal{P}_{AB}(\psi_{ABR}) \approx_{c,\delta} \mathcal{F} \circ \mathrm{Tr}_A(\psi_{ABR}) \ , \tag{7}$$

where $\mathcal{P}_{AB}$ is the map corresponding to an execution of the protocol with an honest Alice and a Bob specified by the maps $\mathcal{F}_i$ and $\circ$ denotes composition. When no map acts on a space, it is to be assumed that the identity is applied.

Definition 2.15. A DQC protocol provides independent δ-local verifiability if for all efficient adversaries $\{\mathcal{F}_i : \mathrm{L}(\mathcal{H}_{CB}) \to \mathrm{L}(\mathcal{H}_{CB})\}$ in the protocol there exist efficient alternative maps $\{\mathcal{F}'_i : \mathrm{L}(\mathcal{H}_{CBB'}) \to \mathrm{L}(\mathcal{H}_{CBB'})\}$ such that the following hold:

1. For all efficient initial states $\psi_{AR_1} \otimes \psi_{R_2B}$ there is a $0 \leq p^\psi \leq 1$ such that

$$\rho^\psi_{AR_1} \approx_{c,\delta} p^\psi(\mathcal{U} \otimes \mathrm{Id}_{R_1})(\psi_{AR_1}) + (1-p^\psi)\,|ERR\rangle\langle ERR| \otimes \psi_{R_1} \,, \tag{8}$$

   where $\rho^\psi_{AR_1}$ is the final state of Alice and the first part of the reference system;

2. For all efficient initial states ψ_{ABR},

$$\mathrm{Tr}_A \circ \mathcal{Q}_{AB'} \circ \mathcal{P}_{AB}(\psi_{ABR}) \approx_{c,\delta} \mathrm{Tr}_A \circ \mathcal{P}'_{ABB'} \,, \tag{9}$$

   where $\mathcal{P}_{AB}$ and $\mathcal{P}'_{ABB'}$ are the maps corresponding to an execution of the protocol with an honest Alice and a Bob specified by the maps $\mathcal{F}_i$ and $\mathcal{F}'_i$ respectively, and $\mathcal{Q}_{AB'} : \mathrm{L}(\mathcal{H}_A) \to \mathrm{L}(\mathcal{H}_{AB'})$ is a map which generates from A a system B' that contains a copy of the information whether Alice accepts or rejects.

The following is an analogue of [DFPR13, Corollary 6.9] for the computational setting.

Theorem 2.16. *If a DQC protocol π implementing a unitary transformation $\mathcal{U}$ is δ_c-correct and provides δ_b-local-blindness and independent δ_v-local-verifiability for all efficient inputs that are classical on A,*[11] *for some $\delta_c, \delta_b, \delta_v \geq 0$, then it constructs $\mathcal{S}^{blind}_{verif}$ computationally within $\varepsilon = \max(\delta_c, 2\delta_b + 4\sqrt{\delta_v})$.*

Proof. The proof is identical to the proof of [DFPR13, Corollary 6.9] except for ensuring that the simulator is computationally efficient. The first step is to combine local-blindness and local-verifiability to obtain the condition of local-blind-verifiability, i.e. the existence of maps $\mathcal{F}^{ok}$ and $\mathcal{F}^{ver}$ such that, using the notation from Definition 2.15,

$$\rho^\psi_{AR_1R_2B} \approx_{c,\delta} (\mathcal{U} \otimes \mathrm{Id}_{R_1R_2} \otimes \mathcal{F}^{ok})(\psi_{AR_1} \otimes \psi_{R_2B}) + |ERR\rangle\langle ERR| \otimes \psi_{R_1} \otimes (\mathrm{Id}_{R_2} \otimes \mathcal{F}^{err})(\psi_{R_2B}) \,. \tag{10}$$

The maps $\mathcal{F}^{ok}$ and $\mathcal{F}^{ver}$ can be defined from $\mathcal{F}'$ as in the proof of [DFPR13, Lemma 6.6] as

$$\mathcal{F}^{ok} = \mathrm{Tr}_{B'} \circ \mathcal{P}^{ok}_{B'} \circ \mathcal{F}' \,, \qquad \mathcal{F}^{err} = \mathrm{Tr}_{B'} \circ \mathcal{P}^{err}_{B'} \circ \mathcal{F}' \,,$$

where $\mathcal{P}^{ok}_{B'}$ and $\mathcal{P}^{err}_{B'}$ are the projection on the corresponding states of B'. Clearly these maps can be implemented efficiently, given that $\mathcal{F}'$ can. Next we need to show that any efficient distinguisher $\mathcal{D}$ for (10) contradicts either local-blindness or local-verifiability. Write

$$\rho^\psi_{AR_1R_2B} \;=\; \phi^{ok}_{ARB} + |ERR\rangle\langle ERR| \otimes \phi^{err}_{RB} \,.$$

If $\mathcal{D}$ is a distinguisher for (10) with advantage δ, by Lemma 2.5 there exist distinguishers D^{ok} between ϕ^{ok}_{ARB} and $(\mathcal{U} \otimes \mathrm{Id}_{R_1R_2} \otimes \mathcal{F}^{ok})(\psi_{AR_1} \otimes \psi_{R_2B})$, and D^{err} between ϕ^{err}_{RB} and $\psi_{R_1} \otimes (\mathrm{Id}_{R_2} \otimes \mathcal{F}^{err})(\psi_{R_2B})$, with advantage δ_1 and δ_2 respectively such that $\delta_1 + \delta_2 \geq \delta$.

Consider first the case of ϕ^{err}_{RB}. Using (7), (9) and Lemma 2.5 it follows that $\delta_2 \leq \delta_b + \delta_v$. Consider next ϕ^{ok}_{ARB}. Using the triangle inequality, $\mathcal{D}$ must distinguish between $\phi^{ok}_{AR_1R_2B}$ and $p^\psi\mathcal{U}(\psi_{AR_1}) \otimes \phi^{ok}_{R_2B}$,

[11] We make the restriction that the input is classical for convenience; a more general version of the theorem, with some loss in parameters, applies to quantum inputs. See [DFPR13, Corollary 6.9] for details.

and between $p^{\psi}\mathcal{U}(\psi_{AR_1}) \otimes \phi^{ok}_{R_2B}$ and $\mathcal{U}(\psi_{AR_1}) \otimes \mathcal{F}^{ok}_B(\psi_{R_2B})$, with advantage δ'_1 and δ'_2 respectively such that $\delta'_1 + \delta'_2 \geq \delta_1$.

Using property (8) of local verifiability and Lemma 2.5 it follows that no efficient distinguisher can distinguish $\phi^{ok}_{AR_1}$ from $p^{\psi}\mathcal{U}(\psi_{AR_1})$ with advantage larger than δ_v. Using that the state $\mathcal{U}(\psi_{AR_1})$ is efficiently preparable, a specific distinguisher would be to perform a swap test with that state. It follows that $\mathrm{Tr}(\mathcal{U}(\psi_{AR_1})\phi^{ok}_{AR_1}) \leq 2\delta_v$. Using the relation between fidelity and trace distance and Uhlmann's theorem it follows that $\|\phi^{ok}_{AR_1R_2B} - p^{\psi}\mathcal{U}(\psi_{AR_1}) \otimes \phi^{ok}_{R_2B}\|_1 \leq \sqrt{4\delta_v}$, so by Lemma 2.4 it holds that $\delta'_1 \leq \sqrt{4\delta_v}$ as well.

Finally, using again (7), (9) and Lemma 2.5 it follows that $\delta'_2 \leq \delta_b + \delta_v$.

Having established (10) for some $\delta \leq 2\delta_b + 4\sqrt{\delta_v}$, it remains to show the existence of a simulator σ_B such that $\pi \approx_{c,\delta} \mathcal{S}^{blind}_{verif}\sigma_B$. The simulator is identical to the simulator constructed in the proof of [DFPR13, Theorem 5.2]: the simulator simply interacts with Bob as Alice would in the protocol π, using an arbitrary input ψ_B instead of Alice's real input ψ_A. This simulator is clearly efficient. The remainder of the argument is exactly the same, and we omit the details. □

3 Remote state preparation: real protocol

In this section we describe and analyze our implementation of the ideal resource for random remote state preparation with verification, RSP_V. We refer to this implementation as the real protocol (to be contrasted with the ideal protocol/functionality, introduced in the next section). As mentioned in the introduction, our implementation builds upon the randomness certification protocol from [BCM+18b] by, informally, performing the protocol modulo 8 (instead of modulo 2) and adding tests inspired from the study of quantum random access codes to verify that the prover prepares the right state, up to a local isometry.

We start by recalling the definition of a QRAC in Section 3.1, and show a new result about rigidity of $2 \mapsto 1$ QRAC. In Section 3.2 we introduce the main building block for our protocol; we analyze its soundness and rigidity properties in Section 3.3. Finally, we describe and analyze our protocol for RSP_V in Section 3.4.

3.1 Quantum random access codes

Definition 3.1. A $2 \mapsto 1$ quantum random access code (QRAC) is specified by four single-qubit density matrices $\{\phi_u\}_{u\in\{1,3,5,7\}}$ and two single-qubit observables X_0 and X_2. For $u \in \{1,3,5,7\}$ let $u_0, u_2 \in \{0,1\}$ be such that $u_0 = 0$ if and only if $u \in \{1,7\}$ and $u_2 = 0$ if and only if $u \in \{1,3\}$.[12] The success probability of the QRAC is defined as

$$\frac{1}{4}\sum_{u\in\{1,3,5,7\}}\frac{1}{2}\sum_{i\in\{0,2\}}\mathrm{Tr}\big(X_i^{u_i}\phi_u\big)\;.$$

Let $\mathrm{OPT}_Q = \frac{1}{2} + \frac{1}{2\sqrt{2}}$. As shown in [ALMO08, Theorem 3], the highest possible success probability of a single-qubit $2 \mapsto 1$ QRAC is OPT_Q. More generally, we have the following rigidity statement.

Lemma 3.2. *Let $\{\phi_u\}$ and X_0, X_2 be a $2 \mapsto 1$ QRAC whose success probability is at least $(1-\delta)\mathrm{OPT}_Q$, for some $0 \leq \delta < 1$. Then*

$$\frac{1}{4}\sum_{u\in\{1,3,5,7\}}\mathrm{Tr}\big(\{X_0,X_2\}^2\phi_u\big) \;=\; O(\delta)\;.$$

[12] The motivation for the somewhat obscure indexing scheme will become clear later.

Proof. Assume without loss of generality that both observables X_0 and X_2 are in the plane specified by σ_X and σ_Y. Let $v_0 = (x_0, y_0, 0)$ and $v_2 = (x_1, y_1, 0)$ be the Bloch sphere representation of the eigenvalue-1 eigenvector of X_0 and X_2 respectively, i.e. real unit vectors such that for $i \in \{0, 2\}$, $X_i = x_i\sigma_X + y_i\sigma_Y$. As shown in [ALMO08, Section 3.4], the optimal success probability of any $2 \mapsto 1$ QRAC based on X_0 and X_2 is $\frac{1}{2}(1 + \frac{S}{8})$, where $S = 2\|v_0 + v_2\| + 2\|v_0 - v_2\|$. In order for the QRAC to achieve a success probability of $(1 - \delta)\text{OPT}_Q$ it is necessary that $\|v_0 + v_2\| + \|v_0 - v_2\| \geq 2\sqrt{2} - 16\delta$. Using $4 = \|v_0 + v_2\|^2 + \|v_0 - v_2\|^2$ it follows that $|\|v_0 + v_2\|^2 - \|v_0 - v_2\|^2| = O(\sqrt{\delta})$, thus $|v_0 \cdot v_2| = O(\sqrt{\delta})$. Since $\{X_0, X_2\} = 2iv_0 \cdot v_2 \,\text{Id}$, we obtain $\|\{X_0, X_2\}\|^2 = O(\delta)$. □

The following simple test will be used later to estimate the success probability of a QRAC. We introduce it here to set some notation.

Definition 3.3. Let $\{\phi_u\}_{u\in\{0,1,2,\dots,7\}}$ be arbitrary density matrices. In the *QRAC test*, the prover is given ϕ_u for a uniformly random $u \in \{1, 3, 5, 7\}$. The verifier sends a uniformly random $\theta \in \{0, 2\}$ to the prover, who replies with a bit v. If $v \neq u_\theta$, the verifier sets $flag \leftarrow fail_Q$.

3.2 The qubit preparation test

The *qubit preparation test* described in Figure 3 forms the main building block of our remote state preparation protocol. The test relies on an extended variant of the family of claw-free functions used in [BCM+18b], introduced in [Mah18b] and called an *extended noisy trapdoor claw-free family* (ENTCF). We recall the definition of an ENTCF family $(\mathcal{F}, \mathcal{G})$ in Appendix A, where we also present the main properties needed. For the purposes of this section it is sufficient to think of both $\mathcal{F}$ and $\mathcal{G}$ as families of pairs of functions, $(f_{k,0}, f_{k,1})$ or $(g_{k,0}, g_{k,1})$, where k denotes a public key, such that both functions in an f-pair (also called claw-free pair) are bijections with the same domain and range, while both functions in a g-pair (also called injective pair) are bijections with the same domain but non-intersecting ranges, and such that moreover given a key k it is computationally impossible to distinguish if k corresponds to a claw-free or an injective pair.

We first show a completeness property of the qubit preparation test.

Lemma 3.4 (Completeness)**.** *There is an efficient quantum prover that is accepted with probability negligibly close to* 1 *in the security parameter* λ *in each of the preimage test and part A. of the* X_θ*-measurement test, and with probability negligibly close to* OPT_Q *in part B. of the* X_θ*-measurement test (Figure 3). Moreover, in case the verifier selects* $G = 0$ *and a key* k*, after having returned a* $y \in \mathcal{Y}$ *in step 2. and an equation* $d \in \mathbb{Z}_8^w$ *at the beginning of step 3. the state of the prover is the state*

$$\frac{1}{\sqrt{2}}\left(e^{\frac{2i\pi}{8}d\cdot J(x_0)}\,|0\rangle\,|x_0\rangle + e^{\frac{2i\pi}{8}d\cdot J(x_1)}\,|1\rangle\,\right), \tag{11}$$

where x_0, x_1 *are the two preimages of* y *under* $f_{k,0}$ *and* $f_{k,1}$ *respectively and* J *is a simple map described in Appendix A.*

Proof. The honest strategy for the prover is as follows. Upon receipt of a key k that specifies a pair of functions $f_{k,0}$ and $f_{k,1}$ the prover prepares a state $|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, adjoins a uniform superposition over all $x \in \mathcal{X}$, evaluates f in superposition, and measures the outcome y. The result is the state

$$\frac{1}{\sqrt{2}}\big(\,|0\rangle\,|x_0\rangle + |1\rangle\,|x_1\rangle\,\big)\,, \tag{12}$$

Let λ be a security parameter.

1. The verifier selects $G \leftarrow_U \{0,1\}$. If $G = 0$ they sample a key $(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{F}}(1^\lambda)$. If $G = 1$ they sample $(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{G}}(1^\lambda)$. The verifier sends k to the prover and keeps the trapdoor information t_k private.

2. The prover returns a $y \in \mathcal{Y}$ to the verifier. If $G = 0$, for $b \in \{0,1\}$ the verifier uses the trapdoor to compute $\hat{x}_b \leftarrow \textsc{Inv}_{\mathcal{F}}(t_k, b, y)$. If $G = 1$, the verifier computes $(\hat{b}, \hat{x}_{\hat{b}}) \leftarrow \textsc{Inv}_{\mathcal{G}}(t_k, y)$.

3. The verifier performs either of the following with equal probability.

   (a) (*preimage test*) The verifier requests a preimage. The prover returns $(b, x) \in \{0,1\} \times \mathcal{X}$. If $G = 0$ and $x \neq \hat{x}_b$, or if $G = 1$ and $(b, x) \neq (\hat{b}, \hat{x}_{\hat{b}})$, the verifier sets $flag \leftarrow fail_p$.

   (b) (*measurement test*) The verifier requests an equation $d \in \mathbb{Z}_8^w$ from the prover. If $G = 0$, the verifier computes $\hat{\theta} = \hat{\theta}(d)$ and $\hat{v} = \hat{v}(d)$.

   The verifier performs either of the following tests with equal probability:

   (i) (*Z-measurement test*) The verifier sends the label Z to the prover. The prover replies with a bit b. If $G = 1$ and $b \neq \hat{b}$, the verifier sets $flag \leftarrow fail_Z$.

   (ii) (*X_θ-measurement test*) The verifier selects $\theta \leftarrow_U \{0,1,2,3\}$ and sends θ to the prover. The prover responds with a bit v. If $G = 0$, depending on the value of θ, the verifier performs one the following tests:

   A. If $\theta = \hat{\theta}$ but $v \neq \hat{v}$, the verifier sets $flag \leftarrow fail_X$.

   B. If $\theta \in \{0,2\}$ and $\hat{\theta} \in \{1,3\}$ the verifier performs the QRAC test (Definition 3.3).

Figure 3: The qubit preparation test.

where x_0 and x_1 are the unique preimages of y under $f_{k,0}$ and $f_{k,1}$ respectively.[13] (In case $G = 1$ the state further collapses to a single $|b, x_b\rangle$.)

If the verifier requests a preimage, the prover measures in the computational basis and returns (b, x_b). If the verifier requests an equation, the prover first evaluates the map J on the second register and then measures all but the first register in the Fourier (over $\mathbb{Z}_8$) basis to obtain a string $d \in \mathbb{Z}_8^w$. The resulting state is

$$\frac{1}{\sqrt{2}}\big(e^{\frac{2i\pi}{8}d\cdot J(x_0)}\,|0\rangle\,|x_0\rangle + e^{\frac{2i\pi}{8}d\cdot J(x_1)}\,|1\rangle\,\big)\;, \tag{13}$$

where the inner products are taken modulo 8. Finally, the prover measures the qubit in (11) in the requested basis, σ_Z in the case of a Z-measurement or $\sigma_{X,\theta\frac{\pi}{4}}$ in the case of an X_θ-measurement, to produce its answer. □

3.3 Rigidity

In this section we show that any prover, or *device*, that succeeds with probability close to optimum in the qubit preparation test (Figure 3) must perform measurements that obey a form of rigidity. We generally use ε to denote the failure probability of the device in the test or one of its parts (the definition of ε will always be specified in context), and always assume that ε is larger than any term of the form $\mathrm{negl}(\lambda)$. The main result of the section is the following.

Lemma 3.5. *Let $\varepsilon > 0$ and λ a security parameter assumed to be chosen large enough so that $\varepsilon = \omega(\mathrm{negl}(\lambda))$. Suppose that a quantum polynomial-time prover succeeds in the qubit preparation test with probability at least $1-\varepsilon$. Let Z be the observable associated with the prover's strategy in step (b)(i) of the protocol, and $\{X_\theta\}_{\theta\in\{0,1,2,3\}}$ the observables associated with the prover's strategy in step (b)(ii).*

Then there exists a universal constant $c > 0$, a $\delta = O(\varepsilon^c)$, an efficiently computable isometry $\Phi : \mathcal{H}_B \to \mathbb{C}^2 \otimes \mathcal{H}_{B'}$, where $\mathcal{H}_B$ is the Hilbert space on which the prover's observables act, and a state $|\mathrm{AUX}\rangle \in \mathcal{H}_{B'} \otimes \mathcal{H}_{B''}$, where $mH_{B'}$ is a purifying system for Bob's initial state in $\mathcal{H}_B$, such that under the isometry Φ the following hold:

- *In case $G = 1$, the joint state of the bit b and the prover's post-measurement state in step (b) of the protocol, after having returned an equation d, is computationally indistinguishable from a state that is within δ trace distance of*

$$\sum_b |b\rangle\langle b| \otimes |b\rangle\langle b| \otimes |\mathrm{AUX}\rangle\langle\mathrm{AUX}|\;.$$

- *In case $G = 0$, the joint state of the angle $\hat{\theta}$, the bit $\hat{v}$ and the prover's post-measurement state in step (b) of the protocol, after having returned an equation d, is computationally indistinguishable from a state that is within δ trace distance of*

$$\sum_{\theta\in\{0,1,2,3\},v\in\{0,1\}} |\theta\rangle\langle\theta| \otimes |v\rangle\langle v| \otimes |{+}_{\theta\frac{\pi}{4}+v\pi}\rangle\langle{+}_{\theta\frac{\pi}{4}+v\pi}| \otimes |\mathrm{AUX}\rangle\langle\mathrm{AUX}|\;.$$

The proof of Lemma 3.5 is given at the end of Section 3.3.5. We start by introducing notation to model the behavior of an arbitrary prover in the test.

[13] Here for clarity we ignore the fact that $f_{k,b}$ ranges over the set of *distributions* over $\mathcal{Y}$, rather than over $\mathcal{Y}$ itself. For details on how the prover can construct the state (12) with success probability exponentially close to 1 in λ, we refer to [BCM+18a].

3.3.1 Devices

Definition 3.6. A device $D = (\phi, \Pi, M, Z, \{X_\theta\}_{\theta \in \{0,1,2,3\}})$ is specified by the following.

1. A (not necessarily normalized) positive semidefinite $\phi \in \text{Pos}(\mathcal{H}_\mathsf{D} \otimes \mathcal{H}_\mathsf{Y})$. Here $\mathcal{H}_\mathsf{D}$ is an arbitrary space private to the device, and $\mathcal{H}_\mathsf{Y}$ is a space of the same dimension as the cardinality of the set $\mathcal{Y}$, also private to the device. (We think of ϕ as the state of the device immediately prior to returning the commitment string y. In particular, ϕ implicitly depends on the key $k \in \mathcal{K}_\mathcal{F} \cup \mathcal{K}_\mathcal{G}$.) For every $y \in \mathcal{Y}$, define
$$\phi_y \;=\; (\text{Id}_\mathsf{D} \otimes \langle y|_\mathsf{Y})\, \phi\, (\text{Id}_\mathsf{D} \otimes |y\rangle_\mathsf{Y}) \;\in\; \text{Pos}(\mathcal{H}_\mathsf{D})\;.$$
Note that ϕ_y is not normalized, and $\sum_{y \in \mathcal{Y}} \text{Tr}(\phi_y) = \text{Tr}(\phi)$.

2. For every $y \in \mathcal{Y}$,

   (a) A projective measurement $\{\Pi_y^{(b,x)}\}$ on $\mathcal{H}_\mathsf{D}$, with outcomes $(b, x) \in \{0,1\} \times \mathcal{X}$. For each y, this measurement has two designated outcomes $(0, x_0)$ and $(1, x_1)$, which are the answers that are accepted in the preimage test; recall that we use the notation V_y for this set. For $b \in \{0,1\}$ we use the shorthand $\Pi_y^b = \Pi_y^{(b,x_b)}$, $\Pi_y = \Pi_y^0 + \Pi_y^1$, and $\Pi_y^2 = \text{Id} - \Pi_y^0 - \Pi_y^1$.

   (b) A projective measurement $\{M_y^d\}$ on $\mathcal{H}_\mathsf{D}$, with outcomes $d \in \mathbb{Z}_8^w$.

   (c) A binary observable Z on $\mathcal{H}_\mathsf{D}$.

   (d) For every $\theta \in \{0,1,2,3\}$, a binary observable X_θ on $\mathcal{H}_\mathsf{D}$.

By Naimark's theorem, up to increasing the dimension of $\mathcal{H}_\mathsf{D}$ the assumption that $\{\Pi_y^{(b,x)}\}$, $\{M_y^d\}$ and Z, X_θ are projective is without loss of generality. For notational convenience we often drop the subscript y from the measurements Π_y and M_y, and the state ϕ_y.

Definition 3.7 (Efficient devices)**.** We say that a device $D = (\phi, \Pi, M, Z, \{X_\theta\})$ is *efficient* if the state ϕ can be prepared efficiently, and each of the measurements can be implemented efficiently.

We introduce notation for some post-measurement states of a device.

Definition 3.8. Let $D = (\phi, \Pi, M, Z, \{X_\theta\})$ be a device. Let $\theta \in \{0,1,2,3\}$ and $v \in \{0,1\}$. Define a sub-normalized density matrix
$$\phi_{y,\theta,v} \;=\; \sum_{d:\,(\hat{\theta}(d),\hat{v}(d))=(\theta,v)} \left(\text{Id}_\mathsf{Y} \otimes M_y^d\right) \phi_y \left(\text{Id}_\mathsf{Y} \otimes M_y^d\right)\;. \tag{14}$$

We sometimes omit y and write $\phi_{\theta,v}$ for the same state. Note that since we assumed that $\{M_y^d\}$ is projective, the 8 states $\phi_{\theta,v}$ are orthogonal. We write $\phi_\theta = \phi_{\theta,0} + \phi_{\theta,1}$.

3.3.2 Preimage test

In this section we draw consequences from the assumption that a device succeeds with probability at least $1 - \varepsilon$ in the preimage test.

Lemma 3.9. *Let $D = (\phi, \Pi, M, Z, \{X_\theta\})$ be an efficient device that succeeds with probability at least $1-\varepsilon$ in the preimage test, for some $0 \le \varepsilon \le 1$. Then there is an efficient device $D' = (\phi', \Pi, M, Z, \{X_\theta\})$ such that $\|\phi' - \phi\|_1 = O(\sqrt{\varepsilon})$ and such that D' succeeds with probability negligibly (in the security parameter λ) close to 1 in the preimage test. In particular, for any $k \in \mathcal{K}_\mathcal{F}$ the state of D' after having returned y has the form*

$$|\phi'_y\rangle = \sum_{b\in\{0,1\}} |b, x_b\rangle \, |\phi_{y,b}\rangle \ , \tag{15}$$

where for $b \in \{0,1\}$, $x_b = \text{INV}_\mathcal{F}(t_k, b, y)$, $|\phi_{y,0}\rangle$ and $|\phi_{y,1}\rangle$ are arbitrary, and the basis is chosen such that the measurement $\{\Pi^{(b,x_b)}\}$ is a computational basis measurement of the first two registers. Similarly, for $k \in \mathcal{K}_\mathcal{G}$ the same state can be expressed as

$$|\phi'_y\rangle = |\hat{b}, x_{\hat{b}}\rangle \, |\phi_{y,\hat{b}}\rangle \ , \tag{16}$$

where $(\hat{b}, x_{\hat{b}}) = \text{INV}_\mathcal{G}(t_k, y)$.

Proof. The proof is analogous to the reduction to a "perfect prover" shown in [Mah18c, Claim 7.2], and we only sketch it here. Given ϕ_y, the device can evaluate $\text{CHK}_\mathcal{F}$ in superposition to check if it would succeed in the preimage test. The device D' then repeatedly prepares ϕ and measures y as D would, until it has obtained a state ϕ_y that passes the preimage test with certainty (or until a polynomial number of attempts to do so have failed). The distance between D and D' is bounded by the gentle measurement lemma (Lemma 9 in [Win99]). □

Lemma 3.10. *Let D be an efficient device that succeeds with probability 1 in the preimage test. Then for every $\theta \in \{0,1,2,3\}$ and $v \in \{0,1\}$, no polynomial-time quantum procedure can predict $\hat{\theta}(d)$ given $(y, d, \phi_{\theta,v})$ with advantage non-negligibly larger than $\frac{1}{4}$. Moreover, for every $\theta \in \{0,1,2,3\}$ no polynomial-time quantum procedure can predict $\hat{v}(d)$ given $(y, d, \phi_{\theta,v}, \theta)$ with advantage non-negligibly larger than $\frac{1}{2}$.*

In particular, the joint distribution of $(\hat{\theta}(d), \hat{v}(d))$ computed by the verifier in the measurement test is negligibly close to uniform, where the probability is taken over the device's actions, including the choice of y and d.

Proof. Suppose for contradiction that there exists a distinguisher that achieves success probability noticeably larger than $\frac{1}{8}$, where the probability is over y and d as computed by the device as well as the distinguisher's internal randomness. Suppose first that the distinguisher can predict $\hat{\theta}(d))$ with advantage noticeably larger than $\frac{1}{4}$. Using the collapsing property (Lemma A.7) and the fact that $\{M_y^d\}$ is efficient and the distinguisher are assumed efficient, it is still the case that the distinguisher has advantage noticeably larger than $\frac{1}{4}$ in predicting $\hat{\theta}(d))$ when the device first measures $\{\Pi_y^{(b,x_b)}\}$ to obtain (b, x_b) and then only applies $\{M_y^d\}$ to obtain d. This contradicts the hardcore bit property (35).

Similarly, if the distinguisher has advantage noticeably larger than $\frac{1}{2}$ in predicting $\hat{v}(d)$, conditioned on its guess for $\hat{\theta}(d)$ being correct, using the collapsing property we construct an adversary that contradicts the hardcore bit property (34). □

3.3.3 Z-measurement test

Lemma 3.11. *Let D be an efficient device that succeeds with probability 1 in the preimage test, and at least $1-\varepsilon$ in the Z-measurement test. Then on average over $y \in \mathcal{Y}$,*

$$\sum_{d,b} \text{Tr}\big((M^d\Pi^b - Z^b M^d)^\dagger (M^d\Pi^b - Z^b M^d)\phi\big) \ \le \ 2\varepsilon + \text{negl}(\lambda) \ . \tag{17}$$

Proof. The assumption of success $1-\varepsilon$ in the Z-measurement test implies that, on average over $k \in \mathcal{K}_{\mathcal{G}}$ and $y \in \mathcal{Y}$,

$$\sum_{b,d} \operatorname{Tr}\big(Z^b M^d \Pi^b \phi \Pi^b M^d\big) \;\geq\; 1-\,\varepsilon - \operatorname{negl}(\lambda)\;, \tag{18}$$

where we used that for $k \in \mathcal{K}_{\mathcal{G}}$ by Lemma 3.9 it holds that $\phi = \sum_b \Pi^b \phi \Pi^b$. Since Π, M and Z can all be efficiently implemented, using the collapsing property (Lemma A.7) (18) holds on average over $k \in \mathcal{K}_{\mathcal{F}}$ as well.

Let $\Pi = \Pi^0 - \Pi^1$ act on the first qubit of ϕ (written as in (15)). Again using the collapsing property, $\Pi\phi\Pi$ and ϕ are computationally indistinguishable, so

$$\sum_{b,d} \big|\operatorname{Tr}\big(Z^b M^d (\phi - \Pi\phi\Pi) M^d\big)\big| \;=\; \operatorname{negl}(\lambda)\;. \tag{19}$$

Using that $\phi - \Pi\phi\Pi = 2(\Pi^0\phi\Pi^1 + \Pi^1\phi\Pi^0)$, combining (18) and (19) gives

$$\sum_{b,d} \big(\operatorname{Tr}\big(Z^b M^d \Pi^b \phi M^d\big) + \operatorname{Tr}\big(Z^b M^d \phi \Pi^b M^d\big)\big) \;\geq\; 2\big(1 - O(\varepsilon)\big) - \operatorname{negl}(\lambda)\;.$$

Expanding the square in (17), this proves the lemma. □

3.3.4 X_θ-measurement test, part A

Lemma 3.12. *Let $D = (\phi, \Pi, M, Z, \{X_\theta\})$ be an efficient device. Define a sub-normalized density*

$$\begin{aligned}\tilde{\phi}_{YBXD} &= \sum_{y\in\mathcal{Y}} |y\rangle\langle y|_Y \otimes \sum_{b\in\{0,1\}} |b, x_b\rangle\langle b, x_b|_{BX} \otimes \Pi_y^{(b,x_b)}\, \phi_y\, \Pi_y^{(b,x_b)} \\ &= \sum_{b\in\{0,1\}} |b, x_b\rangle\langle b, x_b|_{BX} \otimes \tilde{\phi}^{(b)}_{YD}\,. \end{aligned} \tag{20}$$

Then $\tilde{\phi}_{YBXD}$ is the post-measurement state of the device at the end of the preimage test. For $v \in \{0,1\}$ and $\theta \in \{0,1,2,3\}$ let

$$\sigma_{\theta,v} = \sum_{y\in\mathcal{Y}} \sum_{b\in\{0,1\}} |b, x_b\rangle\langle b, x_b|_{BX} \otimes \sum_{d:\hat{\theta}(d)=\theta} |d\rangle\langle d| \otimes (\mathrm{Id}_Y \otimes X_\theta^v M_y^d)\tilde{\phi}^{(b)}_{YD}(\mathrm{Id}_Y \otimes M_y^d X_\theta^v)\;. \tag{21}$$

Then for any $\theta \in \{0,1,2,3\}$, $\sigma_{\theta,0}$ and $\sigma_{\theta,1}$ are computationally indistinguishable.

Proof. The proof is almost identical to the proof of [BCM+18a, Lemma 7.1]. Suppose for contradiction that there exists a $\theta \in \{0,1,2,3\}$ and an efficient observable O such that

$$\operatorname{Tr}(O(\sigma_{\theta,0} - \sigma_{\theta,1})) \;\geq\; \mu\;, \tag{22}$$

for some non-negligible function $\mu(\lambda)$. We derive a contradiction with the hardcore bit property (34).

Consider the following efficient procedure $\mathcal{A}$. $\mathcal{A}$ first prepares the state $\tilde{\phi}_{\mathsf{YBXD}}$ in (20). This can be done efficiently by first preparing ϕ_{YD}, then measuring a $y \in \mathcal{Y}$, then applying the measurement $\{\Pi_y^{(b,x)}\}$ to ϕ_y, and returning a special abort symbol if the outcome is invalid, i.e. $\mathrm{CHK}_{\mathcal{F}}(k, b, x, y) = 0$.

$\mathcal{A}$ then applies the measurement $\{M_y^d\}$ to $\tilde{\phi}_{\mathsf{YBXD}}$, obtaining an outcome $d \in \mathbb{Z}_8^w$. Next, it measures using $\{X_\theta^v\}$ to obtain $v \in \{0,1\}$. At this point, the procedure has prepared either $\sigma_{\theta,0}$ or $\sigma_{\theta,1}$. Finally, the procedure measures O to obtain a bit u, and returns $(b, x, d, \theta, u \oplus v)$.

This defines an efficient procedure. Using (22) it follows that the procedure violates the hardcore bit property (34). To see why, note that the guarantee (22) only holds when $\theta = \hat{\theta}(d)$, but this is precisely when (34) requires that there should be no distinguishing advantage. □

Corollary 3.13. *Let $D = (\phi, \Pi, M, Z, \{X_\theta\})$ be an efficient device that succeeds in the preimage test with probability* 1*, and in the Z-measurement test with probability at least $1-\varepsilon$. Then on average over y,*

$$\sum_{\theta \in \{0,1,2,3\}} \sum_{b \in \{0,1\}} \Big| \mathrm{Tr}\big(X_\theta^0 Z^b \phi_\theta Z^b\big) - \mathrm{Tr}\big(X_\theta^1 Z^b \phi_\theta Z^b\big) \Big| \;=\; O\big(\sqrt{\varepsilon}\big) \;,$$

where ϕ_θ is defined in Definition 3.8.

Proof. Lemma 3.12 implies that $\sigma_{\theta,0}$ and $\sigma_{\theta,1}$ must have traces that are negligibly far from each other, i.e. for every θ and on average over $y \in \mathcal{Y}$,

$$\sum_{d:\,\hat{\theta}(d)=\theta} \sum_b \Big| \mathrm{Tr}\big(X_\theta^0 M^d \Pi^b \phi \Pi^b M^d\big) - \mathrm{Tr}\big(X_\theta^1 M^d \Pi^b \phi \Pi^b M^d\big)\Big) \Big| \;=\; \mathrm{negl}(\lambda) \;.$$

Using Lemma 3.11 and the Cauchy-Schwarz inequality, this expression is within $O(\sqrt{\varepsilon})$ of

$$\sum_{d:\,\hat{\theta}(d)=\theta} \sum_b \Big| \mathrm{Tr}\big(X_\theta^0 Z^b M^d \phi M^d Z^b\big) - \mathrm{Tr}\big(X_\theta^1 Z^b M^d \phi M^d Z^d\big)\Big) \Big| \;=\; \mathrm{negl}(\lambda) \;,$$

as desired. □

The following lemma shows a strong form of incompatibility between the measurements Z and X_θ, for any efficient device.

Lemma 3.14. *Let $D = (\phi, \Pi, M, Z, \{X_\theta\})$ be an efficient device such that D succeeds with probability* 1 *in the preimage test, and with probability at least $1-\varepsilon$ in both the Z-measurement test and part A. of the X_θ-measurement test. Then there exists $\varepsilon_{\mathrm{AC}} = O(\varepsilon^{1/4})$ such that on average over $y \in \mathcal{Y}$,*

$$\sum_{\theta \in \{0,1,2,3\}} \mathrm{Tr}\big(\{Z, X_\theta\}^2 \phi_\theta\big) \;\leq\; \varepsilon_{\mathrm{AC}} \;.$$

Proof. The assumption that D succeeds with probability $1-\varepsilon$ in part A. of the X_θ-measurement test implies that on average over $y \in \mathcal{Y}$ and θ distributed according to $\mathrm{Tr}(\phi_\theta)$,

$$\sum_{v \in \{0,1\}} \mathrm{Tr}(X_\theta^v \phi_{\theta,v}) \geq 1 - \varepsilon \;. \tag{23}$$

Let $\tilde{\phi}_\theta$ be the normalized state $\phi_\theta / \mathrm{Tr}(\phi_\theta)$. Using Lemma 3.10 to argue that the renormalization is roughly uniform for all but a negligible fraction of all y, Corollary 3.13 implies that on average over y,

$$\Big| \sum_b \mathrm{Tr}(X_\theta^0 Z^b \tilde{\phi}_\theta Z^b) - \sum_b \mathrm{Tr}(X_\theta^1 Z^b \tilde{\phi}_\theta Z^b) \Big| \;=\; O\big(\sqrt{\varepsilon}\big) + \mathrm{negl}(\lambda) \;.$$

Since $\sum_{b,v} \mathrm{Tr}(X_\theta^v Z^b \tilde{\phi}_\theta Z^b) = 1$, it follows that for any $\theta \in \{0,1,2,3\}$ and $v \in \{0,1\}$,

$$\mu_{\theta,v} = \Big| \frac{1}{2} - \sum_b \mathrm{Tr}(X_\theta^v Z^b \tilde{\phi}_\theta Z^b) \Big| \;=\; O\big(\sqrt{\varepsilon}\big) \;. \tag{24}$$

Conditions (23) and (24) place us in a position to apply [BCM+18a, Lemma 7.2], with $\phi = \phi_{\theta,v}$ (renormalized), $M = X_\theta^v$, and $\Pi = Z^0$. Taking $\omega = \frac{1}{2} + \Omega(\varepsilon^{1/4})$, the lemma implies that the projection K on eigenspaces of the operator

$$\frac{1}{2}\big(Z X_\theta^v Z + X_\theta^v\big) \;=\; Z^0 X_\theta^v Z^0 + Z^1 X_\theta^v Z^1$$

with associated eigenvalue bounded away from $\frac{1}{2}$ by $\Omega(\varepsilon^{1/4})$ satisfies $\mathrm{Tr}((\mathrm{Id} - K)\phi) = O(\sqrt{\varepsilon})$. Thus for $v \in \{0,1\}$, $\mathrm{Tr}(([Z, X_\theta^v] - \frac{1}{2}Z)^2 \phi) = O(\varepsilon^{1/4})$. The lemma follows. □

Lemma 3.14 specifies that Z and X_θ are close to anti-commuting on the state ϕ_θ. The following lemma uses the collapsing property and the hardcore bit property to argue that anti-commutation extends to any ϕ_u, for $u \in \{0,1,2,3\}$.

Lemma 3.15. *Under the same assumptions as Lemma 3.14, on average over $y \in \mathcal{Y}$ and for all $\theta \in \{0,1,2,3\}$,*

$$\sum_{u\in\{0,1,2,3\}} \mathrm{Tr}\big(\{Z,X_\theta\}^2\phi_u\big) \,\leq\, \varepsilon'_{\mathrm{AC}}\;,$$

for some $\varepsilon'_{\mathrm{AC}} = O(\sqrt{\varepsilon_{\mathrm{AC}}})$.

Proof. First we observe that for a (possibly unknown) u the value $\mathrm{Tr}\big(\{Z,X_\theta\}^2\phi_u\big)$ can be estimated efficiently. This is because for any $|\psi\rangle$, it is possible to implement

$$|\psi\rangle \mapsto \frac{1}{\sqrt{2}}\big(\,|\psi\rangle\,|0\rangle + |\psi\rangle\,|1\rangle\,\big) \mapsto \frac{1}{\sqrt{2}}\big(ZX_\theta\,|\psi\rangle\,|0\rangle + X_\theta Z\,|\psi\rangle\,|1\rangle\,\big)\;, \tag{25}$$

at which point a measurment of the last qubit in the Hadamard basis returns $|+\rangle$ with probability $\frac{1}{2}\langle\psi|\,\{Z,X_\theta\}^2\,|\psi\rangle$. By the collapsing property, $\sum_u \mathrm{Tr}\big(\{Z,X_\theta\}^2\phi_u\big)$ is within negligible distance of $\sum_u \mathrm{Tr}\big(\{Z,X_\theta\}^2\tilde{\phi}_u\big)$, where $\tilde{\phi}_u$ is the result of first measuring $\{\Pi^{(b,x_b)}\}$ on ϕ and then measuring $\{M^d\}$.

Now suppose for contradiction that there exists an u' such that $\mathrm{Tr}\big(\{Z,X_\theta\}^2\phi_{u'}\big)$ is noticeably larger than $\mathrm{Tr}\big(\{Z,X_\theta\}^2\phi_u\big)$, for all $u \neq u'$. As argued above, by the collapsing property the same holds with respect to the states $\tilde{\phi}_{u'}$ and $\tilde{\phi}_u$.

Consider the following efficient procedure $\mathcal{A}$. Starting from ϕ, measure $\{\Pi^{(b,x_b)}\}$ to obtain (b,x_b). Then measure $\{M^d\}$ to obtain d. Finally, implement the test described in (25). If the outcome is $|+\rangle$, return u'. If the outcome is $|-\rangle$, repeat a uniformly random $u \in \{0,1,2,3\}$.

Then $\mathcal{A}$ returns (b,x_b,d,u') such that $u' = \hat{\theta}(d)$ with probability noticeably larger than $1/4$, violating the adaptive hardcore bit property (35). □

3.3.5 X_θ-measurement test, part B

Lemma 3.16. *Let $D = (\phi,\Pi,M,Z,\{X_\theta\})$ be an efficient device, such that D succeeds with probability* 1 *in the preimage test, with probability at least $1-\varepsilon$ in both the Z-measurement test and part A. of the X_θ-measurement test, and with probability at least $(1-\varepsilon)\mathrm{OPT}_B$ in part B. of the X_θ-measurement test. Then on average over $y \in \mathcal{Y}$,*

$$\sum_{u\in\{0,1,2,3\}} \frac{1}{2}\sum_{\theta\in\{0,1\}} \mathrm{Tr}\big(\{X_\theta,X_{\theta+2}\}^2\phi_u\big) \,\leq\, \varepsilon''_{\mathrm{AC}}\;,$$

for some $\varepsilon''_{\mathrm{AC}} = O((\varepsilon'_{\mathrm{AC}})^{1/4})$.

Proof. We perform a reduction to Lemma 3.2. The main work we need to do is argue that the observables X_0 and X_2 can be represented as observables acting on the same qubit. (The case of X_1 and X_3 is similar.)

Applying Lemma 3.15 for $\theta = 0$ and $\theta = 2$ followed by Lemma 2.9 we deduce that there is an isometry $V : \mathcal{H}_{\mathsf{D}} \to \mathbb{C}^2 \otimes \mathcal{H}_{\mathsf{D}'}$ and $\delta' = O(\sqrt{\varepsilon_{\mathrm{AC}}})$ under which which $Z \simeq_{\delta'} \sigma_Z \otimes \mathrm{Id}$, $X_0 \simeq_{\delta'} \sigma_X \otimes \mathrm{Id}$, and $X_2 \simeq_{\delta'} \sigma_X \otimes A_X + \sigma_Y \otimes A_Y$, where A_X, A_Y are Hermitian commuting such that $A_X^2 + A_Y^2 = \mathrm{Id}$.

Let $\{|v_j\rangle\}$ be a joint diagonalization basis of A_X and A_Y. Let $\rho = \sum_{u\in\{0,1,2,3\}} V\phi_u V^\dagger$, as a density matrix on $\mathbb{C}^2 \otimes \mathcal{H}_{\mathsf{D}'}$ (ρ implicitly depends on y, so it is not normalized). Define a distribution $p_{j,y} = \mathrm{Tr}(\rho^{(j)})$, with $\rho^{(j)}$ the single-qubit density matrix

$$\rho^{(j)} = (\mathrm{Id} \otimes \langle v_j|)\rho(\mathrm{Id} \otimes |v_j\rangle)\;. \tag{26}$$

For any j, y define a $2 \mapsto 1$ QRAC as follows. The encoding of $u + 4v \in \{1,3,5,7\}$, with $u \in \{1,3\}$ and $v \in \{0,1\}$, is the renormalized density matrix $\rho_{u,v}^{(j)}$, defined as $\rho^{(j)}$ in (26) with $\rho_{u,v}$ instead of ρ. The observables are σ_X and $\langle v_j| A_X |v_J\rangle \sigma_X + \langle v_j| A_Y |v_j\rangle \sigma_Y$.

Using Lemma 3.10 and the assumption on the device's success probability in part B. of the X_θ-measurement test it follows that this QRAC, on average over (y, j), satisfies the assumption of Lemma 3.2, for some $\delta = O(\sqrt{\delta'})$. The conclusion follows. □

We end with the proof of the main lemma of the section, Lemma 3.5.

Proof of Lemma 3.5. Fix a strategy for the prover that is accepted with probability at least $1 - \varepsilon$ in the qubit preparation test. Then the strategy is accepted with probability at least $1 - 2\varepsilon$ in the preimage test, and at least $1 - 4\varepsilon$ in each of the Z-measurement test and the X_θ-measurement test.

Applying Lemma 3.15 and Lemma 3.16 followed by Lemma 2.9 to Z, X_0 and X_2 if follows that there exists an efficient isometry V from $\mathcal{H}_B$ to $\mathbb{C}^2 \otimes \mathcal{H}_{B'}$ under which $Z \simeq_{\delta_1} \sigma_Z \otimes \mathrm{Id}$, $X_0 \simeq_{\delta_1} \sigma_X$ and $X_2 \simeq_{\delta_1} \sigma_Y \otimes \mathrm{Id}$, for some $\delta_1 = O(\sqrt{\varepsilon''_{ac}})$. It follows from success in part (b)(ii)A. that, under the isometry, for $\theta \in \{0,2\}$ and $v \in \{0,1\}$ the state $\phi_{\theta,v}$ is within $O(\delta_1)$ of $|+_{\theta\frac{\pi}{4}+v\pi}\rangle\langle+_{\theta\frac{\pi}{4}+v\pi}| \otimes |\textsc{aux}_{\theta,v}\rangle\langle\textsc{aux}_{\theta,v}|$, for some states $|\textsc{aux}_{\theta,v}\rangle$. Using success in part (b)(ii)B. (the QRAC test) and applying Lemma 3.2 it similarly follows that for $\theta \in \{1,3\}$ and $v \in \{0,1\}$ the state $\phi_{\theta,v}$ is, under the same isometry, within $O(\delta_1)$ of a state of the form $|+_{\theta\frac{\pi}{4}+v\pi}\rangle\langle+_{\theta\frac{\pi}{4}+v\pi}| \otimes |\textsc{aux}_{\theta,v}\rangle\langle\textsc{aux}_{\theta,v}|$.

Recall that by Lemma 3.10 the states ϕ_θ are computationally indistinguishable. According to the previous paragraph,

$$\begin{aligned}\phi_\theta \simeq_{\delta_1} \frac{1}{2}(|0\rangle\langle 0| + |1\rangle\langle 1|\,) \otimes \big(\,|\textsc{aux}_{\theta,0}\rangle\langle\textsc{aux}_{\theta,0}| + |\textsc{aux}_{\theta,1}\rangle\langle\textsc{aux}_{\theta,1}|\,\big)\\ + \big(e^{i\theta\frac{\pi}{4}}\,|0\rangle\,\langle 1| + e^{-i\theta\frac{\pi}{4}}\,|1\rangle\,\langle 0|\,\big) \otimes \big(\,|\textsc{aux}_{\theta,0}\rangle\langle\textsc{aux}_{\theta,0}| - |\textsc{aux}_{\theta,1}\rangle\langle\textsc{aux}_{\theta,1}|\,\big)\;.\end{aligned}$$

Since the operators $e^{i\theta\frac{\pi}{4}}\,|0\rangle\,\langle 1| + e^{-i\theta\frac{\pi}{4}}\,|1\rangle\,\langle 0|$ have constant trace distance for distinct values of θ, it follows that $|\textsc{aux}_{\theta,0}\rangle\langle\textsc{aux}_{\theta,0}| \approx_c |\textsc{aux}_{\theta,1}\rangle\langle\textsc{aux}_{\theta,1}|$ for all θ, and that they are computationally indistinguishable for different values of θ.

This gives the second condition in the lemma. To obtain the first, recall that under V it holds that $Z \simeq_{\delta_1} \sigma_Z \otimes \mathrm{Id}$. Using success in part (b)(i) of the protocol the state at the beginning of step (b)(i) is of the form $\sum_b |b\rangle\langle b| \otimes |b\rangle\langle b| \otimes |\textsc{aux}_b\rangle\langle\textsc{aux}_b|$, where the first b is held by the verifier and $|\textsc{aux}_b\rangle \in \mathcal{H}_{B'}$ are abitrary (not necessarily normalized). Using the collapsing property (Lemma A.7), for $b \in \{0,1\}$, $|\textsc{aux}_b\rangle\langle\textsc{aux}_b|$ is computationally indistinguishable from any of the $|\textsc{aux}_{\theta,0}\rangle\langle\textsc{aux}_{\theta,0}| + |\textsc{aux}_{\theta,1}\rangle\langle\textsc{aux}_{\theta,1}|$. □

3.4 Real protocol for remote state preparation

In this section we introduce a many-round protocol that repeatedly calls the qubit preparation test. Eventually, the protocol returns either "abort", or an angle $\theta \in \Theta$. The protocol is described in Figure 4.

To show that the protocol can be used to implement the RSP_V resource, which will be done in the next section, we prove the following.

Theorem 3.17. *Suppose the remote state preparation protocol (Figure 4) with security parameter $\lambda > 0$, maximum number of rounds $N \in \mathbb{N}$ and error tolerance $\delta \in [0,1]$ is executed with an arbitrary quantum polynomial-time prover. Assume that the protocol succeeds with probability at least ω, for some $\omega > 0$ such that $N \geq \delta^{-3}\log(2/\delta\omega)$. Then there exists an efficient isometry Φ and a state $|\textsc{aux}\rangle$ such that the*

Let λ be a security parameter, $N \geq 1$ a maximum number of rounds, δ an error tolerance parameter, and $W \in \{X, Z\}$ a basis choice.
At the start of the protocol, the verifier communicates N to the prover. The verifier privately samples a number of rounds $R \leftarrow_U \{1, \ldots, N\}$.

1. For $i = 1, \ldots, R$, the verifier executes the qubit preparation test, Figure 3. They record the outcome of the verifier in the test: either $pass$, or $fail_p$, $fail_Z$, $fail_X$ or $fail_Q$.

2. The verifier sets $flag \leftarrow abort$ if any of the following conditions is satisfied:

   (a) The fraction of preimage tests that returned $flag = fail_p$ is larger than δ;

   (b) The fraction of Z-measurement tests that returned $flag = fail_Z$ is larger than δ;

   (c) The fraction of X_θ-measurement tests, part A., that returned $flag = fail_X$ is larger than δ;

   (d) The fraction of X_θ-measurement tests, part B., that returned $flag = fail_Q$ is larger than $(1 - \textsc{opt}_Q) + \delta$.

   If $flag = abort$, the verifier aborts and sends the message ERR to the prover.

3. The verifier samples a key $(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{F}}(1^\lambda)$ (if $W = Z$) or $(k, t_k) \leftarrow \textsc{Gen}_{\mathcal{G}}(1^\lambda)$ (if $W = X$), sends k to the prover and keeps the trapdoor information t_k private.

4. The prover returns a $y \in \mathcal{Y}$ to the verifier.

5. The verifier requests an equation $d \in \mathbb{Z}_8^w$. If $W = Z$ the verifier computes $(b, x_b) \leftarrow \textsc{Inv}_{\mathcal{G}}(t_k, y)$ and returns b. If $W = X$ the verifier computes $(\hat{\theta}, \hat{v}) = (\hat{\theta}(d), \hat{v}(d))$ and returns $\theta = \hat{\theta}\frac{\pi}{4} + v\pi$.

Figure 4: The remote state preparation protocol. See Section A for notation associated with the extended NTCF family $\mathcal{F}$.

joint state of the verifier's input bit W, his output angle b ($W = Z$) or θ ($W = X$), and the prover's final state, conditioned on not aborting in step 2., is such that

$$\rho_{S\Theta B} \simeq_\varepsilon p_Z \, |Z\rangle\langle Z|_S \otimes \frac{1}{2} \sum_{b\in\{0,1\}} |b\rangle\langle b|_\Theta \otimes \Phi(|b\rangle \, |\text{AUX}\rangle \,)_B \\ + p_X \, |X\rangle\langle X|_S \otimes \frac{1}{8} \sum_{\theta\in\Theta} |\theta\rangle\langle\theta|_\Theta \otimes \Phi(|+_\theta\rangle \, |\text{AUX}\rangle)_B \, , \quad (27)$$

where $\varepsilon = O(\delta^c) + negl(\lambda)$, for some constant $c > 0$, and p_Z, p_X are the verifier's prior probability of choosing $W = Z$ or $W = X$ respectively. Moreover, the honest strategy introduced in the proof of Lemma 3.4 succeeds with probability negligibly close to 1 *in the protocol. At the end of the protocol, the verifier returns a uniformly random $b \in \{0,1\}$ and the prover's state is $|b\rangle$ ($W = Z$) or $\theta \in \Theta$ ($W = X$) and the prover's state is $|+_\theta\rangle$.*

Proof. The completeness property follows from Lemma 3.4 and a standard concentration bound.

To show soundness, fix a strategy for the prover that succeeds with probability at least ω. For $i \in \{1,\ldots,N\}$ let $T_i \in \{0,1\}$ be a random variable that equals 1 if and only if the prover does not cause the verifier to raise a $fail$ flag in the i-th round. By assumption on the prover's success probability it holds that $T = \frac{1}{R}\sum_{i=1}^R T_i \geq (1-\delta)\text{OPT}$ with probability at least ω, where $\text{OPT} = \frac{3}{4} + \frac{1}{4}\text{OPT}_Q$.

Applying Azuma's inequality, the probability that T deviates from its expectation by more than δ is at most $2e^{-\delta^2 R/2}$. It follows that as long as

$$2e^{-\delta^2 R/2}\omega \leq \delta \, , \quad (28)$$

the expectation of T conditioned on success satisfies $\mathrm{E}[T|\text{not } abort] \geq (1-\delta)\text{OPT}$. Under the assumption on N, δ and ω made in the theorem, condition (28) is satisfied with probability at least $1-\delta$ over the choice of R. Applying Markov's inequality, a randomly chosen round satisfies $\mathrm{E}[T_i|\text{not } abort] \geq 1 - O(\sqrt{\delta})$ with probability at least $1 - O(\sqrt{\delta})$. Provided such a round is chosen as the R-th round, we can apply Lemma 3.5 to conclude. □

4 Remote state preparation: ideal functionality

In this section we show that the remote state preparation protocol introduced in Section 3 constructs the ideal RSP_V resource described in Section 1. The definition of RSP_V (illustrated in Figure 1) is as follows:

Definition 4.1 (Random Remote State Preparation with Verification)**.** The resource receives $W \in \{X, Z\}$ from Alice's interface and the bit $c \in \{0,1\}$ from Bob's interface. If $c = 0$ and $W = Z$, Alice receives a uniformly random bit $b \in \{0,1\}$ and Bob receives the state $|b\rangle$. If $c = 0$ and $W = X$ Alice receives a uniformly random value $\theta \in \Theta = \{0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4}\}$ and Bob receives the state $|+_\theta\rangle$. If $c = 1$ both Alice and Bob receive an ERR message.

Recall that our goal is to show that an implementation of the RSP protocol described in the previous section, based on a classical channel and a measurement buffer as communication resources,[14] securely implements RSP_V. In the AC language, the implementation of RSP using the communication resources is

[14] The measurement buffer is used each time the measurement test, step 3(b) of the qubit preparation test in Figure 3, is executed. The classical channel is used for all other steps.

known as the real protocol, and we denote it BRSP, for buffered remote state preparation. As an abstract functionality, we illustrate it in Figure 5. Alice has an input W,[15] specifying the basis for state preparation, and produces as output either a random b if $W = Z$, a random θ if $W = X$, or ERR if Bob behaved maliciously. Bob takes as input the bit c, specifying whether he should behave honestly or maliciously. Mirroring Alice, his output is either a state $|b\rangle$, a state $|+_\theta\rangle$, or ERR. The two interact via a classical channel and a measurement buffer according to the specification given in Figure 5.

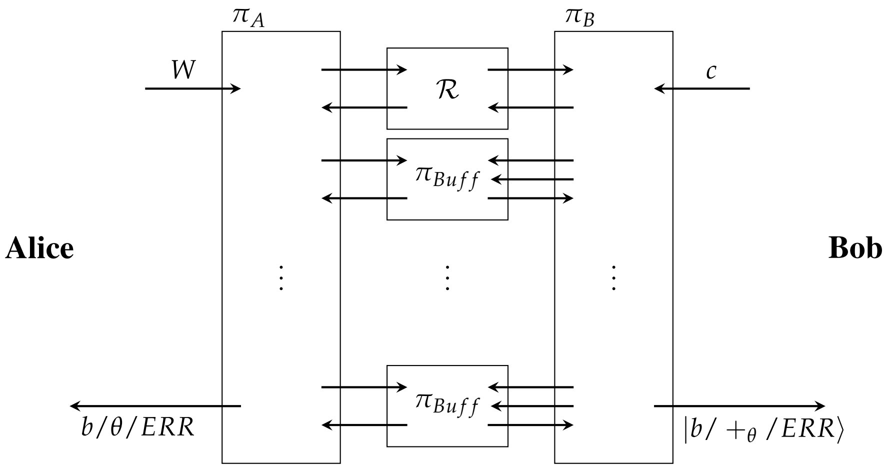

Figure 5: The remote state preparation protocol, illustrated here schematically as an AC functionality. Alice and Bob interact with the measurement buffer (which behaves as described in Figure 2), π_{Buff}, and the classical channel, $\mathcal{R}$, for a number of rounds. At the end of the interaction, upon success Alice obtains a bit b or an angle θ and Bob obtains the state $|b\rangle$ or the state $|+_\theta\rangle$. Otherwise, both parties obtain ERR. While not explicitly shown in the figure, Bob exchanges both classical and quantum messages with the buffer, whereas Alice interacts only classically with the buffer and $\mathcal{R}$.

We note that in BRSP the buffer is only needed for step 3(b) of the protocol — from Figure 4 — in which there is a constant number of challenges, so that it remains efficient for Bob to forward a specification of each of its measurements. The fact that we build the real protocol from a measurement buffer is necessary for the security proof to go through. Informally, and outside of the AC framework, the measurement buffer is "without loss of generality": in any execution of the protocol, Bob's answer to a challenge from Alice is obtained by making a measurement on a quantum state; since we assume that Bob is computationally efficient an explicit description of the measurement exists, and this is all that is needed for the stand-alone security proof.

Obtaining composable security is more subtle, and this is why the buffer is needed. Note that its use does *not* preclude Bob from sharing a prior entangled state with the environment, nor from exchanging quantum messages with the environment in-between any two uses of the measurement buffer. In this sense the use of the buffer is comparable yet much less restrictive to the way a "device" is defined to obtain composable security of device-independent protocols for e.g. randomness expansion [Por17]. In that context, Alice (the verifier) interacts with a device that is prepared by Bob (the eavesdropper). Bob is allowed to provide the initial state of the device, but he does not interact with it at any later stage of the protocol, and in particular is not allowed to receive the contents of the internal memory of the device at the end of the protocol. In our setting, this latter point is allowed.

[15] Alternatively, we say that Alice and Bob receive their inputs from (and return their outputs to) an environment.

Summarizing, we view the protocol as consisting of three parties $\pi = (\pi_A, \pi_{Buff}, \pi_B)$, where π_A denotes Alice's actions, π_B denotes Bob's actions and π_{Buff} the actions of the measurement buffer. Whenever step 3 of the protocol is executed Bob sends his state and measurements to the buffer, where it is measured according to the measurement specified by Alice's challenge. The measurement result is sent to both Alice and Bob. Bob, in addition, receives Alice's challenge and the post-measurement state. We now show that BRSP constructs the ideal RSP_V resource from classical channels.

Theorem 4.2. *The buffered remote state preparation protocol implements the ideal* RSP_V *functionality. In other words, let* $\mathcal{R}$ *denote a classical channel and* $\pi = (\pi_A, \pi_{Buff}, \pi_B)$ *denote the BRSP protocol:*

- π_A *takes as input a bit* W*, specifying either the* Z *basis or the* X *basis and produces as output either a bit* b*, an angle* θ *or the* ERR *flag. The actions that Alice performs in* π_A *are exactly the same as in the RSP protocol, as described in Figure 5.* π_A *interacts with the classical channel* $\mathcal{R}$*.*

- π_B *takes as input a bit* c*, specifying either honest (when* $c = 0$*) or malicious (when* $c = 1$*) behavior and outputs either* $|b\rangle$*,* $|+_\theta\rangle$ *or the* ERR *flag. It ignores the bit* c *and always behaves honestly, i.e. perform the actions instructed by Alice in RSP. For each measurement test performed,* π_B *sends the state to be measured to* π_{Buff} *and expects to receive the measurement outcome and the post-measurement state.*

- π_{Buff} *receives a message from Alice. It receives from Bob a specification* $\mathcal{F}$ *of a measurement to perform for each of Alice's messages, as well as a state* ρ *to be measured. It measures the state according to* $\mathcal{F}(M)$*. The measurement outcome is returned to both Alice and Bob. The post-measurement state is returned to Bob.*

We denote by $\perp_B$ *a filtered functionality for Bob that has him set* $c = 0$ *in* RSP_V*. Additionally, let* $\lambda > 0$ *denote the security parameter used in BRSP, and* $\delta > 0$ *denote the error tolerance parameter of BRSP. Then it holds that*

$$\pi_A \mathcal{R} \pi_{Buff} \pi_B \approx_{c,\varepsilon_1} \mathrm{RSP}_V \perp_B , \tag{29}$$

and there exists a polynomial-time quantum simulator σ_B *such that*

$$\pi_A \mathcal{R} \pi_{Buff} \approx_{c,\varepsilon_2} \mathrm{RSP}_V \, \sigma_B , \tag{30}$$

where $\varepsilon_1 = \mathrm{negl}(\lambda)$ *and* $\varepsilon_2 = O(\delta^c) + \mathrm{negl}(\lambda)$*, for some constant* $c > 0$*.*

Proof. Eq. (29) follows immediately from the completeness of BRSP, that is inherited from the completeness of RSP (see Theorem 3.17). Indeed, if Alice, Bob and the buffer follow the protocol, their results are exactly those obtained in the ideal functionality.

For Eq. (30), note that we are assuming that Bob is the only malicious party, whereas Alice and the buffer still follow their honest actions in the protocol. Let us consider a simulator σ_B that executes the BRSP protocol with Bob. More specifically, the simulator executes the "buffered" analogue of the protocol from Figure 4 with Bob.

If the protocol aborts before step 3, the simulator sets $c = 1$ in the ideal RSP_V functionality, indicating ERR. If the protocol does not abort, the simulator chooses uniformly at random whether to perform the $W_S = Z$ or the $W_S = X$ run of the protocol. (We denote by W_S the simulator's choice of basis to avoid confusion with W, which denotes Alice's choice.) In step 4, the simulator takes Bob's state and performs the measurement that returns the string y. Note that at step 5, from the point of view of Bob, the situation is indistinguishable from step 3(b) in the qubit preparation test (Figure 3). The simulator requests measurements for Bob associated with that step, that is, a "preimage measurement" as well as measurements Z and X_θ associated with part (b)(ii). (To ensure that Bob does not detect this modification, we may assume that in the buffered protocol execution Bob's measurements for step 3 are always requested at the end of step 2, irrespective of whether step 3 is performed or not, i.e. prior to round R or not.)

If the simulator does not receive a state and measurements of matching dimension it sets $c = 1$ and causes the ideal functionality to abort. Assuming the simulator has not aborted after receiving Bob's final state, it sets $c = 0$ in the RSP_V functionality, and takes the resulting state. Using the specification of Bob's measurement operators Z and X_θ it computes the isometry Φ whose existence is guaranteed by Theorem 3.17,[16] "undoes" the isometry by applying its inverse, replaces the first qubit of the B register in (27) by the qubit obtained from RSP_V, and re-applies the isometry. It returns the resulting state to Bob. Let us now consider a distinguisher that interacts with either $\pi_A \mathcal{R} \pi_{Buff}$ or $\mathrm{RSP}_V \sigma_B$ and which has the initial state ψ. In the first case we denote the distinguisher's final state as τ^ψ_{AB}. Assuming that in the protocol Alice accepts with probability $1 - p^\psi$, for some $p^\psi > 0$, we have that:

$$\tau^\psi_{AB} = (1 - p^\psi)\rho^\psi_{AB} + p^\psi \,|ERR\rangle \langle ERR|_A \otimes \gamma^\psi_B \tag{31}$$

where

$$\rho^\psi_{AB} \simeq_{\varepsilon_2} p^\psi_Z \,|Z\rangle\langle Z|_A \otimes \frac{1}{2} \sum_{b\in\{0,1\}} |b\rangle\langle b|_A \otimes \Phi^\psi(|b\rangle\,|\mathrm{AUX}^\psi\rangle\,)_B +$$
$$+ \, p^\psi_X \,|X\rangle\langle X|_A \otimes \frac{1}{8} \sum_{\theta\in\Theta} |\theta\rangle\langle\theta|_A \otimes \Phi^\psi(|+_\theta\rangle\,|\mathrm{AUX}^\psi\rangle))_B \tag{32}$$

and γ^ψ_B is some state on Bob's side that is consistent with the protocol having aborted and the initial state of the system being ψ. The expression from Eq. (32) follows from the rigidity theorem (Theorem 3.17), since, conditioned on success, the reduced state of Alice and Bob takes the form in (32). Note that the probabilities for the choices $W = Z$ and $W = X$, respectively, as well as the isometry on Bob's system and his auxiliary state, are determined by the initial state ψ.

Now let us consider the case when the distinguisher interacts with the ideal functionality and the simulator. We denote the final state, *prior to the simulator performing the qubit swap*, as σ^ψ_{ASB}. Once again, assuming the probability of acceptance for BRSP (as run by the simulator this time) is $1 - p^\psi$, with $p^\psi > 0$, we have that:

$$\sigma^\psi_{ASB} = (1 - p^\psi)\rho^\psi_{ASB} + p^\psi \,|ERR\rangle \langle ERR|_A \otimes |ERR\rangle \langle ERR|_S \otimes \gamma^\psi_B$$

where γ^ψ_B is the same as in (31). At this point in the protocol, $\rho^\psi_{ASB} = \rho^\psi_A \otimes \rho^\psi_{SB}$, since, conditioned on acceptance in BRSP, the simulator has not yet interacted with the ideal functionality. Using the rigidity theorem again, we get

$$\rho^\psi_{SB} \simeq_{\varepsilon_2} \frac{1}{2}\,|Z\rangle\langle Z|_S \otimes \frac{1}{2} \sum_{b\in\{0,1\}} |b\rangle\langle b|_S \otimes \Phi^\psi(|b\rangle_S\,|\mathrm{AUX}^\psi\rangle_{SB}\,)+$$
$$+ \, \frac{1}{2}\,|X\rangle\langle X|_S \otimes \frac{1}{8} \sum_{\theta\in\Theta} |\theta\rangle\langle\theta|_S \otimes \Phi^\psi(|+_\theta\rangle_S\,|\mathrm{AUX}^\psi\rangle_{SB}))\ .$$

[16] As long as the measurements are observables of the same dimension the isometry is always well-defined, whether the assumptions of the theorem are satisfied or not.

Note that the probabilities for the Z and X tests are equal, since we've established that the simulator chooses which to perform uniformly at random. Also note that the $|\textsc{aux}^{\psi}\rangle$ system is shared between the simulator and Bob. The simulator now sets $c = 0$ in the ideal functionality and receives the state it provides. As already described, it then "undoes" the isometry on the state it has from Bob, replaces the first qubit with the one from the ideal functionality, reapplies the isometry and sends the state to Bob.

As in the interaction with the real protocol, supposing that Alice (as controlled by the distinguisher) chooses to perform the two tests with probabilities p_Z^{ψ} and p_X^{ψ}, respectively, the output she receives from the ideal functionality is either b or θ, with the associated probabilities. The state that the simulator receives and swaps into Bob's system is of course classically correlated with this output, being either $|b\rangle$ or $|+_{\theta}\rangle$. If we now write the state of the system upon the completion of this last step we have

$$\tilde{\sigma}_{ASB}^{\psi} = (1 - p^{\psi})\tilde{\rho}_{ASB}^{\psi} + p^{\psi} |ERR\rangle \langle ERR|_A \otimes |ERR\rangle \langle ERR|_S \otimes \gamma_B^{\psi} ,$$

where

$$\tilde{\rho}_{ASB}^{\psi} \simeq_{\varepsilon_2} p_Z^{\psi} |Z\rangle\langle Z|_A \otimes \frac{1}{2} \sum_{b \in \{0,1\}} |b\rangle\langle b|_A \otimes \zeta_{SB}^{\psi}(b) + p_X^{\psi} |X\rangle\langle X|_A \otimes \frac{1}{8} \sum_{\theta \in \Theta} |\theta\rangle\langle\theta|_A \, \eta_{SB}^{\psi}(\theta)$$

and $\zeta_{SB}^{\psi}(b)$, $\eta_{SB}^{\psi}(\theta)$ are states on the joint system of the simulator and Bob, given by:

$$\begin{aligned}\zeta_{SB}^{\psi}(\mathbf{b}) \simeq_{\varepsilon_2} & \frac{1}{2} |Z\rangle\langle Z|_S \otimes \frac{1}{2} \sum_{b' \in \{0,1\}} |b'\rangle\langle b'|_S \otimes |b'\rangle\langle b'|_S \otimes \Phi^{\psi}(|\mathbf{b}\rangle \, |\textsc{aux}^{\psi}\rangle \,)_B + \\ & \qquad \frac{1}{2} |X\rangle\langle X|_S \otimes \frac{1}{8} \sum_{\theta' \in \Theta} |\theta'\rangle\langle\theta'|_S \otimes |+_{\theta'}\rangle\langle +_{\theta'}|_S \otimes \Phi^{\psi}(|\mathbf{b}\rangle \, |\textsc{aux}^{\psi}\rangle))_B ,\end{aligned}$$

$$\begin{aligned}\eta_{SB}^{\psi}(\boldsymbol{\theta}) \simeq_{\varepsilon_2} & \frac{1}{2} |Z\rangle\langle Z|_S \otimes \frac{1}{2} \sum_{b' \in \{0,1\}} |b'\rangle\langle b'|_S \otimes |b'\rangle\langle b'|_S \otimes \Phi^{\psi}(|\boldsymbol{+_{\theta}}\rangle \, |\textsc{aux}^{\psi}\rangle \,)_B + \\ & \qquad \frac{1}{2} |X\rangle\langle X|_S \otimes \frac{1}{8} \sum_{\theta' \in \Theta} |\theta'\rangle\langle\theta'|_S \otimes |+_{\theta'}\rangle\langle +_{\theta'}|_S \otimes \Phi^{\psi}(|\boldsymbol{+_{\theta}}\rangle \, |\textsc{aux}^{\psi}\rangle))_B .\end{aligned}$$

The boldface letters highlight the state that was planted in Bob's system. Now notice that if we trace out the simulator's system from both of these states we get:

$$\begin{aligned} Tr_S(\zeta_{SB}^{\psi}) &= \zeta_B^{\psi}(\mathbf{b}) \simeq_{\varepsilon_2} \Phi^{\psi}(|\mathbf{b}\rangle \, |\textsc{aux}^{\psi}\rangle)))_B , \\ Tr_S(\eta_{SB}^{\psi}) &= \eta_B^{\psi}(\boldsymbol{\theta}) \simeq_{\varepsilon_2} \Phi^{\psi}(|\boldsymbol{+_{\theta}}\rangle \, |\textsc{aux}^{\psi}\rangle)))_B .\end{aligned}$$

Tracing out the simulator from $\tilde{\rho}_{ASB}$ and plugging in the above expressions we obtain

$$\begin{aligned}\tilde{\rho}_{AB}^{\psi} \simeq_{\varepsilon_2} & p_Z^{\psi} |Z\rangle\langle Z|_A \otimes \frac{1}{2} \sum_{b \in \{0,1\}} |b\rangle\langle b|_A \otimes \Phi^{\psi}(|b\rangle \, |\textsc{aux}^{\psi}\rangle)))_B + \\ & \qquad + p_X^{\psi} |X\rangle\langle X|_A \otimes \frac{1}{8} \sum_{\theta \in \Theta} |\theta\rangle\langle\theta|_A \, \Phi^{\psi}(|+_{\theta}\rangle \, |\textsc{aux}^{\psi}\rangle)))_B .\end{aligned}$$

Finally, if we trace out the simulator from $\tilde{\sigma}_{ASB}^{\psi}$ and use the above state, we have:

$$\tilde{\sigma}_{AB}^{\psi} = (1 - p^{\psi})\tilde{\rho}_{AB}^{\psi} + p^{\psi} |ERR\rangle \langle ERR|_A \otimes \gamma_B^{\psi} .$$

From the triangle inequality $\rho^{\psi}_{AB} \simeq_{2\varepsilon_2} \tilde{\rho}^{\psi}_{AB}$, and therefore we have that:

$$\pi_A \mathcal{R} \pi_{Buff} \approx_{c,2\varepsilon_2} \mathrm{RSP}_V \sigma_B \ ,$$

concluding the proof. □

5 Blind and verifiable computation from remote state preparation

In [DK16] the authors show that a measurement-based protocol for blind delegation of quantum circuits, the Universal Blind Quantum Computing protocol of Broadbent et al. [BFK09] (BFK) can be constructed from the ideal functionality RSP_B and classical communication channels. Their result builds upon the work of Dunjko et al. [DFPR14], who showed composable security of the BFK protocol in the AC framework. Dunjko et al. also showed composable security of a blind and verifiable variant of the BFK protocol introduced by Fitzsimons and Kashefi [FK17] (FK). Both protocols are designed to delegate a computation that is expressed in the model of measurement-based quantum computing (MBQC). In this model, a quantum computation is implemented by preparing a graph state (a collection of qubits that are entangled according to the structure of a graph) and then performing adaptive measurements on the qubits in the graph state. The main difference between the FK protocol and the BFK protocol is the use of traps to ensure verifiability in FK. Informally, trap qubits are qubits initialized in a $|+_\theta\rangle$ state, with θ chosen uniformly at random from Θ, and such that all neighbors of the trap qubit in the underlying graph state are initialized in a random computational basis state; these are called dummy qubits. The role of the dummy qubits is to isolate the trap qubits from the computation, so that the prover's measurements on the trap qubits can be verified independently of the computation. This isolation happens because, in the specific implementation of MBQC used by the FK and BFK protocols, the graph state is prepared by entangling $|+_\theta\rangle$ states using the Controlled-Z operation. But note that this operation does not create entanglement if either of its input qubits is a state in the computational basis. We sketch the structure of the FK protocol in Figure 6, at a level that is sufficient to follow the arguments in this section; we refer to the description of Protocol 7 and Protocol 8 in [FK17] for full details.

Our goal in this section is to show that by replacing the quantum communication channel used by Alice to send single qubits to Bob in the FK protocol with BRSP we obtain a protocol that implements the ideal $\mathcal{S}^{blind}_{verif}$ resource for blind and verifiable delegated computation. Importantly, the resulting protocol involves only classical communication and is composable. We proceed in a number of incremental steps.

Note first that the set of single-qubit states prepared by the verifier in the FK protocol, $|+_\theta\rangle$ for $\theta \in \Theta$ and $|0\rangle\,,|1\rangle$, is precisely the set of states that can be generated using the BRSP resource. We therefore define two variants of the FK protocol which we call RSP_V-FK and RSP-FK. The former is identical to the FK protocol, except Alice uses the ideal resource RSP_V in order to prepare the states she is supposed to send to Bob in FK. RSP-FK is the same, except Alice uses the BRSP protocol to perform this preparation.

From the description of the FK protocol given in Figure 6, it is clear that the number of times Alice uses the ideal RSP_V functionality or the BRSP protocol respectively is equal to the number of qubits she sends to the prover. As was shown in [KW17], there exist graph states such that this number is linear in the size of the quantum circuit she wishes to delegate. Given this, we can show the following:

Lemma 5.1. *Let $\delta_{BRSP} > 0$ be the error tolerance parameter of the BRSP protocol used by Alice in RSP-FK, $\delta_{FK} > 0$ be the error (soundness) of FK, $T > 0$ the size of the computation Alice wishes to delegate to Bob*[17]

[17] Alternatively, we can say that T is the size of Alice's input and assume that she always delegates a universal circuit.

The inputs to the protocol are an error parameter $\varepsilon > 0$, a unitary quantum circuit C, and a classical input ψ_A.

1. Alice selects a set of measurement angles $\{\phi_i\}_{i\in\{1,\ldots,N\}}$, such that each $\phi_i \in \Theta$, that implement the computation specified by C. (For clarity we omit the choice of graph and flow.) Alice also selects a set of dummy qubit locations $D \subseteq \{1,\ldots,N\}$ and trap locations $T \subseteq \{1,\ldots,N\}$. Alice determines an update function $C(i,\phi_i,\theta_i,r_i,s)$.

2. Alice selects angles $\{\theta_i\}_{i\in\{1,\ldots,N\}}$ uniformly at random from Θ, $\{r_i\}_{i\in\{1,\ldots,N\}}$ uniformly at random from $\{0,1\}$ and $\{d_i\}_{i\in D}$ uniformly at random from $\{0,1\}$. She initializes values $\{s_i\}_{i\in\{1,\ldots,N\}}$ to 0.

3. Alice prepares qubits in the state $|d_i\rangle$ for $i \in D$, and $Z^{d'_i}\,|+_{\theta_i}\rangle$ for $i \notin D$, where d'_i is a predetermined function of $\{d_j\}_{j\in\{1,\ldots,N\}}$, and sends the qubits one by one to Bob.

4. For i from 1 to N:

    (a) Alice computes an angle $\delta_i = C(i,\phi_i,\theta_i,r_i,s)$ and sends it to Bob.

    (b) Bob returns $b_i \in \{0,1\}$.

    (c) Alice sets $s_i \leftarrow b_i + r_i$.

5. Alice accepts if $s_i = r_i$ for all $i \in T$. She returns the state contained in the output qubits of the computation.

Figure 6: Summary of the FK protocol. For an explanation of the notation and more details, see Protocol 8 in [FK17].

and $\lambda > 0$ the security parameter used in BRSP. Then, RSP-FK constructs RSP_V*-FK, computationally, within distance $2\delta_{FK} + O(T\varepsilon)$, where $\varepsilon = O(\delta_{BRSP}^c) + \mathrm{negl}(\lambda)$, for some constant $c > 0$.*

Proof. From Theorem 4.2 we know that if δ_{BRSP} is the error of BRSP, then BRSP implements RSP_V to within computational distance ε. We also know that RSP_V-FK involves $O(T\ \log(1/\delta_{FK}))$ uses of the RSP_V functionality, since to achieve error δ_{FK} the FK protocol uses $O(T\ \log(1/\delta_{FK}))$ qubits [KW17]). In RSP-FK these are replaced with calls to BRSP. The compositionality theorem of AC (see [MR11, DFPR14]) implies that each replacement comes at an additive cost of ε. In other words, up to an error $O(T\ \log(1/\delta_{FK})\varepsilon)$, RSP-FK behaves exactly the same as RSP_V-FK. The fact that RSP_V-FK has error δ_{FK} means that (conditioned on acceptance) it arrives at the correct result, except with error δ_{FK}. The same will be true of RSP-FK, with the added error of $O(T\ \log(1/\delta_{FK})\varepsilon)$ stemming from the use of BRSP. A triangle inequality leads us to conclude that RSP-FK implements RSP_V-FK within distance $2\delta_{FK} + O(T\ \log(1/\delta_{FK})\varepsilon)$. □

As a point of clarification, δ_{FK} represents the maximum deviations from the correct outcomes of the respective protocols, conditioned on Alice accepting. Also note that Alice can make the $O(T\ \log(1/\delta_{FK})\varepsilon)$ term be of order δ_{FK} by taking $\delta_{BRSP} = (\delta_{FK}/(T\ \log(1/\delta_{FK})))^{1/c}$.
We now show the following:

Lemma 5.2. *Let $\delta_{FK} > 0$ be the error of FK.* RSP_V*-FK implements FK within distance $2\delta_{FK}$.*

Proof. We denote the two protocols as $\pi^{\mathrm{RSP}_V-FK} = (\pi_A^{\mathrm{RSP}_V-FK}, \pi_B^{\mathrm{RSP}_V-FK})$ and $\pi^{FK} = (\pi_A^{FK}, \pi_B^{FK})$ respectively. Additionally, let $\mathcal{R}^{cq}$ denote a resource consisting of classical and quantum channels. We will show that

$$\pi_A^{\mathrm{RSP}_V-FK}\mathcal{R}^{cq}\pi_B^{\mathrm{RSP}_V-FK} = \pi_A^{FK}\mathcal{R}^{cq}\pi_B^{FK}\ , \qquad \pi_A^{\mathrm{RSP}_V-FK}\mathcal{R}^{cq} \approx_{2\delta} \pi_A^{FK}\mathcal{R}^{cq}\sigma_B\ , \tag{33}$$

for some simulator σ_B.

Correctness is immediate: if Alice and Bob behave honestly in both RSP_V-FK and in FK the results are statistically indistinguishable.

For security note the following. In RSP_V-FK, for each use of RSP_V Alice chooses the preparation bases for the $|+_\theta\rangle$ states at random, and for the dummies she consistently chooses the Z basis. The only difference between this and the actual FK protocol is that because she is using the RSP_V functionality, Bob can force Alice to abort in the preparation stage by triggering the ERR flag. To show that RSP_V-FK implements FK, we need to show that the simulator, σ_B, interacting with $\pi_A^{FK}\mathcal{R}^{cq}$, can make its interaction with Bob indistinguishable from that of $\pi_A^{\mathrm{RSP}_V-FK}\mathcal{R}^{cq}$. First of all, to match RSP_V-FK in terms of inputs and outputs it must be that for each qubit to be prepared the simulator receives the c bit from Bob indicating whether he wants to cause the current preparation to abort, as per the specification of RSP_V (see Figure 1).

Consider a simulator that works as follows. The simulator first collects all the qubits sent by Alice through $\mathcal{R}^{cq}$. Then, for each qubit that it is supposed to send to Bob, it first receives the bit c corresponding to that qubit. If $c = 0$, indicating to not abort, the simulator sends that qubit to Bob. Otherwise, it sends the ERR flag to Bob and also causes Alice to abort[18]. If the simulator sends all of the qubits to Bob (i.e. there was no abort in the preparation stage) it then acts as a classical channel between Alice and Bob, forwarding the messages Alice sends (step 4.(a)) to Bob and then forwarding his responses to Alice (step 4.(b)).

[18] The simulator can cause Alice to abort by providing random responses to the measurement outcomes she expects from Bob. The probability that these responses will match all of Alice's expected outcomes on the trap states is exponentially small. In other words, Alice will abort with probability $1 - \exp(-O(T))$. This will mean that RSP_V-FK implements FK within distance $2\delta_{FK} + \exp(-O(T))$ but since we will always consider $\delta_{FK} = \Omega(\exp(-T))$, we omit this inverse exponential term.

From the soundness of FK it follows that at the end of either $\pi_A^{\mathrm{RSP}_V - FK}\mathcal{R}^{cq}$ or $\pi_A^{FK}\mathcal{R}^{cq}\sigma_B$ the state of the system (conditioned on acceptance) is δ_{FK}-close to the correct output. Applying the triangle inequality, the output states in the two situations are $2\delta_{FK}$-close to each other. This concludes the proof. □

Note that the result of Lemma 5.2 holds within statistical distance. Of course, the result is also true when restricting to the computationally efficient setting, since the simulator is efficient (it simply needs to store states and forward messages received from Alice and Bob).

Finally, we use the following result from [DFPR13]:

Lemma 5.3 (Lemma C.1 in [DFPR13])**.** *If the FK protocol is run with parameters such that it has error δ_{FK}, then it is $4\sqrt{2}\delta_{FK}^{1/4}N^2$-blind-verifiable, where N is the dimension of the subsystem of Alice's input which is quantum.*

For convenience we restrict our attention to classical inputs for Alice. In this case it follows that FK (with classical input) with soundness parameter δ_{FK} implements the ideal $\mathcal{S}_{verif}^{blind}$ resource within distance $O(\delta_{FK}^{1/4})$. With this fact, we can show the main result of this section.

Theorem 5.4. *The RSP-FK protocol with error $\delta_{FK} > 0$ and security parameter $\lambda > 0$ implements the ideal $\mathcal{S}_{verif}^{blind}$ resource within distance $O(\delta_{FK}^{1/4}) + \mathrm{negl}(\lambda)$.*

Proof. From Lemmas 5.1 and 5.2 we see that RSP-FK with error δ_{FK} and security parameter $\lambda > 0$ implements FK with error $O(\delta_{FK}) + \mathrm{negl}(\lambda)$. Combining this with Lemma 5.3 leads us to conclude that RSP-FK implements the $\mathcal{S}_{verif}^{blind}$ resource within distance $O(\delta_{FK}^{1/4}) + \mathrm{negl}(\lambda)$. □

The result of Theorem 5.4 states that using BRSP together with the FK protocol yields a protocol that is computationally indistinguishable from the ideal blind-verifiability functionality. If we take the distinguishing advantage to be $\delta > 0$, what will be the total complexity (in terms of total of number of operations performed by the verifier) of RSP-FK for a computation of size T? From Theorem 5.4 it follows that in order to implement $\mathcal{S}_{verif}^{blind}$ to within distance δ we need to perform RSP-FK with soundness error δ^4. Implementing the FK protocol so that it achieves soundness error δ^4 requires $O(T\ \log(1/\delta^4))$ operations, where T is the size of the computation. The change from δ to δ^4 only increases the overhead by a constant factor, so that overall the prover requires $O(T\ \log(1/\delta))$ operations to implement FK. In our case, however, for each state sent by the verifier to the prover, the verifier executes BRSP. Thus the overhead is $O(C_{BRSP}\,T\ \log(1/\delta))$, where C_{BRSP} is the cost of running one instance of BRSP. If we wish to achieve soundness error δ^4 in RSP-FK, BRSP needs to have error at most $(\delta^4/(T\ \log(1/\delta)))^{1/c}$. The specific constant c can be determined from the proof of Theorem 3.17 to be $c = 1/3$. Thus we can estimate the cost of BRSP as $C_{BRSP} = (T^3/\delta^{12})\log^3(1/\delta)\,\mathrm{poly}(\lambda)$, where λ is the security parameter. This gives a total cost of $O((T^4/\delta^{12})\ \log^4(1/\delta)\ \mathrm{poly}(\lambda))$. Note that this is the cost of implementing the ideal blind-verifiable resource. If we merely wish to implement FK itself, the cost would be $O((T^4/\delta^3)\ \log^4(1/\delta)\ \mathrm{poly}(\lambda))$, since we would not incur the $1/\delta \to 1/\delta^4$ increase stemming from Lemma 5.3.

A Claw-free functions with adaptive hardcore

Our construction relies on a variant of a cryptographic primitive called a "noisy trapdoor claw-free family (NTCF)," introduced in [BCM$^+$18b], and its extension to an "extended noisy trapdoor claw-free family (ENTCF)," given in [Mah18b]. We rely on definitions and notation from [BCM$^+$18a, Section 3] and [Mah18c, Section 4].

A key property of an NTCF is the adaptive hardcore bit property, property 4. in [BCM$^+$18a, Definition 3.1]. We need a slightly stronger variant of the property, that works over $\mathbb{Z}_8$ instead of $\mathbb{Z}_2$. The property we need is formulated in the following definition.

Definition A.1. Let λ be a security parameter. Let $\mathcal{X}$ and $\mathcal{Y}$ be finite sets. Let $\mathcal{K}_{\mathcal{F}}$ be a finite set of keys. A NTCF family

$$\mathcal{F} = \{f_{k,b} : \mathcal{X} \to \mathcal{D}_{\mathcal{Y}}\}_{k \in \mathcal{K}_{\mathcal{F}}, b \in \{0,1\}}$$

is said to have *adaptive* $\mathbb{Z}_8$ *hardcore* if it satisfies the following conditions, for some integer w that is a polynomially bounded function of λ.

1. For all $b \in \{0,1\}$ and $x \in \mathcal{X}$, there exists a set $G_{k,b,x} \subseteq \mathbb{Z}_8$ such that $\Pr_{d \leftarrow_U \mathbb{Z}_8^w}[d \notin G_{k,b,x}]$ is negligible, and moreover there exists an efficient algorithm that checks for membership in $G_{k,b,x}$ given k, b, x and the trapdoor t_k.

2. There is an efficiently computable injection $J : \mathcal{X} \to \mathbb{Z}_8^w$, such that J can be inverted efficiently on its range, and such that the following holds. For any $y \in \mathcal{Y}$, define functions $\hat{\theta} : \mathbb{Z}_8^w \to \{0,1,2,3\}$ and $\hat{v} : \mathbb{Z}_8^w \to \{0,1\}$ as the unique values such that $d \cdot (J(x_0) + J(x_1)) \bmod 8 = \hat{\theta}(d) + 4\hat{v}(d)$, where for $b \in \{0,1\}$, $x_b = \textsc{Inv}_{\mathcal{F}}(t_k, b, y)$, if $d \in G_{k,0,x_0} \cap G_{k,1,x_1}$,[19] and $\hat{\theta}(d) = \hat{v}(d) = \perp$ otherwise. Then if

$$\begin{aligned} H_k &= \{(b, x_b, d, \theta, v) \mid b \in \{0,1\},\ (x_0, x_1) \in \mathcal{R}_k,\ (\theta, v) = (\hat{\theta}(d), \hat{v}(d))\}\ ,^{20} \\ \overline{H}_k &= \{(b, x_b, d, \theta, v) \mid (b, x, d, \theta, v \oplus 1) \in H_k\}\ , \end{aligned}$$

then for any quantum polynomial-time procedure $\mathcal{A}$ there exists a negligible function $\mu(\cdot)$ such that

$$\Big| \Pr_{(k,t_k) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(k) \in H_k] - \Pr_{(k,t_k) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}(k) \in \overline{H}_k] \Big| \leq \mu(\lambda)\ . \tag{34}$$

Similarly, if for $w \in \{0,1,2,3\}$,

$$H_k^{(w)} = \{(b, x_b, d, \theta) \mid b \in \{0,1\},\ (x_0, x_1) \in \mathcal{R}_k,\ \theta = \hat{\theta}(d) + w\}\ ,$$

then for any quantum polynomial-time procedure $\mathcal{A}'$ there exists a negligible function $\mu(\cdot)$ such that for all $w \in \{1,2,3\}$,

$$\Big| \Pr_{(k,t_k) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}'(k) \in H_k^{(0)}] - \Pr_{(k,t_k) \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)}[\mathcal{A}'(k) \in H_k^{(w)}] \Big| \leq \mu(\lambda)\ . \tag{35}$$

A.1 The adaptive hardcore property

It is straightforward to verify that the same construction of an NTCF introduced in [BCM$^+$18a] has adaptive $\mathbb{Z}_8$ hardcore.

Lemma A.2. *The NTCF family introduced in [BCM$^+$18a] has adaptive $\mathbb{Z}_8$ hardcore, i.e. it satisfies item 2. in Definition A.1.*

[19] The sets $G_{k,b,x}$ are defined in (40).

[20] Note that although both x_0 and x_1 are referred to to define the set H_k, only one of them, x_b, is explicitly specified in any 4-tuple that lies in H_k.

The proof of Lemma A.2 is very similar to the adaptive hardcore bit condition shown in [BCM$^+$18a, Lemma 4.7], with some modifications to obtain a hardness statement $\bmod 8$ instead of $\bmod 2$. We indicate the main changes needed, referring directly to statements from [BCM$^+$18a, Section 4.4].

The main step of the proof consists in showing the following Lemma, a direct analogue of [BCM$^+$18a, Lemma 4.2]. The main difference is the requirement on $\hat{d}$. For a string $x \in \{0,1\}^n$ we write $|x|_H$ for the Hamming weight of x.

Lemma A.3. *Let q be a prime, $\ell, n \geq 1$ integers, and $\mathbf{C} \in \mathbb{Z}_q^{\ell \times n}$ a uniformly random matrix. With probability at least $1 - q^\ell \cdot 2^{-\frac{n}{8}}$ over the choice of $\mathbf{C}$ the following holds. For a fixed $\mathbf{C}$, all $\mathbf{v} \in \mathbb{Z}_q^\ell$ and $\hat{d} \in \{0,1\}^n$ such that $|\hat{d}|_H \geq \frac{n}{8}$, the distribution of $(\hat{d} \cdot s \bmod 8)$, where s is uniform in $\{0,1\}^n$ conditioned on $\mathbf{C}s = \mathbf{v}$, is within statistical distance $O(q^{\frac{3\ell}{2}} \cdot 2^{-\frac{n}{80}})$ of the uniform distribution over $\{0,1\}$.*

The first change in the proof of [BCM$^+$18a, Lemma 4.2] required to obtain Lemma A.3 is to the definition of a moderate matrix:

Definition A.4. Let $\mathbf{b} \in \mathbb{Z}_q^n$. We say that $\mathbf{b}$ is moderate if it contains at least $\frac{n}{4}$ entries whose unique representative in $(-q/2, q/2]$ has its absolute value in the range $(\frac{q}{32}, \frac{3q}{32}]$. A matrix $\mathbf{C} \in \mathbb{Z}_q^{\ell \times n}$ is moderate if its entire row span (except 0^n) is moderate.

Lemma A.5. *Let q be prime and ℓ, n be integers. Then*

$$\Pr_{\mathbf{C} \leftarrow_U \mathbb{Z}_q^{\ell \times n}} \big(\mathbf{C} \textit{ is moderate}\big) \geq 1 - q^\ell \cdot 2^{-\frac{n}{32}} .$$

Proof. The proof is identical to [BCM$^+$18a, Lemma 4.5], except for replacing $q/8$ with $q/32$. □

Lemma A.6. *Let $\mathbf{C} \in \mathbb{Z}_q^{\ell \times n}$ be an arbitrary moderate matrix and let $\hat{d} \in \{0,1\}^n$ be such that $|\hat{d}| \geq \frac{n}{8}$. Let s be uniform over $\{0,1\}^n$ and consider the random variables $\mathbf{v} = \mathbf{C}s \bmod q$ and $z = \hat{d} \cdot s \bmod 2$. Then $(\mathbf{v}, z)$ is within total variation distance at most $2q^{\frac{\ell}{2}} \cdot 2^{-\frac{n}{80}}$ of the uniform distribution over $\mathbb{Z}_q^\ell \times \{0,1\}$.*

The proof of the lemma is similar to the proof of [BCM$^+$18a, Lemma 4.5], with a small difference due to the $\bmod 8$ condition. This is where the additional requirement that $|\hat{d}| \geq \frac{n}{10}$ (as opposed to simply $\hat{d} \neq 0$ in [BCM$^+$18a]) is used.

Proof. Let f be the probability density function of $(\mathbf{v}, z)$. Interpreting z as an element of $\mathbb{Z}_8$, let $\hat{f}$ be the Fourier transform over $\mathbb{Z}_q^\ell \times \mathbb{Z}_8$. Let U denote the density of the uniform distribution over $\mathbb{Z}_q^\ell \times \mathbb{Z}_8$. Applying the Cauchy-Schwarz inequality,

$$\begin{aligned} \frac{1}{2}\|f - U\|_1 &\leq 2\sqrt{q^\ell}\|f - U\|_2 \\ &= \frac{1}{2}\|\hat{f} - \hat{U}\|_2 \\ &= \frac{1}{2}\Big(\sum_{(\hat{\mathbf{v}},\hat{z}) \in \mathbb{Z}_q^\ell \times \mathbb{Z}_8 \setminus \{(\mathbf{0},0)\}} |\hat{f}(\hat{\mathbf{v}},\hat{z})|^2\Big)^{1/2} , \end{aligned} \tag{36}$$

where the second line follows from Parseval's identity, and for the third line we used $\hat{f}(\mathbf{0},0) = \hat{U}(0,0) = 1$ and $\hat{U}(\hat{\mathbf{v}},\hat{z}) = 0$ for all $(\hat{\mathbf{v}},\hat{z}) \neq (0^\ell, 0)$. To bound (36) we estimate the Fourier coefficients of f. Denoting

$\omega_{8q} = e^{-\frac{2\pi i}{8q}}$, for any $(\hat{\mathbf{v}}, \hat{z}) \in \mathbb{Z}_q^\ell \times \mathbb{Z}_2$ we can write

$$\begin{aligned} \hat{f}(\hat{\mathbf{v}}, \hat{z}) &= \mathrm{E}_{\mathbf{s}}\left[\omega_{8q}^{(2\cdot\hat{\mathbf{v}}^T C + q\cdot\hat{z}\hat{\mathbf{d}}^T)\mathbf{s}}\right] \\ &= \mathrm{E}_{\mathbf{s}}\left[\omega_{8q}^{\mathbf{w}^T\mathbf{s}}\right] \\ &= \prod_i \mathrm{E}_{s_i}\left[\omega_{8q}^{w_i s_i}\right] \,, \end{aligned} \tag{37}$$

where we wrote $\mathbf{w}^T = 8 \cdot \hat{\mathbf{v}}^T\mathbf{C} + q \cdot \hat{z}\hat{\mathbf{d}}^T \in \mathbb{Z}_{8q}^n$.

We first bound $\hat{f}(0^\ell, \hat{z})$ for $\hat{z} \in \mathbb{Z}_8\backslash\{0\}$. In this case (37) simplifies to

$$\begin{aligned} \left|\hat{f}(\hat{\mathbf{v}}, \hat{z})\right| &= \prod_{i:\hat{d}_i=1} \left|\mathrm{E}_{s_i}\left[e^{-\frac{2i\pi\hat{z}}{8}s_i}\right]\right| \\ &= \prod_{i:\hat{d}_i=1} \left|\cos\left(\frac{\pi}{2}\frac{\hat{z}}{4}\right)\right| \\ &\leq \prod_{i:\hat{d}_i=1} \cos\left(\frac{\pi}{8}\right) \,\leq\, 2^{-\frac{n}{80}} \,. \end{aligned} \tag{38}$$

Next we observe that for all $i \in \{1, \ldots, n\}$ such that the representative of $(\hat{\mathbf{v}}^T\mathbf{C})_i$ in $(-q/2, q/2]$ has its absolute value in $(\frac{q}{32}, \frac{3q}{32}]$ it holds that $\frac{w_i}{q} \in (\frac{1}{4}, \frac{3}{4}]$ mod 1, in which case

$$\left|\mathrm{E}_{s_i}[\omega_{8q}^{w_i s_i}]\right| \,=\, \left|\cos\left(\frac{\pi}{2}\cdot\frac{w_i}{q}\right)\right| \,\leq\, \cos\left(\frac{\pi}{8}\right) \,\leq\, 2^{-\frac{1}{10}} \,. \tag{39}$$

Since $\mathbf{C}$ is moderate, there are at least $\frac{n}{4}$ such entries, so that from (37) it follows that $|\hat{f}(\hat{\mathbf{v}}, \hat{z})| \leq 2^{-\frac{n}{40}}$ for all $\hat{\mathbf{v}} \neq \mathbf{0}$. Recalling (36) and (38), the lemma is proved. □

The proof of Lemma A.3 follows from Lemma A.6 exactly as in [BCM$^+$18a], and we omit the details. With Lemma A.3 in hand, the proof of the adaptive $\mathbb{Z}_8$ condition, item 2. in Definition A.1, is very similar to the proof of the adaptive hardcore bit condition in [BCM$^+$18a]. The main change needed is in the definition of the sets $G_{k,b,x}$, for $k = (\mathbf{A}, \mathbf{As} + \mathbf{e}), b \in \{0,1\}$ and $x \in \mathcal{X}$, that is defined as follows:

$$G_{k,b,x} \,=\, \left\{d \in \{0,1\}^w \,\middle|\, \left|I_{b,x}(d)_{\{b\frac{n}{2},\ldots,b\frac{n}{2}+\frac{n}{2}\}}\right|_H \geq \frac{n}{4}\right\} \,, \tag{40}$$

where $I_{b,x}(d)$ is the vector whose each coordinate is obtained by taking the inner product mod 2 of the corresponding block of $\lceil \log q \rceil$ coordinates of d and of $J(x) \oplus J(x - (-1)^b\mathbf{1})$, where $J : \mathcal{X} \to \{0,1\}^w$ is such that $J(x)$ returns the binary representation of $x \in \mathcal{X}$ and $\mathbf{1} \in \mathbb{Z}_q^n$ is the vector with all its coordinates equal to $1 \in \mathbb{Z}_q$.

A.2 The collapsing property

The following lemma shows that any ENTCF has the *collapsing* property, introduced by Unruh [Unr16].

Lemma A.7. *Let $(\mathcal{F}, \mathcal{G})$ be an ENTCF family. Let $\phi = \sum_{y\in\mathcal{Y}} |y\rangle\langle y| \otimes \phi_y$ be a state that can be prepared efficiently, given as input a key $k \in \mathcal{K}_\mathcal{F} \cup \mathcal{K}_\mathcal{G}$. Let $\Pi = \{\Pi^{(b,x_b)}\}$ be an efficiently implementable POVM such that* $\mathrm{Tr}(\Pi^{(b,x_b)}\phi_y) = 0$ *if $f_{k,b}(x_b) \neq y$ (if $k \in \mathcal{K}_\mathcal{F}$) or $g_{k,b}(x_b) \neq y$ (if $k \in \mathcal{K}_\mathcal{G}$). Then there is no* *efficient procedure such that, given $k \leftarrow GEN_{\mathcal{F}}(1^\lambda)$ and y distributed according to* $\mathrm{Tr}(\phi_y)$*, the procedure has a non-negligible advantage in distinguishing ϕ_y from $\phi'_y = \Pi^{(0,x_0)}\phi_y\Pi^{(0,x_0)} + \Pi^{(1,x_1)}\phi_y\Pi^{(1,x_1)}$, where for $b \in \{0,1\}$, x_b is such that $f_{k,b}(x_b) = y$.*

Proof. Suppose for contradiction that there exists such a procedure. Since the procedure is efficient, using the property of injective invariance of an ENTCF (Definition 4.2 in [Mah18c]) it should produce computationally indistinguishable outcomes given $k \leftarrow \mathrm{GEN}_{\mathcal{F}}(1^\lambda)$ or $k \leftarrow \mathrm{GEN}_{\mathcal{G}}(1^\lambda)$. In the second case, $\phi'_y = \phi_y$, so that no such procedure exists. □

References

[ABOE08] Dorit Aharonov, Micahel Ben-Or, and Elad Eban. Interactive Proofs For Quantum Computations. *Arxiv preprint arXiv:0810.5375*, 2008.

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

[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+18a] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. Certifiable randomness from a single quantum device. *arXiv preprint arXiv:1804.00640*, 2018.

[BCM+18b] 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.

[Chi01] A.M. Childs. Secure assisted quantum computation. *Arxiv preprint quant-ph/0111046*, 2001.

[DFPR13] Vedran Dunjko, Joseph F Fitzsimons, Christopher Portmann, and Renato Renner. Composable security of delegated quantum computation. *arXiv preprint arXiv:1301.3662*, 2013.

[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.

[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.

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

[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.

[GW07] Gus Gutoski and John Watrous. Toward a general theory of quantum games. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 565–574. ACM, 2007.

[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.

[KSVV02] Alexei Yu Kitaev, Alexander Shen, Mikhail N Vyalyi, 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.

[Mah18c] Urmila Mahadev. Classical verification of quantum computations. *arXiv preprint arXiv:1804.01082*, 2018.

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

[MYS12] Matthew McKague, Tzyh Haur Yang, and Valerio Scarani. Robust self-testing of the singlet. *Journal of Physics A: Mathematical and Theoretical*, 45(45):455304, 2012.

[Por17] Christopher Portmann. Composability in quantum cryptography, 2017. Tutorial given at QCRYPT'17, Cambridge, UK.

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

[Unr16] Dominique Unruh. Computationally binding quantum commitments. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 497–527. Springer, 2016.

[Win99] Andreas Winter. Coding theorem and strong converse for quantum channels. *IEEE Transactions on Information Theory*, 45(7):2481–2485, 1999.