Security Limitations of Classical-Client Delegated Quantum Computing

Christian Badertscher[1] , Alexandru Cojocaru[2], Léo Colisson[3] , Elham Kashefi[2,3], Dominik Leichtle[3], Atul Mantri[4], Petros Wallden[2]

[1] IOHK, Zurich, Switzerland
christian.badertscher@iohk.io
[2] School of Informatics, University of Edinburgh,
10 Crichton Street, Edinburgh EH8 9AB, UK
ekashefi@inf.ed.ac.uk, a.d.cojocaru@sms.ed.ac.uk, petros.wallden@ed.ac.uk
[3] Laboratoire d'Informatique de Paris 6 (LIP6), Sorbonne Université,
4 Place Jussieu, 75252 Paris CEDEX 05, France
leo.colisson@lip6.fr, dominik.leichtle@lip6.fr
[4] Joint Center for Quantum Information and Computer Science (QuICS),
University of Maryland, College Park, USA
amantri@umd.edu

Abstract. Secure delegated quantum computing is a two-party cryptographic primitive, where a computationally weak client wishes to delegate an arbitrary quantum computation to an untrusted quantum server in a privacy-preserving manner. Communication via quantum channels is typically assumed such that the client can establish the necessary correlations with the server to securely perform the given task. This has the downside that all these protocols cannot be put to work for the average user unless a reliable quantum network is deployed. Therefore the question becomes relevant whether it is possible to rely solely on classical channels between client and server and yet benefit from its quantum capabilities while retaining privacy. Classical-client remote state preparation ($\mathsf{RSP}_{\mathsf{CC}}$) is one of the promising candidates to achieve this because it enables a client, using only classical communication resources, to remotely prepare a quantum state. However, the privacy loss incurred by employing $\mathsf{RSP}_{\mathsf{CC}}$ as sub-module to avoid quantum channels is unclear.

In this work, we investigate this question using the Constructive Cryptography framework by Maurer and Renner [MR11]. We first identify the goal of $\mathsf{RSP}_{\mathsf{CC}}$ as the construction of ideal RSP resources from classical channels and then reveal the security limitations of using $\mathsf{RSP}_{\mathsf{CC}}$ in general and in specific contexts:

1. We uncover a fundamental relationship between constructing ideal RSP resources (from classical channels) and the task of cloning quantum states with auxiliary information. Any classically constructed ideal RSP resource must leak to the server the full classical description (possibly in an encoded form) of the generated quantum state, even if we target computational security only. As a consequence, we find that the realization of common RSP resources, without weakening their guarantees drastically, is impossible due to the no-cloning theorem.
2. The above result does not rule out that a specific $\mathsf{RSP}_{\mathsf{CC}}$ protocol can replace the quantum channel at least in some contexts, such as the Universal Blind Quantum Computing (UBQC) protocol of Broadbent et al. [BFK09]. However, we show that the resulting UBQC protocol cannot maintain its proven composable security as soon as $\mathsf{RSP}_{\mathsf{CC}}$ is used as a subroutine.
3. We show that replacing the quantum channel of the above UBQC protocol by the $\mathsf{RSP}_{\mathsf{CC}}$ protocol QFactory of Cojocaru et al. [CCKW19], preserves the weaker, game-based, security of UBQC.

Table of Contents

1 Introduction . . . 3
1.1 Overview of our Contributions . . . 4
1.2 Related Work . . . 6
2 Preliminaries . . . 7
2.1 The Constructive Cryptography Framework . . . 7
2.2 Notation . . . 9
3 Impossibility of Composable Classical RSP . . . 9
3.1 Remote State Preparation and Describable Resources . . . 9
3.2 Classically-Realizable RSP are Describable . . . 14
3.3 RSP Resources Impossible to Realize Classically . . . 15
3.4 Accepting the Limitations: Fully Leaky RSP resources . . . 17
4 Impossibility of Composable Classical-Client UBQC . . . 19
4.1 Impossibility of Composable $\mathsf{UBQC}_{\mathsf{CC}}$ on 1 Qubit . . . 21
4.2 Impossibility of Composable $\mathsf{UBQC}_{\mathsf{CC}}$ on Any Number of Qubits . . . 26
5 Game-Based Security of QF-UBQC . . . 26
5.1 Implementing Classical-Client UBQC with QFactory . . . 27
5.2 Single-Qubit QF-UBQC . . . 28
A Game-Based Security and Constructive Cryptography . . . 34
B QFactory: Remote State Preparation, Revisited . . . 36
B.1 4-states and 8-states QFactory protocol . . . 36
B.2 Correctness of QFactory . . . 37
B.3 Security of QFactory . . . 38
C Distance Measures for Quantum States . . . 39

1 Introduction

The expected rapid advances in quantum technologies in the decades to come are likely to further disrupt the field of computing. To fully realize the technological potential, remote access, and manipulation of data must offer strong privacy and integrity guarantees and currently available quantum cloud platform designs have still a lot of room for improvement.

There is a large body of research that exploits the client-server setting defined in [Chi05] to offer different functionalities, including secure delegated quantum computation [BFK09,MF12,DFPR14,Bro15a,Mah18a] [5], verifiable delegated quantum computation [ABOE08,RUV12,FK17,HM15,Bro15b,FHM18,TMM+18,Mah18b] [6], secure multiparty quantum computation [KP17,KMW17,KW17], quantum fully homomorphic encryption [BJ15,DSS16]. It turns out that one of the central building blocks is secure *remote state preparation* (RSP) that was first defined in [DKL12]. At a high level, RSP resources enable a client to remotely prepare a quantum state on the server and are, therefore, the natural candidate to replace quantum channel resources in a modular fashion. These resources further appear to enable a large ecosystem of composable protocols [DKL12,DFPR14], including in particular the important *Universal Blind Quantum Computation* (UBQC) [BFK09] protocol used to delegate a computation to a remote quantum server who has no knowledge of the ongoing computation.

However, in most of the above-mentioned works, the users and providers do have access to quantum resources to achieve their goals, in particular to quantum channels in addition to classical communication channels. This might prove to be challenging for some quantum devices, e.g. those with superconducting qubits, and in general, it also restricts the use of these quantum cloud services to users with suitable quantum technology. Motivated by this practical constrain, [CCKW18] introduced a protocol mimicking this remote state preparation resource over a purely *classical* channel (under the assumption that learning with error problem is computationally hard for quantum servers). This is a cryptographic primitive between a fully classical client and a server (with a quantum computer). By the end of the interactive protocol the client has "prepared" remotely on the server's lab, a quantum state (typically a single qubit $|+_\theta\rangle := \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$). This protocol further enjoys some important privacy guarantees with respect to the prepared state.

The important role of such a classical RSP primitive as part of larger protocols – most notably in their role in replacing quantum channels between client and server – stems from their ability to make the aforementioned protocols available to classical users, in particular clients without quantum-capable infrastructure on their end. It is therefore of utmost importance to develop an understanding of this primitive, notably its security guarantees when composed in larger contexts such as in [GV19].

In this paper, we initiate the study of analyzing classical remote state-preparation from first principles. We thereby follow the Constructive Cryptography (CC) framework [MR11,Mau11] to provide a clean treatment of the RSP primitive from a composable perspective. (Note that the framework is also referred to as Abstract Cryptography (AC) in earlier works.) Armed with such a definition, we then investigate the limitations and possibilities of using classical RSP both in general and in more specific contexts. Using CC is a common approach to analyze classical as well as quantum primitives and their composable security guarantees in general and in related works including [DFPR14,DK16,MK13].

[5] For more details see review of this field in [Fit17]

[6] For more details see recent reviews in [GKK19,Vid20]

1.1 Overview of our Contributions

We present an informal overview of our main results. In this work, we cover the security of $\mathsf{RSP}_{\mathsf{CC}}$, the class of remote state preparation protocols which only use a classical channel, and the use-case that corresponds to its arguably most important application: Universal Blind Quantum Computing (UBQC) protocols with a completely classical client. More specifically, we analyze the security of $\mathsf{UBQC}_{\mathsf{CC}}$, the family of protocols where a protocol in $\mathsf{RSP}_{\mathsf{CC}}$ is used to replace the quantum channel from the original quantum-client UBQC protocol. An example of an RSP resource is the $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ [7] resource (depicted in Fig. 1) outputting the quantum state $|+_\theta\rangle$ on its right interface, and the classical description of this state, θ, on its left interface.

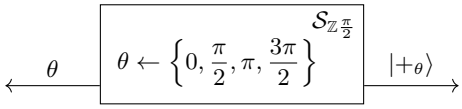


Fig. 1: Ideal resource $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$

We show in Section 3 a wide-ranging limitation to the universally composable guarantees that any protocol in the family $\mathsf{RSP}_{\mathsf{CC}}$ can achieve. The limitation follows just from the relation between (i) the notion of classical realization and (ii) a property we call describability – which roughly speaking measures how leaky an RSP resource is. The limitation directly affects the amount of additional leakage on the classical description of the quantum state. In this way, it rules out a wide set of desirable resources, even against computationally bounded distinguishers.

Theorem 3.6 (Security Limitations of $\mathsf{RSP}_{\mathsf{CC}}$). *Any* RSP *resource, realizable by an* $\mathsf{RSP}_{\mathsf{CC}}$ *protocol with security against quantum polynomial-time distinguishers, must leak an encoded, but complete description of the generated quantum state to the server.*

The importance of Theorem 3.6 lies in the fact that it is drawing a connection between the composability of an $\mathsf{RSP}_{\mathsf{CC}}$ protocol – a *computational* notion – with the statistical leakage of the ideal functionality it is constructing – an *information-theoretic* notion. This allows us to use fundamental physical principles such as no-cloning or no-signaling in the security analysis of *computationally* secure $\mathsf{RSP}_{\mathsf{CC}}$ protocols. As one direct application of this powerful tool, we show that secure implementations of the ideal resource in Fig. 1 give rise to the construction of a quantum cloner, and are hence impossible.

Proof sketch. While Theorem 3.6 applies to much more general RSP resources having arbitrary behavior at its interfaces and targeting any output quantum state, for simplicity we exemplify the main ideas of our proof for the ideal resource $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$.

The composable security of a protocol realizing $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ implies, by definition, the existence of a simulator σ which turns the right interface of the ideal resource into a completely classical interface as depicted in Fig. 2. Running the protocol of the honest server with access to this classical interface allows the distinguisher to reconstruct the quantum state $|+_\theta\rangle$ the simulator received from the ideal resource. Since the distinguisher also has access to θ via the left interface of the ideal resource, he can perform a simple measurement to verify the consistency of the state obtained after interacting with the simulator. By the correctness of the protocol, the obtained quantum state $|+_\theta\rangle$ must therefore indeed comply with θ. We emphasize that this consistency check can be performed efficiently, i.e. by *polynomially-bounded* quantum distinguishers.

[7] The notation $\mathbb{Z}_{\frac{\pi}{2}}$ denotes the set of the 4 angles $\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$.

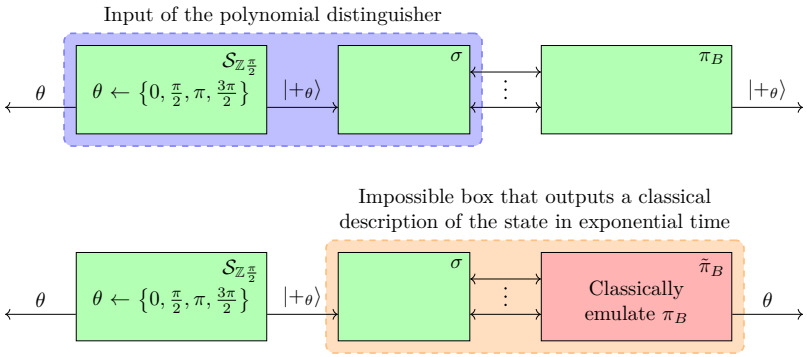


Fig. 2: Idea of the proof of impossibility of composable $\mathsf{RSP}_{\mathsf{CC}}$, exemplified by the $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ primitive from Fig. 5. The green boxes run in polynomial time, while the red box runs in exponential time. $\tilde{\pi}_B$ runs the same computations as π_B by emulating it. In this way, the classical description of the quantum state can be extracted.

Since the quantum state $|+_\theta\rangle$ is transmitted from σ to the distinguisher over a classical channel, the ensemble of exchanged classical messages must contain a complete encoding of the description of the state, θ. A (possibly unbounded) algorithm can hence extract the actual description of the state by means of a classical emulation of the honest server. This property of the ideal resource is central to our proof technique, we call it *describability*. □

Having a full description of the quantum state produced by $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ would allow us to clone it, a procedure prohibited by the no-cloning theorem. We conclude that the resource $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ cannot be constructed from a classical channel only.

One could attempt to modify the ideal resource, to incorporate such an extensive leakage, which is necessary as the above proof implies. However, this yields an ideal resource that is actually not a useful idealization or abstraction of the real world (because it is fully leaky) which puts in question whether they are at all useful in a composable analysis. Consider for example constructions of composite protocols that utilize the (non-leaky) ideal resource as a sub-module. These constructions require a fresh security analysis if the sub-module is replaced by any leaky version of it, but since the modified resource is very specific and must mimic its implementation (in terms of leakage) it appears that this replacement does not give any benefit compared to directly using the implementation as a subroutine and then examining the compsoable security of the combined protocol as a whole. This latter way is therefore examined next. More precisely, we might still be able to use $\mathsf{RSP}_{\mathsf{CC}}$ protocols as a subroutine in other, specific protocols, and expect the overall protocol to still construct a useful ideal functionality. The protocol family $\mathsf{UBQC}_{\mathsf{CC}}$ is such an application. Unfortunately, as we show in Section 4, $\mathsf{UBQC}_{\mathsf{CC}}$ fails to provide the expected composable security guarantees once classical remote state preparation is used to replace the quantum channel from client to server (where composable security for UBQC refers to the goal of achieving the established ideal functionality of [DFPR14] which we recall in Section 4). This holds even if the distinguisher is computationally bounded.

Theorem 4.10 (Impossibility of $\mathsf{UBQC}_{\mathsf{CC}}$). *No* $\mathsf{RSP}_{\mathsf{CC}}$ *protocol can replace the quantum channel in the* UBQC *protocol while preserving composable security.*

Proof sketch. We first show that the existence of any composable $\mathsf{UBQC}_{\mathsf{CC}}$ protocol (in the sense of achieving the ideal UBQC resource) implies the existence of a composable *single-qubit* $\mathsf{UBQC}_{\mathsf{CC}}$ protocol. In turn, the impossibility of composable single-qubit

$\mathsf{UBQC}_{\mathsf{CC}}$ protocols is then proven in two steps. First, we show that single-qubit $\mathsf{UBQC}_{\mathsf{CC}}$ protocols can, in fact, be turned into RSP protocols. This allows us to employ the toolbox we developed before on RSP protocols. As a second step, we deduce that an RSP protocol of this specific kind (that leaks the classical description, even in the form of an encoded message) would violate the no-signaling principle, thereby showing that a composable $\mathsf{UBQC}_{\mathsf{CC}}$ protocol could not have existed in the first place. □

Finally in Section 5, we show that the protocol family $\mathsf{RSP}_{\mathsf{CC}}$ is not trivial with respect to privacy guarantees. It contains protocols with reasonably restricted leakage that can be used as subroutines in specific applications resulting in combined protocols that offer a decent level of security. Specifically, we prove the blindness property of $\mathsf{QF\text{-}UBQC}$, a concrete $\mathsf{UBQC}_{\mathsf{CC}}$ protocol that consists of the universal blind quantum computation (UBQC) protocol of [BFK09] and the specific LWE-based remote state preparation ($\mathsf{RSP}_{\mathsf{CC}}$) protocol from [CCKW19]. This yields the first provably secure $\mathsf{UBQC}_{\mathsf{CC}}$ protocol from standard assumptions with a classical RSP protocol as a subroutine.

Theorem 5.3 (Game-Based Security of $\mathsf{QF\text{-}UBQC}$). *The universal blind quantum computation protocol with a classical client* $\mathsf{UBQC}_{\mathsf{CC}}$ *that combines the* $\mathsf{RSP}_{\mathsf{CC}}$ *protocol of* [CCKW19] *and the* UBQC *protocol of* [BFK09] *is adaptively blind in the game-based setting. We call this protocol* $\mathsf{QF\text{-}UBQC}$*. This protocol is secure under standard assumptions.*

The statement of Theorem 5.3 can be summarized as follows: No malicious (but computationally bounded) server in the $\mathsf{QF\text{-}UBQC}$ protocol could distinguish between two runs of the protocol performing different computations. This holds even when it is the adversary that chooses the two computations that he will be asked to distinguish. The security is achieved in the plain model, i.e., without relying on additional setup such as a measurement buffer. The protocol itself is a combination of UBQC with the QFactory protocol. For every qubit that the client would transmit to the server in the original UBQC protocol, QFactory is invoked as a subprocedure to the end of remotely preparing the respective qubit state on the server over a classical channel.

Proof sketch. By a series of games, we show that the real protocol on a single qubit is indistinguishable from a game where the adversary guesses the outcome of a hidden coin flip. We generalize this special case to the full protocol on graphs with a polynomial number of qubits by induction over the size of the graph. □

1.2 Related Work

While $\mathsf{RSP}_{\mathsf{CC}}$ was first introduced in [CCKW18] (under a different terminology), (game-based) security was only proven against weak (honest-but-curious) adversaries. Security against malicious adversaries was proven for a modified protocol in [CCKW19][8], this protocol, called *QFactory*, is the basis of the positive results in this work. In parallel [GV19] gave another protocol that offers a stronger notion of *verifiable* $\mathsf{RSP}_{\mathsf{CC}}$ and proved the security of their primitive in the CC framework. The security analysis, however, requires an assumption of *measurement buffer* resource in addition to the classical channel to construct a verifiable $\mathsf{RSP}_{\mathsf{CC}}$. Our result confirms that the measurement buffer resource is a strictly non-classical assumption.

[8] In [CCKW19] a verifiable version of $\mathsf{RSP}_{\mathsf{CC}}$ was also given, but security was not proven in full generality.

In the information-theoretic setting with perfect security[9], the question of secure delegation of quantum computation with a completely classical client was first considered in [MK14]. The authors showed a negative result by presenting a *scheme-dependent* impossibility proof. This was further studied in [DK16,ACGK19] which showed that such a classical delegation would have implications in computational complexity theory. To be precise, [ACGK19] conjecture that such a result is unlikely by presenting an oracle separation between BQP and the class of problems that can be classically delegated with perfect security (which is equivalent to the complexity class $\mathrm{NP}/\mathrm{poly} \cap \mathrm{coNP}/\mathrm{poly}$ as proven by [AFK87]). On the other hand, a different approach to secure delegated quantum computation with a completely classical client, without going via the route of $\mathsf{RSP}_{\mathsf{CC}}$, was also developed in [MDMF17] where the server is unbounded and in [Mah18a,Bra18] with the bounded server. The security was analysed for the overall protocol (rather than using a module to replace quantum communication). It is worth noting that [MDMF17] is known to be not composable secure in the Constructive Cryptography framework [Man19].

2 Preliminaries

We assume basic familiarity with quantum computing, for a detailed introduction see [NC00] (note that in this paper all Hilbert spaces are assumed to have a finite dimension). We just formalize here what we mean in this paper by *quantum instrument*, which is a concept introduced by [DL70], and which is a generalization of completely positive trace preserving (CPTP) maps to maps having both classical and quantum outputs:

Definition 2.1 (Quantum Instrument). *A map $\Lambda : \mathbb{C}^{n\times n} \to \{0,1\}^{m_1} \times \mathbb{C}^{m_2\times m_2}$ is said to be a quantum instrument if there exists a collection $\{\mathcal{E}_y\}_{y\in\{0,1\}^{m_1}}$ of trace-non-increasing completely positive maps such that the sum is trace-preserving (i.e. for any positive operator ρ, $\sum_y \mathcal{E}_y(\rho) = \mathrm{Tr}(\rho)$), and, if we define $\rho_y = \frac{\mathcal{E}_y(\rho)}{\mathrm{Tr}(\mathcal{E}_y(\rho))}$, then $\Pr[\,\Lambda(\rho) = (y,\rho_y)\,] = \mathrm{Tr}(\mathcal{E}_y(\rho))$.*

2.1 The Constructive Cryptography Framework

The Constructive Cryptography (CC) framework (also sometimes referred to as the Abstract Cryptography (AC) framework) introduced by Maurer and Renner [MR11] is a top-down and axiomatic approach, where the desired functionality is described as an (ideal) *resource* $\mathcal{S}$ with a certain input-output behavior independent of any particular implementation scheme. A resource has some interfaces $\mathcal{I}$ corresponding to the different parties that could use the resource. In our case, we will have only two interfaces corresponding to Alice (the client) and Bob (the server), therefore $\mathcal{I} = \{A, B\}$. Resources are not just used to describe the desired functionality (such as a perfect state preparation resource), but also to model the assumed resources of a protocol (e.g., a communication channel). The second important notion is the *converter* which, for example, are used to define a protocol. Converters always have two interfaces, an inner and an outer one, and the inner interface can be connected to the interface of a resource. For example, if $\mathcal{R}$ is a resource and π_A, $\pi_B \in \Sigma$ are two converters (corresponding to a given protocol making use of resource $\mathcal{R}$) we can connect these two converters to the interface A and B, respectively, (the resulting object being a resource as well) using the following notation: $\pi_A\mathcal{R}\pi_B$.

In order to characterize the distance between two resources (and therefore the security), we use the so-called *distinguishers*. We then say that two resources $\mathcal{S}_1$ and $\mathcal{S}_2$

[9] By perfect security we mean at most input size is allowed to be leaked

are indistinguishable (within ε), and denote it as $\mathcal{S}_1 \approx_\varepsilon \mathcal{S}_2$, if no distinguisher can distinguish between $\mathcal{S}_1$ and $\mathcal{S}_2$ with an advantage greater than ε. In the following, we will mostly focus on quantum-polynomial-time (QPT) distinguishers.

Central to Constructive Cryptography is the notion of a secure construction of an (ideal) resource $\mathcal{S}$ from an assumed resource $\mathcal{R}$ by a protocol (specified as a pair of converters). We directly state the definition for the special case we are interested in, namely in two-party protocols between a client A and a server B, where A is always considered to be honest. The definition can therefore be simplified as follows:

Definition 2.2 (See [Mau11,MR11]). *Let $\mathcal{I} = \{A, B\}$ be a set of two interfaces (A being the left interface and B the right one), and let $\mathcal{R}, \mathcal{S}$ be two resources. Then, we say that for the two converters π_A, π_B, the protocol $\boldsymbol{\pi} := (\pi_A, \pi_B)$* (securely) constructs *$\mathcal{S}$ from $\mathcal{R}$ within ε, or that $\mathcal{R}$* realizes *$\mathcal{S}$ within ε, denoted:*

$$\mathcal{R} \xrightarrow[\varepsilon]{\boldsymbol{\pi}} \mathcal{S} \tag{1}$$

if the following two conditions are satisfied:

- *Availability (i.e. correctness):*

$$\pi_A \mathcal{R} \pi_B \approx_\varepsilon \mathcal{S} \vdash \tag{2}$$

  (where $\vdash$ represents a filter, i.e. a trivial converter that enforces honest/correct behavior [10], and $A \approx_\varepsilon B$ means that no polynomial quantum distinguisher can distinguish between A and B (given black-box access to A or B) with an advantage better than ε)
- *Security: there exists $\sigma \in \Sigma$ (called a simulator) such that:*

$$\pi_A \mathcal{R} \approx_\varepsilon \mathcal{S} \sigma \tag{3}$$

We also extend this definition when ε is a function $\varepsilon : \mathbb{N} \to \mathbb{R}$: we say that $\mathcal{S}$ is ε-classically-realizable *if for any $n \in \mathbb{N}$, $\mathcal{S}$ is $\varepsilon(n)$-realizable*[11].

The intuition behind this definition is that if no distinguisher can know whether he is interacting with an ideal resource or with the real protocol, then it means that any attack done in the "real world" can also be done in the "ideal world". Because the ideal world is secure by definition, so is the real world. Using such a definition is particularly useful to capture the "leakage" of information to the server. This is quite subtle to capture in the real world, but very natural in the ideal world.

In our work, we instantiate a general model of computation to capture general quantum computations within converters which ensures that they follow the laws of quantum physics (e.g., excluding that the input-output behavior is signaling). Indeed, without such a restriction, we could not base our statements on results from quantum physics, because an arbitrary physical reality must not respect them, such as cloning of quantum states, signaling, and more. More specifically, in this work, we assume that any converter that interacts classically on its inner interface and outputs a single quantum message on its outer interface can be represented as a sequence of quantum instruments (which is a generalization of CPTP maps taking into account both quantum and classical outputs, see Definition 2.1) as represented in Fig. 4 and constitutes the most general expression

[10] Usually, a filter simply sends a bit $c = 0$ and then forwards all communications between its two interfaces (this filter will be denoted by $\vdash^{c=0}$), but it could be a more general converter. When the filter is not clear from the context, we need to specify also which filter we consider.

[11] Note that here the protocols $\pi_A^{(n)}$ and $\pi_B^{(n)}$ may or may not be efficient to compute given n, so our nogo-result will apply to non-uniform circuits, and therefore also to uniform circuits.

of allowed quantum operations. More precisely, this model takes into account interactive converters (and models the computation in sequential dependent stages). This is similar to if one would in the classical world instantiate the converter by a sequence of classical Turing machines (passing state to each other) [Gol01]. For more details and to see why such definitions are enough to provide composability, see Appendix A.

2.2 Notation

We denote by $\mathbb{Z}_{\frac{\pi}{2}}$ the set of the 4 angles $\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$, and $\mathbb{Z}_{\frac{\pi}{4}} = \{0, \frac{\pi}{4}, ..., \frac{7\pi}{4}\}$ the similar set of 8 angles. If ρ is a quantum state, $[\rho]$ is the *classical* representation (as a density matrix) of this state. We also denote the quantum state $|+_\theta\rangle := \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$, where $\theta \in \mathbb{Z}_{\frac{\pi}{4}}$, and for any angle θ, $[\theta]$ will denote $[|+_\theta\rangle\langle+_\theta|]$, i.e. the classical description of the density matrix corresponding to $|+_\theta\rangle$. For a protocol $\mathcal{P} = (P_1, P_2)$ with two interacting algorithms P_1 and P_2 denoting the two participating parties, let $r \leftarrow \langle P_1, P_2\rangle$ denote the execution of the two algorithms, exchanging messages, with output r. We use the notation $\mathcal{C}$ to denote the *classical channel* resource, that just forwards classical messages between the two parties.

3 Impossibility of Composable Classical RSP

In this section, we first define the general notion of what RSP tries to achieve in terms of resources and subsequently quantify information that an ideal RSP resource must leak at its interface to the server even if the distinguisher is computationally bounded. One would expect, that against bounded distinguisher, the resource can express clear privacy guarantees, which we prove cannot be the case.

The reason is roughly as follows: assuming that there exists a simulator making the ideal resource indistinguishable from the real protocol, we can exploit this fact to construct an algorithm that can classically describe the quantum state given by the ideal resource. It is not difficult to verify that there could exist an inefficient algorithm (i.e. with exponential run-time) that achieves such a task. We show that even a computationally bounded distinguisher can distinguish the real protocol from the ideal protocol whenever a simulator's strategy is independent of the classical description of the quantum state. This would mean that for an RSP protocol to be composable there must exist a simulator that possesses at least a classical transcript encoding the description of a quantum state. This fact coupled with the quantum no-cloning theorem implies that the most meaningful and natural RSP resources cannot be realized from a classical channel alone. We finally conclude the section by looking at the class of imperfect (describable) RSP resources which avoid the no-go result at the price of being "fully-leaky", not standard, and having an unfortunately unclear composable security.

3.1 Remote State Preparation and Describable Resources

We first introduce, based on the standard definition in the Constructive Cryptography framework, the notion of *correctness* and *security* of a two-party protocol which constructs (realizes) a resource from a *classical* channel $\mathcal{C}$.

Definition 3.1 (Classically-Realizable Resource). *An ideal resource $\mathcal{S}$ is said to be* ε-classically-realizable *if it is realizable (in the sense of Definition 2.2) from a* classical *channel, i.e. if there exists a protocol $\boldsymbol{\pi} = (\pi_A, \pi_B)$ between two parties (interacting classically) such that:*

$$\mathcal{C} \xrightarrow[\varepsilon]{\boldsymbol{\pi}} \mathcal{S} \tag{4}$$

We would like to point out that since Alice is honest, this definition incorporates already the case when Alice and Bob share purely classical resources that are achievable by Alice emulating the resource and sending Bob's output over a classical channel.
A simple ideal prototype that captures the goal of a RSP protocol could be phrased as follows: the resource outputs a quantum state (chosen from a set of states) on one interface and classical description of that state on the other interface to the client. For our purposes, this view is too narrow and we want to generalize this notion. For instance, a resource could accept some inputs from the client or interact with the server and be powerful enough to comply with the above basic behavior if both follow the protocol. We would like to capture that any resource can be seen as an RSP resource as soon as we fix a way to efficiently convert the client and server interfaces to comply with the basic prototype. To make this formal, we need to introduce some converters that will witness this:

1. A converter $\mathcal{A}$ will output, after interacting with the ideal resource[12], a classical description $[\rho]$ which is one of the following:
   (a) A density matrix (positive and with trace 1) corresponding to a quantum state ρ.
   (b) The null matrix, which is useful to denote the fact that we detected some deviation that should not happen in an honest run.
2. A converter $\mathcal{Q}$, whose goal is to output a quantum state ρ' as close as possible to the state ρ output by $\mathcal{A}$.
3. A converter $\mathcal{P}$, whose goal is to output a classical description $[\rho']$ of a quantum state ρ' which is on average "close" to ρ.

An RSP must meet two central criteria:

1. Accuracy of the classical description of the obtained quantum state: We require that the quantum state ρ described by $\mathcal{A}$'s output is close to $\mathcal{Q}$'s output ρ'. This is to be understood in terms of the trace distance.
2. Purity of the obtained quantum state: Since the RSP resource aims to replace a noise-free quantum channel, it is desirable that the quantum state output by $\mathcal{Q}$ admit a high degree of purity, i.e. more formally, that $\mathrm{Tr}\left(\rho'^2\right)$ be close to one. Since ρ' is required to be close to ρ, this implies a high purity of ρ as well.

It turns out that these two conditions can be unified and equivalently captured requiring that the quantity $\mathrm{Tr}(\rho\rho')$ is close to one. A rigorous formulation of this claim and its proof is provided by Lemma C.3.

We can also gain a more operational intuition of the notion of RSP by considering that an RSP resource (together with $\mathcal{A}$ and $\mathcal{Q}$) can be seen, not only as a box that produces a quantum state together with its description but also as a box whose accuracy can be easily *tested*[13]. For example, if such a box produces a state ρ', and pretends that the description of that state corresponds to $|\phi\rangle$ (i.e. $[\rho] = [|\phi\rangle\langle\phi|]$), then the natural way to test it would be to measure ρ' by doing a projection on $|\phi\rangle$. This test would pass with probability $p_s := \langle\phi|\rho'|\phi\rangle$, and therefore if the box is perfectly accurate (i.e. if $\rho' = |\phi\rangle\langle\phi|$), the test will always succeed. However, when ρ' is far from $|\phi\rangle\langle\phi|$, this test is unlikely to pass, and we will have $p_s < 1$. We can then generalise this same idea for arbitrary (eventually not pure) states by remarking that $p_s = \langle\phi|\rho'|\phi\rangle = \mathrm{Tr}(|\phi\rangle\langle\phi|\rho') = \mathrm{Tr}(\rho\rho')$. Indeed, this last expression corresponds[14] exactly to the probability of outputting E_0

[12] $\mathcal{A}$ is allowed to interact with the (ideal) resource in a non-trivial manner. However, $\mathcal{A}$ will often be the trivial converter in the sense that it simply forwards the output of the ideal resource, or – when the resource waits for a simple activation input – picks some admissible value as input to the ideal resource and forwards the obtained description to its outer interface.

[13] This testable property will be of great importance in our argument later.

[14] Note that it also turns out to be equal to the (squared) fidelity between ρ and ρ' when ρ is pure.

when measuring the state ρ' according to the POVM $\{E_0 := \rho, E_1 := I - \rho\}$, and since the classical description of ρ is known, it is possible to perform this POVM and test the (average) accuracy of our box. This motivates the following definition for general RSP resources.

Definition 3.2 (RSP resources). *A resource $\mathcal{S}$ is said to be a* remote state preparation resource *within ε with respect to converters $\mathcal{A}$ and $\mathcal{Q}$ if the following three conditions hold: (1) both converters output a single message at the outer interface, where the output $[\rho]$ of $\mathcal{A}$ is classical and is either a density matrix or the null matrix, and the output ρ' of $\mathcal{Q}$ is a quantum state; (2) the equation:*

$$\mathop{\mathbb{E}}_{([\rho],\rho') \leftarrow \mathcal{A}\mathcal{S} \vdash \mathcal{Q}} [\operatorname{Tr}(\rho\rho')] \geq 1 - \varepsilon \tag{5}$$

is satisfied, where the probability is taken over the randomness of $\mathcal{A}$, $\mathcal{S}$ and $\mathcal{Q}$, and finally, (3) for all the possible outputs $[\rho]$ of $([\rho], \rho') \leftarrow \mathcal{A}\mathcal{S} \vdash \mathcal{Q}$, if we define $E_0 = \rho$, $E_1 = I - \rho$, then the POVM $\{E_0, E_1\}$ must be efficiently implementable[15] by any distinguisher.

Whenever we informally speak of a resource $\mathcal{S}$ as being an RSP resource, this has to be understood always in a context where the converters $\mathcal{A}$ and $\mathcal{Q}$ are fixed.

Describable resources. So far, we have specified that a resource qualifies as an RSP resource if, when all parties follow the protocol, we know how to compute a quantum state on the right interface and classical description of a "close" state on the other interface. A security-related question now is, if it is also possible to extract (possibly inefficiently) from the right interface a *classical* description of a quantum state that is close to the state described by the client. If we find a converter $\mathcal{P}$ doing this, we would call the (RSP) resource *describable*. The following definition captures this.

Definition 3.3 (Describable Resource). *Let $\mathcal{S}$ be a resource and $\mathcal{A}$ a converter outputting a single classical message $[\rho]$ on its outer interface (either equal to a density matrix or the null matrix). Then we say that $(\mathcal{S}, \mathcal{A})$ is* ε-describable *(or, equivalently, that $\mathcal{S}$ is describable within ε with respect to $\mathcal{A}$) if there exists a (possibly unbounded) converter $\mathcal{P}$ (outputting a single classical message $[\rho']$ on its outer interface representing a density matrix) such that:*

$$\mathop{\mathbb{E}}_{([\rho],[\rho']) \leftarrow \mathcal{A}\mathcal{S}\mathcal{P}} [\operatorname{Tr}(\rho\rho')] \geq 1 - \varepsilon \tag{6}$$

(the expectation is taken over the randomness of $\mathcal{S}$, $\mathcal{A}$ and $\mathcal{P}$).

Reproducible converters. In the proof of our first result, we will encounter a crucial decoding step. Roughly speaking, the core of this decoding step is the ability to convert the classical interaction with a client, which can be seen as an arbitrary encoding of a quantum state, back into an explicit representation of the state prepared by the server. The ability of such a conversion can be phrased by the following definition.

Definition 3.4 (Reproducible Converter). *A converter π that outputs (on the right interface) a quantum state ρ is said to be reproducible if there exists a (possibly inefficient) converter $\tilde{\pi}$ such that:*

1. *the outer interface of $\tilde{\pi}$ outputs only a classical message $[\rho']$*

[15] We could also define a similar definition when this POVM can only be approximated (for example because the distinguishers can only perform quantum circuits using a finite set of gates) and the theorems would be similar, up to this approximation, but for simplicity we will stick to that setting.

2. *the converter π is perfectly indistinguishable from $\tilde{\pi}$ against any unbounded distinguisher $D \in \mathcal{D}^u$, up to the conversion of the classical messages $[\rho']$ into a quantum state ρ'. More precisely, if we denote by $\mathcal{T}$ the converter that takes as input on its inner interface a classical description $[\rho']$ of a quantum state and outputs that quantum state ρ' (as depicted in Fig. 3), we have:*

$$\mathcal{C}\pi \approx_0^{\mathcal{D}^u} \mathcal{C}\tilde{\pi}\mathcal{T} \tag{7}$$

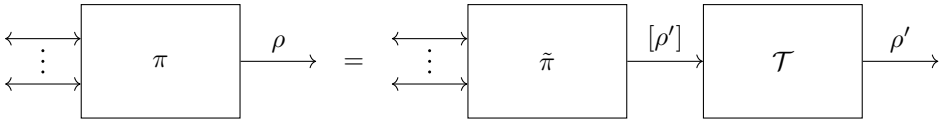


Fig. 3: Reproducible converter.

Classical communication and reproducibility. We see that in general, being reproducible is a property that stands in conflict with the quantum no-cloning theorem. More precisely, the ability to reproduce implies that there is a way to extract knowledge of a state sufficient to clone it. However, whenever communication is classical, quite the opposite is true. This is formalized in the following lemma. Intuitively, it says that in the principle it is always possible to compute the exact description of the state from the classical transcript and the *quantum instruments* (circuit) used to implement the action of the converter, where an instrument is a generalized CPTP map which allows a party to output both a quantum and a classical state and is formalized more precisely in Definition 2.1. Recall that this is the most general way of representing a quantum operation.

In the proof, we just need to assume that π interacts (classically) with the inner interface first, and finally outputs a quantum state on the outer interface, so for simplicity we will stick to that setting. In this way we can decompose π as depicted in Fig. 4 using the following notation:

$$\pi := (\pi_i)_i \tag{8}$$

Each π_i represents a round, and we denote with $(y_i, \rho_{i+1}) \leftarrow \pi_i(x_i, \rho_i)$ the output of the i-th round, assuming that $x_i \in \{0,1\}^{l_i}$ is a classical input message sent from the inner interface, ρ_i is the internal quantum state (density matrix) after round $i-1$, ρ_{i+1} is the internal state after round i, and $y_i \in \{0,1\}^{l'_i} \cup \perp$ is a classical message, sent to the inner interface when $y_i \neq \perp$. For the first protocol, we set $\rho_0 = (1)$, which is the trivial density matrix of dimension 1. Moreover, when $y_i = \perp$, we do not send any message to the inner interface and instead we send ρ_{i+1} to the outer interface and we stop the protocol. Note that if we want to let π send the first message instead of receiving it, we can set $x_0 = \perp$, and similarly, if the last message is sent instead of received, we can add one more round where we set $x_{n+1} = \perp$.

Now, we can prove that a party, that produces a quantum state at the end of a protocol with exclusively classical communication, is reproducible:

Lemma 3.5. *Let $\pi = (\pi_i)_i$ (using the notation introduced Eq. (8)) be a converter such that:*

1. *it receives and sends only classical messages from the inner interfaces*
2. *it outputs at the end a quantum state on the outer interface*
3. *each π_i is a quantum instrument*

then π is reproducible.

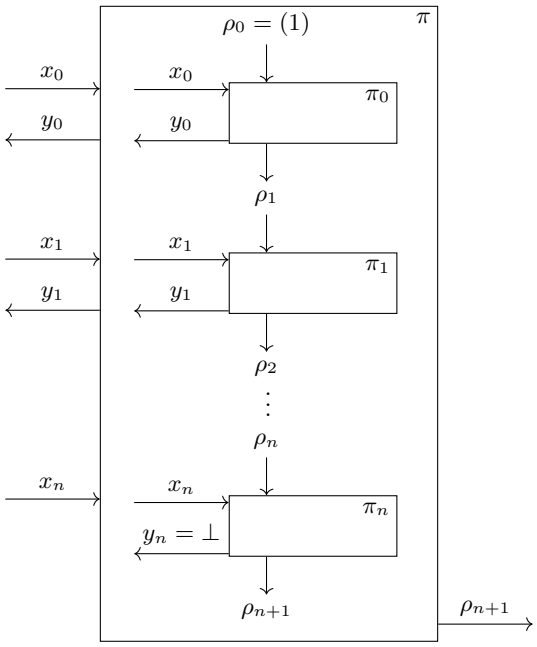


Fig. 4: Representation of an interactive protocol π into a sequence of quantum instruments.

Proof. The intuition behind the proof is to argue that because the only interactions with the outside world are classical as seen from Fig. 4, the internal state of π can always be computed (in exponential time) manually.
More precisely, for all i, because π_i is a quantum instrument, there exists a set $\{\mathcal{E}_{y_i}\}$ of maps having the properties defined in Definition 2.1. And because for all y_i, $\mathcal{E}_{y_i}$ is completely positive, there exists a finite set of matrices $\{B_k^{(i,y_i)}\}_k$, known as Kraus operators, such that we have for all ρ (and in particular for $\rho = |x_i\rangle\langle x_i| \otimes \rho_i$):

$$\mathcal{E}_{y_i}(\rho) = \sum_k B_k^{(i,y_i)} \rho B_k^{(i,y_i)\dagger} \tag{9}$$

Therefore, for all x_i, ρ_i and y_i, we have with probability $p_{y_i} := \mathrm{Tr}(\mathcal{E}_{y_i}(|x_i\rangle\langle x_i| \otimes \rho_i))$:

$$\pi_i(x_i, \rho_i) = (y_i, \mathcal{E}_{y_i}(|x_i\rangle\langle x_i| \otimes \rho_i)) \tag{10}$$

$$= (y_i, \underbrace{\sum_k B_k^{(i,y_i)}(|x_i\rangle\langle x_i| \otimes \rho_i)B_k^{(i,y_i)\dagger}}_{\rho_{i+1}}) \tag{11}$$

We remark that if we know $[\rho_i]$, the coefficients of the matrix ρ_i, then for all y_i we can compute the probability p_{y_i} of outputting y_i, and the corresponding $[\rho_{i+1}]$, (the coefficients of the matrix ρ_{i+1}) by just doing the above computation. So to construct $\tilde{\pi}$ (using notations from Definition 3.4) we do as follows:

- first, for all i we construct $\tilde{\pi}_i$, which on input $(x_i, [\rho_i])$ outputs $(y_i, [\rho_{i+1}])$ with probability p_{y_i} using the formula Eq. (11).
- then, we define $\tilde{\pi}$ as $(\tilde{\pi}_i)$ with $[\rho_0] = (1)$.

Then, we trivially have $\mathcal{C}\pi \approx_0 \mathcal{C}\tilde{\pi}\mathcal{T}$, even for unbounded distinguishers, because $\tilde{\pi}$ is exactly the same as π, except that the representations of the quantum states in $\tilde{\pi}$ are matrices, while they are actual quantum states in π. Therefore, adding $\mathcal{T}$ (which turns any $[\rho_i]$ into ρ_i) on the outer interface (which is the only interface that sends a classical state $[\rho_i]$) gives us $\pi \approx_0 \mathcal{C}\tilde{\pi}\mathcal{T}$. □

3.2 Classically-Realizable RSP are Describable

In this section we show our main result about remote state preparation resources, which interestingly links a constructive notion (*composability*) with respect to a computational notion with an information theoretic property (*describability*).

This implies directly the *impossibility result* regarding the existence of non-describable $\mathsf{RSP}_{\mathsf{CC}}$ composable protocols (secure against *bounded* BQP distinguishers). While this theorem does not rule out all the possible RSP resources, it shows that most "*useful*" RSP resources are impossible. Indeed, the describable property is usually not a desirable property, as it means that an unbounded adversary could learn the description of the state he received from an ideal resource. To illustrate this theorem, we will see in the Section 3.3 some examples showing how this result can be used to prove the impossibility of classical protocols implementing some specific resources, and in Section 3.4 we will see some example of "imperfect" resources escaping the impossibility result.

Theorem 3.6 (Classically-Realizable RSP are Describable). *If an ideal resource $\mathcal{S}$ is both an ε_1-remote state preparation with respect to some $\mathcal{A}$ and $\mathcal{Q}$ and ε_2-classically-realizable (including against only polynomially bounded distinguishers), then it is $(\varepsilon_1 + 2\varepsilon_2)$-describable with respect to $\mathcal{A}$. In particular, if $\varepsilon_1 = \mathsf{negl}(n)$ and $\varepsilon_2 = \mathsf{negl}(n)$, then $\mathcal{S}$ is describable within a negligible error $\varepsilon_1 + 2\varepsilon_2 = \mathsf{negl}(n)$.*

Proof. Let $\mathcal{S}$ be an ε_1-remote state preparation resource with respect to $(\mathcal{A}, \mathcal{Q})$ which is ε_2-classically-realizable. Then there exist π_A, π_B, σ, such that:

$$\mathop{\mathbb{E}}_{([\rho],\rho') \leftarrow \mathcal{A}\mathcal{S} \vdash \mathcal{Q}} [\operatorname{Tr}(\rho\rho')] \geq 1 - \varepsilon_1 \tag{12}$$

$$\pi_A \mathcal{C} \pi_B \approx_{\varepsilon_2} \mathcal{S} \vdash \tag{13}$$

and

$$\pi_A \mathcal{C} \approx_{\varepsilon_2} \mathcal{S}\sigma \tag{14}$$

Now, using (13), we get:

$$\mathcal{A}\pi_A \mathcal{C} \pi_B \mathcal{Q} \approx_{\varepsilon_2} \mathcal{A}\mathcal{S} \vdash \mathcal{Q} \tag{15}$$

So it means that we can't distinguish between $\mathcal{A}\mathcal{S} \vdash \mathcal{Q}$ and $\mathcal{A}\pi_A\mathcal{C}\pi_B\mathcal{Q}$ with an advantage better than ε_2 (i.e. with probability better than $\frac{1}{2}(1+\varepsilon_2)$). But, if we construct the following distinguisher, that runs $([\rho], \rho') \leftarrow \mathcal{A}\mathcal{S} \vdash \mathcal{Q}$, and then measures ρ' using the POVM $\{E_0, E_1\}$ (possible because this POVM is assumed to be efficiently implementable by distinguishers in $\mathcal{D}$), with $E_0 = [\rho]$ and $E_1 = I - [\rho]$ (which is possible because we know the classical description of ρ, which is positive and smaller than I, even when $[\rho] = 0$), we will measure E_0 with probability $1 - \varepsilon_1$. So it means that by replacing $\mathcal{A}\mathcal{S} \vdash \mathcal{Q}$ with $\mathcal{A}\pi_A\mathcal{C}\pi_B\mathcal{Q}$, the overall probability of measuring E_0 needs to be close to $1 - \varepsilon_1$. More precisely, we need to have:

$$\mathop{\mathbb{E}}_{([\rho],\rho') \leftarrow \mathcal{A}\pi_A\mathcal{C}\pi_B\mathcal{Q}} [\operatorname{Tr}(\rho\rho')] \geq 1 - \varepsilon_1 - \varepsilon_2 \tag{16}$$

Indeed, if the above probability is smaller than $1 - \varepsilon_1 - \varepsilon_2$, then we can define a distinguisher that outputs 0 if he measures E_0, and 1 if he measures E_1, and his probability of distinguishing the two distributions would be equal to:

$$\frac{1}{2} \mathop{\mathbb{E}}_{([\rho],\rho') \leftarrow \mathcal{A}\mathcal{S} \vdash \mathcal{Q}} [\operatorname{Tr}(\rho\rho')] + \frac{1}{2} \mathop{\mathbb{E}}_{([\rho],\rho') \leftarrow \mathcal{A}\pi_A\mathcal{C}\pi_B\mathcal{Q}} [\operatorname{Tr}((I-\rho)\rho')] \tag{17}$$

$$> \frac{1}{2}\left((1-\varepsilon_1) + 1 - (1 - \varepsilon_1 - \varepsilon_2)\right) \tag{18}$$

$$= \frac{1}{2}(1+\varepsilon_2) \tag{19}$$

So this distinguisher would have an advantage greater than ε_2, which is in contradiction with Eq. (15).

Using a similar argument and Eq. (13), we have:

$$\mathop{\mathbb{E}}_{([\rho],\rho')\leftarrow\mathcal{A}\mathcal{S}\sigma\pi_B\mathcal{Q}}[\,\mathrm{Tr}(\rho\rho')\,]\geq 1-\varepsilon_1-2\varepsilon_2 \tag{20}$$

We will now use $\pi_B\mathcal{Q}$ to construct a $\mathcal{B}$ that can describe the state given by the ideal resource. To do that, because $\pi_B\mathcal{Q}$ interacts only classically with the inner interface and outputs a single quantum state on the outer interface, then according to Lemma 3.5, $\pi_B\mathcal{Q}$ is reproducible, i.e. there exists[16] $\mathcal{B}$ such that $\mathcal{C}\pi_B\mathcal{Q}\approx_0\mathcal{C}\mathcal{B}\mathcal{T}$. Therefore[17], we have:

$$\mathop{\mathbb{E}}_{([\rho],\rho')\leftarrow\mathcal{A}\mathcal{S}\sigma\mathcal{B}\mathcal{T}}[\,\mathrm{Tr}(\rho\rho')\,]\geq 1-\varepsilon_1-2\varepsilon_2 \tag{21}$$

But because $\mathcal{T}$ simply converts the classical description $[\rho']$ into ρ', we also have:

$$\mathop{\mathbb{E}}_{([\rho],[\rho'])\leftarrow\mathcal{A}\mathcal{S}\sigma\mathcal{B}}[\,\mathrm{Tr}(\rho\rho')\,]\geq 1-\varepsilon_1-2\varepsilon_2 \tag{22}$$

After defining $\mathcal{P}=\sigma\mathcal{B}$, we have that $\mathcal{S}$ is $(\varepsilon_1+2\varepsilon_2)$-describable, which ends the proof. □

3.3 RSP Resources Impossible to Realize Classically

In the last section we proved that if an RSP functionality is classically-realizable (secure against polynomial quantum distinguishers), then this resource is describable by an unbounded adversary having access to the right interface of that resource.

Our main result in the previous section directly implies that as soon as there exists *no unbounded* adversary that, given access to the right interface, can find the classical description given on the left interface, then the RSP resource is *impossible* to classically realize (against *bounded* BQP distinguishers). Very importantly, this no-go result shows that the *only* type of RSP resources that can be classically realized are the ones that *leak* on the right interface enough information to allow an (possibly unbounded) adversary to determine the classical description given on the left interface. From a security point of view, this property is highly non-desirable, as the resource must leak the *secret description* of the state at least in *some representation*.

In this section we present some of these RSP resources that are impossible to classically realize.

Definition 3.7 (Ideal Resource $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$). *$\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ is the verifiable* RSP *resource (*RSP *which does not allow any deviation from the server), that receives no input, that internally picks a random $\theta\leftarrow\mathbb{Z}_{\frac{\pi}{2}}$, and that sends θ on the left interface, and $|+_\theta\rangle$ on the right interface as shown in Fig. 5.*

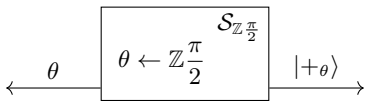


Fig. 5: Ideal resource $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$

[16] Note that here $\mathcal{B}$ is not efficient anymore, so that's why in the describable definition we don't put any bound on $\mathcal{B}$, but of course the proof does apply when the distinguisher is polynomially bounded.

[17] Indeed, we also have in particular $\mathcal{A}\mathcal{S}\sigma\mathcal{C}\pi_B\mathcal{Q}\approx_0\mathcal{A}\mathcal{S}\sigma\mathcal{C}\mathcal{B}\mathcal{T}$, and because $\mathcal{C}$ is a neutral resource [MR11, Sec. C.2] we can remove $\mathcal{C}$.

Lemma 3.8. *There exists a universal constant $\eta > 0$, such that for all $0 \leq \varepsilon < \eta$ the resource $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ is not ε-classically-realizable.*

Proof. This proof is at its core a direct consequence of quantum no-cloning: If we define $\mathcal{A}(\theta) := [|+_\theta\rangle\langle+_\theta|]$ ($\mathcal{A}$ just converts θ into its classical density matrix representation) and $\mathcal{Q}$ the trivial converter that just forwards any message, then $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ is a 0-remote state preparation resource with respect to $\mathcal{A}$ and $\mathcal{Q}$ because:

$$\mathop{\mathbb{E}}_{([\rho],\rho') \leftarrow \mathcal{A}\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}} \vdash \mathcal{Q}} [\operatorname{Tr}(\rho\rho')] = \frac{1}{4} \sum_{\theta \in \mathbb{Z}_{\frac{\pi}{2}}} \operatorname{Tr}(|+_\theta\rangle\langle+_\theta||+_\theta\rangle\langle+_\theta|) = 1 \geq 1 - 0 \tag{23}$$

Then, we remark also that there exists a constant $\eta > 0$ such that for all $\delta < \eta$, $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ is not δ-describable with respect to $\mathcal{A}$.

Indeed, it is first easy to see that $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ is not 0-describable with respect to $\mathcal{A}$. Indeed, we can assume by contradiction that there exists $\mathcal{P}$ such that:

$$\mathop{\mathbb{E}}_{([\rho],[\rho']) \leftarrow \mathcal{A}\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}} \mathcal{P}} [\operatorname{Tr}(\rho\rho')] = 1 \tag{24}$$

Then, because $\rho = |+_\theta\rangle\langle+_\theta|$ is a pure state, $\operatorname{Tr}(\rho\rho')$ corresponds to the fidelity of ρ and ρ', so $\operatorname{Tr}(\rho\rho') = 1 \Leftrightarrow \rho = \rho'$. But this is impossible because $\mathcal{P}$ just has a quantum state ρ as input, and if he can completely describe this quantum state then he can actually clone perfectly the input state with probability 1. But because the different possible values of ρ are not orthogonal, this is impossible due to the no-cloning theorem.

Moreover, it is also not possible to find a sequence $(\mathcal{P}^{(n)})_{n \in N}$ of CPTP maps that produces two copies of ρ with a fidelity arbitrary close to 1 (when $n \to \infty$), because CPTP maps are compact and the fidelity is continuous.

Therefore, there exists a constant $\eta > 0$,[18] such that:

$$\mathop{\mathbb{E}}_{([\rho],[\rho']) \leftarrow \mathcal{A}\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}} \mathcal{P}} [\operatorname{Tr}(\rho\rho')] < 1 - \eta \tag{25}$$

Now, by contradiction, we assume that $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ is ε-classically-realizable. Because $\lim_{n\to\infty} \varepsilon(n) = 0$, there exists $N \in \mathbb{N}$ such that $\varepsilon(N) < \eta/2$. So, using Theorem 3.6, $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ is $2\varepsilon(N)$-describable with respect to $\mathcal{A}$, which contradicts $2\varepsilon(N) < \eta$. □

Next, we describe RSP_V, a variant of $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$ introduced in [GV19]. In the latter, RSP_V, the adversary can make the resource abort, that the set of output states is bigger, and that the client can partially choose the basis of the output state. Similar to the $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$, we prove that classically-realizable RSP_V is not possible. Before going into the details of the no-go result, we formalize the ideal resource for a verifiable remote state preparation, RSP_V, below.

Definition 3.9 (Ideal Resource RSP_V, See [GV19]). *The ideal verifiable remote state preparation resource, RSP_V, takes an input $W \in \{X, Z\}$ on the left interface, but no honest input on the right interface. The right interface has a filtered functionality that corresponds to a bit $c \in \{0, 1\}$. When $c = 1$, RSP_V outputs error message ERR on both the interfaces, otherwise:*

1. *if $W = Z$ the resource picks a random bit b and outputs $b \in \mathbb{Z}_2$ to the left interface and a computational basis state $|b\rangle\langle b|$ to the right interface;*

[18] Note that for finding a more precise bound for η, it is possible to use Semidefinite Programming (SDP), or the method presented in [KRK12, p. 2]. However in our case it is enough to say that $\varepsilon > 0$ as we are interested only in asymptotic security.

2. *if $W = X$ the resource picks a random angle $\theta \in \mathbb{Z}_{\frac{\pi}{4}}$ and outputs θ to the left interface and a quantum state $|+_\theta\rangle\langle+_\theta|$ to the right interface.*

Corollary 3.10. *There exists a universal constant $\eta > 0$, such that for all $0 \leq \varepsilon < \eta$ the resource RSP_V is not ε-classically-realizable.*

Proof. The proof is quite similar to the proof of impossibility of $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$. The main difference is that we need to address properly the abort case when $c = 1$. The main idea is to define $\mathcal{A}$ a bit differently: $\mathcal{A}$ picks always $W = X$, and outputs as ρ the classical density matrix corresponding to s when $s \neq \text{ERR}$, and when $s = \text{ERR}$, $\mathcal{A}$ outputs the null matrix $\rho = 0$ ($\mathcal{Q}$ is still the trivial converter). It is easy to see again that this resource is a 0-remote state preparation resource, and it is also impossible to describe it with arbitrary small probability: indeed, when $c = 1$, $\rho = 0$, so the trace $\text{Tr}(\rho\rho')$ (that appears in Eq. (6)) is equal to 0. Therefore, from a converter $\mathcal{P}$ that (sometimes) inputs $c = 1$, we can always increase the value of $\text{Tr}(\rho\rho')$ by creating a new converter $\mathcal{P}'$ turning c into 0. And we are basically back to the same picture as $\mathcal{S}_{\mathbb{Z}_{\frac{\pi}{2}}}$, where we have a set of states that is impossible to clone with arbitrary small probability, which finishes the impossibility proof. □

Remark 3.11. Note that our impossibility of classically-realizing RSP_V does not contradict the result of [GV19]. Specifically, in their work they make use of an additional assumption (the so called "Measurement Buffer" resource), which "externalizes" the measurement done by the distinguisher onto the simulator. In practice, this allows the simulator to change the state on the distinguisher side without letting him know. However, what our result shows is that it is impossible to realize this Measurement Buffer resource with a protocol interacting purely classically. Intuitively, the Measurement Buffer re-creates a quantum channel between the simulator and the server: when the simulator is not testing that the server is honest, the simulator replaces the state of the server with the quantum state sent by the ideal resource. This method has however a second drawback: it is possible for the server to put a known state as the input of the Measurement Buffer, and if he is not tested on that run (occurring with probability $\frac{1}{n}$), then he can check that the state has not been changed, leading to polynomial security (a polynomially bounded distinguisher can distinguish between the ideal and the real world). And because in CC, the security of the whole protocol is the sum of the security of the inner protocols, any protocol using this inner protocol will not be secure against polynomial distinguishers.

3.4 Accepting the Limitations: Fully Leaky RSP resources

As explained in the previous section, Theorem 3.6 rules out all resources that are impossible to be *describable* with unbounded power, and that the only type of classically-realizable RSP resources would be the one leaking the full classical description of the output quantum state to an unbounded adversary, which we will refer to as being *fully-leaky* RSP. Fully-leaky RSP resources can be separated into two categories:

1. If the RSP is describable in quantum polynomial time, then the adversary can get the secret in polynomial time. This is obviously not an interesting case as the useful properties that we know from quantum computations (such as UBQC) cannot be preserved if such a resource is employed to prepare the quantum states.
2. If the RSP are only describable using unbounded power, then these *fully-leaky* RSP resources are not trivially insecure, but their universally composable security remains unclear. Indeed, it defeats the purpose of aiming at a nice ideal resource where the provided security should be clear "by definition" and it becomes hard to quantify how the additional leakage could be used when composed with other protocols. A

possible remedy would be to show restricted composition following [JM17] which we discuss at the end of this paragraph.

For completeness, we present an example of a resource that stands in this second category when assuming that post-quantum encryption schemes exist (e.g. based on the hardness of the LWE problem). As explained before, this resource needs to completely leak the description of the classical state, which in our case, is done by leaking an encryption of the description of the output state. The security guarantees therefore rely on the properties of the encryption scheme, and not on an ideal privacy guarantee as one would wish for, which is an obvious limitation.

A concrete example. In this section we focus on the second category of fully-leaky RSP and we show an example of resource that belongs to this class and a protocol realizing this resource. The fully-leaky RSP resource that we will implement, produces a BB84 state (corresponding to the set of states produced by the simpler QFactory protocol) and is described below:

Definition 3.12 (Ideal Resource $\mathsf{RSP}_{\mathsf{CC}}^{4-\mathsf{states},\mathcal{F}}$). *Let $\mathcal{F} = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ be a family of public-key encryption functions. Then, we define $\mathsf{RSP}_{\mathsf{CC}}^{4-\mathsf{states},\mathcal{F}}$ as pictured in Fig. 6. B_1 represents the basis of the output state, and is guaranteed to be random even if the right interface is malicious. B_2 represents the value bit of the output state when encoded in the basis B_1, and in the worst case it can be chosen by the right interface in a malicious scenario[19]. Note however that in a malicious run, the adversary does not have access (at least not directly from the ideal resource) to the quantum state whose classical description is known by the classical client.*

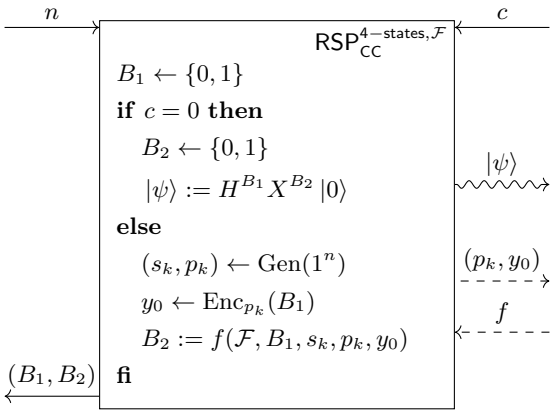


Fig. 6: Ideal resource $\mathsf{RSP}_{\mathsf{CC}}^{4-\mathsf{states},\mathcal{F}}$, which prepares atmone of the four BB84 states. The "snake" arrow is sent only in the honest case ($c = 0$), and the dashed arrows are send/received only in the malicious case ($c = 1$).

Lemma 3.13. *The 4-states QFactory protocol [CCKW19] (Protocol 2) securely constructs $\mathsf{RSP}_{\mathsf{CC}}^{4-\mathsf{states},\mathcal{F}}$ from a classical channel, where $\mathcal{F}$ is defined as follows:*

1. *$(t_K, K) \leftarrow \mathrm{Gen}(1^n)$ outputs two matrices: public K (used to describe the function) and secret t_K (a trapdoor used to invert the function) as defined in [CCKW19,CCKW18] (which is itself based on the learning with errors problem and the construction presented in [MP12]);*

[19] Note that here the right interface can have (in a malicious scenario) full control over B_2, but in the QFactory Protocol 2 it is not clear what an adversary can do concerning B_2.

2. $y_0 \leftarrow \mathrm{Enc}_K(B_1)$, *where* $y_0 = Ks_0 + e_0 + B_1 \left(q/2\ 0 \ldots 0\right)^T$, s_0 *and* e_0 *being sampled accordingly to some distribution presented in* [CCKW19,CCKW18]
3. $B_1 \leftarrow \mathrm{Dec}_{t_K}(y)$ - *using* t_K *we can efficiently obtain* B_1 *from* y_0.

Proof. We already know that the protocol of QFactory (π_A, π_B) is correct with super-polynomial probability if the parameters are chosen accordingly (Theorem B.1), therefore

$$\pi_A \mathcal{C} \pi_B \approx_\varepsilon \mathsf{RSP}_{\mathsf{CC}}^{4-\mathsf{states},\mathcal{F}} \vdash \tag{26}$$

for some negligible ε. We now need to find a simulator σ such that

$$\pi_A \mathcal{C} \approx_{\varepsilon'} \mathsf{RSP}_{\mathsf{CC}}^{4\text{-states},\mathcal{F}} \sigma \tag{27}$$

The simulator is trivial here: it sends $c = 1$ to ideal resource then, it just forwards the (K, y_0) given by the resource to its outer interface, and when it receives the (y, b) corresponding to the measurements performed by the server, it just sets the deviation f to be the same function as the one computed by π_A. Therefore, $\pi_A \mathcal{C} \approx_0 \mathsf{RSP}_{\mathsf{CC}}^{4\text{-states},\mathcal{F}} \sigma$, which ends the proof. □

Concluding remarks. We see that using this kind of leaky resource is not desirable: the resources are non-standard and it seems hard to write a modular protocol with this resource as an assumed resource. The resource is very specific and mimics its implementation. As such, we cannot really judge its security.

On the other hand however, if a higher-level protocol did guarantee that the value B_2 always remains hidden, i.e., a higher level protocol's output does not depend on on B_2 (e.g., by blinding it all the time), it is easy to see that we could simulate y_0 without knowledge about B_1 thanks to the semantic security of the encryption scheme. If we fix this restricted context, the ideal resource in Fig. 6 could be re-designed to not produce the output (p_k, y_0) at all and therefore, by definition, leak nothing extra about the quantum state (note that in such a restricted context, the simulator can simply come up with a fake encryption that is indistinguishable). This can be made formal following [JM17]. We note in passing that this particular example quite severely restricts applicability unfortunately. Indeed, it is interesting future research whether it is possible to come up with restricted yet useful contexts that admit nice ideal resources for RSP following the framework in [JM17].

4 Impossibility of Composable Classical-Client UBQC

In the previous section, we showed that it was impossible to get a (useful) composable $\mathsf{RSP}_{\mathsf{CC}}$ protocol. A (weaker) RSP protocol, however, could still be used internally in other protocols, hoping for the overall protocol to be composably secure. To this end, we analyze the composable security of a well-known delegated quantum computing protocol, universal blind quantum computation (UBQC), proposed in [BFK09]. The UBQC protocol allows a semi-quantum client, Alice, to delegate an arbitrary quantum computation to a (universal) quantum server Bob, in such a way that her input, the quantum computation and the output of the computation are information-theoretically hidden from Bob. The protocol requires Alice to be able to prepare single qubits of the form $|+_\theta\rangle$, where $\theta \in \mathbb{Z}\frac{\pi}{4}$ and send these states to Bob at the beginning of the protocol, the rest of the communication between the two parties being classical. We define the family of protocols $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$ as the RSP protocols that classically delegate the preparation of an output state $|+_\theta\rangle$, where $\theta \in \mathbb{Z}\frac{\pi}{4}$. That is, without loss of generality, we assume a pair of converters P_A, P_B such that the resource $R := P_A \mathcal{C} P_B$ has the behavior of the prototype RSP resource except with negligible probability. Put differently, we assume we

have an (except with negligible error) *correct* RSP protocol, but we make *no assumption about the security* of this protocol. Therefore, one can directly instantiate the quantum interaction with the $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$ at the first step as shown in Protocol 1. While UBQC allows for both quantum and classical outputs and inputs, given that we want to remove the quantum interaction in favor of a completely classical interaction, we only focus on the classical input and classical output functionality of UBQC in the remaining of the paper.

Protocol 1 UBQC with $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$ (See [BFK09])

– **Client's classical input:** An n-qubit unitary U that is represented as set of angles $\{\phi\}_{i,j}$ of a one-way quantum computation over a brickwork state/cluster state [MDF17], of the size $n \times m$, along with the dependencies X and Z obtained via flow construction [DK06].
– **Client's classical output:** The measurement outcome $\bar{s}$ corresponding to the n-qubit quantum state, where $\bar{s} = \langle 0| U |0\rangle$.

1. Client and Server runs $n \times m$ different instances of $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$ (in parallel) to obtain $\theta_{i,j}$ on client's side and $|+_{\theta_{i,j}}\rangle$ on server's side, where $\theta_{i,j} \leftarrow \mathbb{Z}\frac{\pi}{4}$, $i \in \{1, \cdots, n\}$, $j \in \{1, \cdots, m\}$
2. Server entangles all the qubits, $n \times (m-1)$ received from $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$, by applying controlled-Z gates between them in order to create a graph state $\mathcal{G}_{n \times m}$
3. For $j \in [1, m]$ and $i \in [1, n]$
   (a) Client computes $\delta_{i,j} = \phi'_{i,j} + \theta_{i,j} + r_{i,j}\pi$, $r_{i,j} \leftarrow \{0, 1\}$, where $\phi'_{i,j} = (-1)^{s^X_{i,j}} \phi_{i,j} + s^Z_{i,j}\pi$ and $s^X_{i,j}$ and $s^Z_{i,j}$ are computed using the previous measurement outcomes and the X and Z dependency sets. Client then sends the measurement angle $\delta_{i,j}$ to the Server.
   (b) Server measures the qubit $|+_{\theta_{i,j}}\rangle$ in the basis $\{|+_{\delta_{i,j}}\rangle, |-_{\delta_{i,j}}\rangle\}$ and obtains a measurement outcome $s_{i,j} \in \{0, 1\}$. Server sends the measurement result to the client.
   (c) Client computes $\bar{s}_{i,j} = s_{i,j} \oplus r_{i,j}$.
4. The measurement outcome corresponding to the last layer of the graph state ($j = m$) is the outcome of the computation.

Note that Protocol 1 is based on measurement-based model of quantum computing (MBQC). This model is known to be equivalent to the quantum circuit (up to polynomial overhead in resources) and does not require one to perform quantum gates on their side to realize arbitrary quantum computation. Instead, the computation is performed by an (adaptive) sequence of single-qubit projective measurements that steer the information flow across a highly entangled resource state. Intuitively, UBQC can be seen as a distributed MBQC where the measurements are performed by the server whereas the classical update of measurement bases is perfomed by the client. Since the projective measurements in quantum physics, in general, are probabilistic in nature and therefore, the client needs to update the measurement bases (and classically inform the server about the update) based on the outcomes of the earlier measurements to ensure the correctness of the computation. Roughly speaking, this information flow is captured by the X and Z dependencies. For more details, we refer the reader to [RB01,Nie06].

Next, we show that the Universal Blind Quantum Computing protocol [BFK09], which is proven to be secure in the Constructive Cryptography framework [DFPR14], cannot be proven composably secure (for the same ideal resource) when the quantum interaction is replaced with $\mathsf{RSP}_{\mathsf{CC}}$ (this class of protocol is denoted as $\mathsf{UBQC}_{\mathsf{CC}}$). We also give an outlook that the impossibility proof also rules out weaker ideal resources.

4.1 Impossibility of Composable $\mathsf{UBQC_{CC}}$ on 1 Qubit

In order to prove that there exists no $\mathsf{UBQC_{CC}}$ protocol, we will first focus on the simpler case when the computation is described by a single measurement angle. The resource that performs a blind quantum computation on one qubit ($\mathcal{S}_{UBQC1}$) is defined as below:

Definition 4.1 (Ideal resource of single-qubit UBQC (See [DFPR14])). *The definition of the ideal resource $\mathcal{S}_{UBQC1}$, depicted in Fig. 7, achieves blind quantum computation specified by a single angle ϕ. The input (ξ, ρ) is filtered when $c = 0$. The ξ can be any deviation (specified for example using the classical description of a CPTP map) that outputs a classical bit, and which can depend on the computation angle ϕ and on some arbitrary quantum state ρ.*

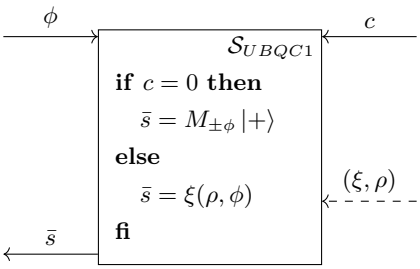


Fig. 7: Ideal resource $\mathcal{S}_{UBQC1}$ for UBQC with one angle, with a filtered (dashed) input. In the case of honest server the output $\bar{s} \in \{0, 1\}$ is computed by measuring the qubits $|+\rangle$ in the $\{|+_\phi\rangle, |-_\phi\rangle\}$ basis. On the other hand if $c = 1$ any malicious behaviour of server can be captured by (ξ, ρ), i.e. the output $\bar{s}$ is computed by applying the CPTP map ξ on the input ϕ and on another auxiliary state ρ chosen by the server.

Theorem 4.2 (No-go composable classical-client single-qubit UBQC). *Let (P_A, P_B) be a protocol interacting only through a classical channel $\mathcal{C}$, such that $(\theta, \rho_B) \leftarrow (P_A \mathcal{C} P_B)$ with $\theta \in \mathbb{Z}_{\frac{\pi}{4}}$, and such that (by correctness) the trace distance between ρ_B and $|+_\theta\rangle\langle+_\theta|$ is negligible with overwhelming probability[20] with overwhelming probability[21]. Then, if we define π_A and π_B as the UBQC protocol on one qubit that makes use of (P_A, P_B) as a sub-protocol to replace the quantum channel (as pictured in Fig. 8), (π_A, π_B) is not composable, i.e. there exists no simulator σ such that:*

$$\pi_A \mathcal{C} \pi_B \approx_\varepsilon \mathcal{S}_{UBQC1} \vdash^{c=0} \tag{28}$$

$$\pi_A \mathcal{C} \approx_\varepsilon \mathcal{S}_{UBQC1} \sigma \tag{29}$$

for some negligible $\varepsilon = \mathsf{negl}(n)$.

Proof. In order to prove this theorem, we will proceed by contradiction. Let us assume that there exists (P_A, P_B), and a simulator σ having the above properties.
Then, for the same resource $\mathcal{S}_{UBQC1}$ we consider a different protocol $\boldsymbol{\pi}' = (\pi'_A, \pi'_B)$ that

[20] In the following, the parties P_A and P_B (and therefore π_A and π_B) and the simulator σ depend on some security parameter n, but, in order to simplify the notations and the proof, this dependence will be implicit. We are as usual interested only in the asymptotic security, when $n \to \infty$.

[21] Note that here ρ_B is different at every run: it corresponds to the density matrix of the state obtained after running P_B, when tracing out the environment and the internal registers of P_B and P_A.

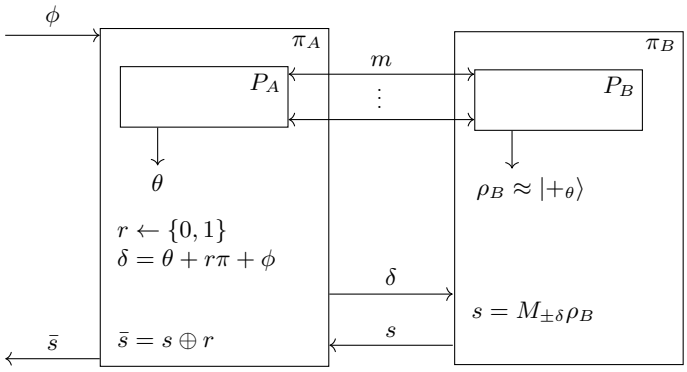


Fig. 8: UBQC with one qubit when both Alice and Bob follows the protocol honestly (see Protocol 1)

realizes it, but using a different filter[22] $\vdash^\sigma$ and a different simulator σ':

$$\pi'_A \mathcal{C} \pi'_B \approx_\varepsilon \mathcal{S}_{UBQC1} \vdash^\sigma \tag{30}$$

$$\pi'_A \mathcal{C} \approx_\varepsilon \mathcal{S}_{UBQC1} \sigma' \tag{31}$$

More specifically, the new filter $\vdash^\sigma_{UBQC1}$ will depend on σ defined in Eq. (29). Then our main proof can be described in the following steps:

1. We first show in Lemma 4.4 that $\mathcal{S}_{UBQC1}$ is also ε-classically-realizable by (π'_A, π'_B) with the filter $\vdash^\sigma$.
2. We then prove in Lemma 4.5 that the resource $\mathcal{S}_{UBQC1}$ is an RSP within $\mathsf{negl}(n)$, with respect to some well chosen converters $\mathcal{A}$ and $\mathcal{Q}$ (see Fig. 9) and this new filter $\vdash^\sigma$.
3. Then, we use the main result about RSP (Theorem 3.6) to show that $\mathcal{S}_{UBQC1}$ is describable within $\mathsf{negl}(n)$ with respect to $\mathcal{A}$ (Corollary 4.6).
4. Finally, in Lemma 4.8 we prove that if $\mathcal{S}_{UBQC1}$ is describable then we could achieve *superluminal signaling*, which concludes the contradiction proof.

□

Definition 4.3. *Let $\pi' = (\pi'_A, \pi'_B)$ the protocol realizing $\mathcal{S}_{UBQC1}$ described in the following way (as pictured Fig. 9):*

- *$\pi'_A = \pi_A$ (Fig. 8)*
- *π'_B: runs P_B, obtains a state ρ_B, then uses the angle δ received from its inner interface to compute $\tilde{\rho} := R_Z(-\delta)\rho_B$, and finally outputs $\tilde{\rho}$ on its outer interface and $s := 0$ on its inner interface.*

Then we define $\vdash^\sigma = \sigma\pi'_B$ depicted in Fig. 10 (with σ the simulator defined in Eq. (29) as explained before).
We define the converters $\mathcal{A}$ and $\mathcal{Q}$ as seen in:

Lemma 4.4. *If $\mathcal{S}_{UBQC1}$ is ε-classically-realizable by (π_A, π_B) with the filter $\vdash^{c=0}$ then $\mathcal{S}_{UBQC1}$ is also ε-classically-realizable by (π'_A, π'_B) with the filter $\vdash^\sigma$.*

Proof. If S_{UBQC1} is ε-classically-realizable with $\vdash^{c=0}$, then as seen in Theorem 4.2, we have:

$$\pi_A \mathcal{C} \pi_B \approx_\varepsilon \mathcal{S}_{UBQC1} \vdash^{c=0} \tag{32}$$

$$\pi_A \mathcal{C} \approx_\varepsilon \mathcal{S}_{UBQC1} \sigma \tag{33}$$

[22] Note that we could include this new filter inside $\mathcal{S}_{UBQC1}$ and use a more traditional filter $\vdash^{c=0}$ but for simplicity we will just use a different filter.

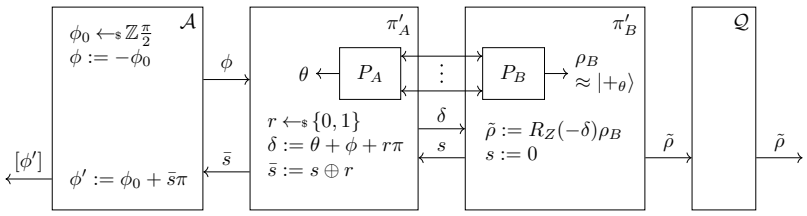


Fig. 9: Definition of $\mathcal{A}$, π'_A, π'_B and $\mathcal{Q}$.

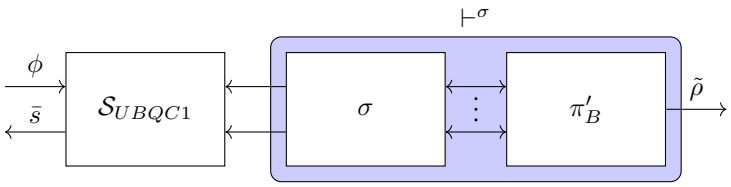


Fig. 10: Description of $\vdash^\sigma$

Now we can show that $\mathcal{S}_{UBQC1}$ is ε-classically-realizable by (π'_A, π'_B) with $\vdash^\sigma$, i.e. that there exists a simulator σ' such that:

$$\pi'_A \mathcal{C} \pi'_B \approx_\varepsilon \mathcal{S}_{UBQC1} \vdash^\sigma \tag{34}$$

$$\pi'_A \mathcal{C} \approx_\varepsilon \mathcal{S}_{UBQC1} \sigma' \tag{35}$$

For the correctness condition, we have:

$$\pi'_A \mathcal{C} \pi'_B = (\pi_A \mathcal{C}) \pi'_B \tag{36}$$

$$\approx_\varepsilon (\mathcal{S}_{UBQC1} \sigma) \pi'_B \tag{37}$$

$$= \mathcal{S}_{UBQC1} \vdash^\sigma \tag{38}$$

For the security condition, we define $\sigma' = \sigma$. Then, we have:

$$\pi'_A \mathcal{C} = \pi_A \mathcal{C} \tag{39}$$

$$\approx_\varepsilon \mathcal{S}_{UBQC1} \sigma \tag{40}$$

Which concludes our proof. □

Lemma 4.5. *If $\mathcal{S}_{UBQC1}$ is $\mathsf{negl}(n)$-classically-realizable with $\vdash^{c=0}$ then $\mathcal{S}_{UBQC1}$ is an $\mathsf{negl}(n)$-remote state preparation resource with respect the converters $\mathcal{A}$ and $\mathcal{Q}$ and filter $\vdash^\sigma$ defined in Fig. 9.*

Proof. We need to prove that:

$$\mathop{\mathbb{E}}_{([\rho],\rho_B) \leftarrow \mathcal{A}\mathcal{S}_{UBQC1} \vdash^\sigma \mathcal{Q}} [\,\mathrm{Tr}(\rho\rho_B)\,] \geq 1 - \varepsilon \tag{41}$$

First, we remark that due to Lemma 4.4:

$$\mathcal{A}\mathcal{S}_{UBQC1} \vdash^\sigma \mathcal{Q} \approx_\varepsilon \mathcal{A}\pi'_A \mathcal{C} \pi'_B \mathcal{Q} \tag{42}$$

However, from the protocol description it is easy to check that in the real world $\bar{s} = 0 \oplus r = r$, and therefore $\phi' := \phi_0 + \bar{s}\pi = \phi_0 + r\pi$ and $\rho = |+_{\phi'}\rangle\langle+_{\phi'}|$. And because the trace distance between ρ_B and $|+_\theta\rangle\langle+_\theta|$ is negligible with overwhelming probability (by the correctness of (P_A, P_B)), then we also have that $\tilde{\rho} = R_Z(-\delta)\rho_B R(-\delta)^\dagger$ is negligibly close in trace distance to $|+_{\theta-\delta}\rangle\langle+_{\theta-\delta}| = |+_{-\phi_0+r\pi}\rangle\langle+_{-\phi_0+r\pi}| = |+_{\phi'}\rangle\langle+_{\phi'}|$. Therefore, we have:

$$\mathop{\mathbb{E}}_{([\rho],\tilde{\rho}) \leftarrow \mathcal{A}\pi'_A \mathcal{C} \pi'_B \mathcal{Q}} [\,\mathrm{Tr}(\rho\tilde{\rho})\,] \geq 1 - \mathsf{negl}(n) \tag{43}$$

Then it also means that:

$$\mathop{\mathbb{E}}_{([\rho],\tilde{\rho})\leftarrow\mathcal{A}\mathcal{S}_{UBQC1}\vdash^{\sigma}\mathcal{Q}}[\,\mathrm{Tr}(\rho\tilde{\rho})\,]\geq 1-\mathsf{negl}(n) \tag{44}$$

otherwise we could (using a similar argument to the one given in the proof of Theorem 3.6) distinguish between the ideal and the real world, contradicting Eq. (42), which concludes the proof. □

Now, using our main Theorem 3.6 we obtain directly that if $\mathcal{S}_{UBQC1}$ is classically-realizable and RSP with respect to filter $\vdash^{\sigma}$, then it is also describable:

Corollary 4.6. *If $\mathcal{S}_{UBQC1}$ is $\mathsf{negl}(n)$-classically-realizable with respect to filter $\vdash^{c=0}$ then $\mathcal{S}_{UBQC1}$ is $\mathsf{negl}(n)$-describable with respect to the converter $\mathcal{A}$ described above.*

Lemma 4.7. *Let $\Omega=\{[\rho_i]\}$ be a set of (classical descriptions of) density matrices, such that $\forall i\neq j$, $\mathrm{Tr}(\rho_i\rho_j)\leq 1-\eta$. Then let $([\rho],[\tilde{\rho}])$ be two random variables (representing classical description of density matrices), such that $[\rho]\in\Omega$ and $\mathop{\mathbb{E}}_{([\rho],[\tilde{\rho}])}[\,\mathrm{Tr}(\rho\tilde{\rho})\,]\geq 1-\varepsilon$, with $\eta>6\sqrt{\varepsilon}$. Then, if we define the following "rounding" operation that rounds $\tilde{\rho}$ to the closest $\tilde{\rho}_r\in\Omega$:*

$$[\tilde{\rho}_r]:=\mathrm{Round}_{\Omega}([\tilde{\rho}]):=\mathop{\arg\max}_{[\tilde{\rho}_r]\in\Omega}\mathrm{Tr}(\tilde{\rho}_r\tilde{\rho}) \tag{45}$$

Then we have:

$$\Pr_{([\rho],[\tilde{\rho}])}[\,\mathrm{Round}_{\Omega}([\tilde{\rho}])=[\rho]\,]\geq 1-\sqrt{\varepsilon} \tag{46}$$

In particular, if $\varepsilon=\mathsf{negl}(n)$, and $\eta\neq 0$ is a constant, $\Pr[\,\mathrm{Round}_{\Omega}([\tilde{\rho}])=[\rho]\,]\geq 1-\mathsf{negl}(n)$.

Proof. We know that $\mathop{\mathbb{E}}_{([\rho],[\tilde{\rho}])}[\,\mathrm{Tr}(\rho\tilde{\rho})\,]\geq 1-\varepsilon$. Therefore, using Markov inequality we get that:

$$\Pr_{([\rho],[\tilde{\rho}])}[\,1-\mathrm{Tr}(\rho\tilde{\rho})\geq\sqrt{\varepsilon}\,]\leq\frac{\mathbb{E}[\,1-\mathrm{Tr}(\rho\tilde{\rho})\,]}{\varepsilon} \tag{47}$$

$$\Pr_{([\rho],[\tilde{\rho}])}[\,\mathrm{Tr}(\rho\tilde{\rho})\leq 1-\sqrt{\varepsilon}\,]\leq\frac{\varepsilon}{\sqrt{\varepsilon}} \tag{48}$$

$$\Pr_{([\rho],[\tilde{\rho}])}[\,\mathrm{Tr}(\rho\tilde{\rho})\geq 1-\sqrt{\varepsilon}\,]\geq 1-\sqrt{\varepsilon} \tag{49}$$

But when $\mathrm{Tr}(\rho\tilde{\rho})\geq 1-\sqrt{\varepsilon}$, we have $\mathrm{Round}_{\Omega}([\tilde{\rho}])=\rho$.

We will indeed show that $\forall\rho_i\in\Omega$, $\mathrm{Tr}(\rho_i\tilde{\rho})\leq\mathrm{Tr}(\rho\tilde{\rho})$. By contradiction, we assume there exists $\rho_i\in\Omega$ such that $\rho_i\neq\rho$ and $\mathrm{Tr}(\rho_i\tilde{\rho})>\mathrm{Tr}(\rho\tilde{\rho})\geq 1-\sqrt{\varepsilon}$. But due to Lemma C.4 we have:

$$\mathrm{Tr}(\rho_i\rho)\geq 1-3(\sqrt{\varepsilon}+\sqrt{\varepsilon})=1-6\sqrt{\varepsilon} \tag{50}$$

However, because both ρ_i and ρ belong to Ω, we also have $\mathrm{Tr}(\rho_i\rho)\leq 1-\eta<1-6\sqrt{\varepsilon}$, which is absurd.

Therefore, using Eq. (49) we have

$$\Pr_{([\rho],[\tilde{\rho}])}[\,\mathrm{Round}_{\Omega}([\tilde{\rho}])=[\rho]\,]\geq 1-\sqrt{\varepsilon} \tag{51}$$

which concludes the proof. □

Lemma 4.8. *$\mathcal{S}_{UBQC1}$ cannot be $\mathsf{negl}(n)$-describable with respect to converter $\mathcal{A}$.*

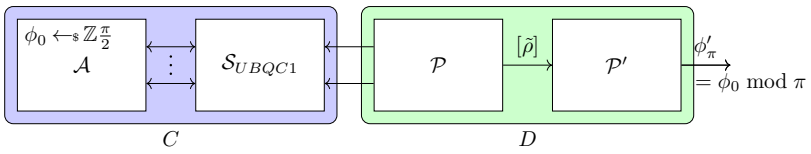


Fig. 11: Illustration of the no-signaling argument

Proof. If we assume that $\mathcal{S}_{UBQC1}$ is $\mathsf{negl}(n)$-describable, then there exists a converter $\mathcal{P}$ (outputting $[\tilde{\rho}]$) such that:

$$\mathop{\mathbb{E}}_{([\rho],[\tilde{\rho}])\leftarrow\mathcal{A}\mathcal{S}_{UBQC1}\mathcal{P}}[\,\mathrm{Tr}(\rho\tilde{\rho})\,] \geq 1-\mathsf{negl}(n) \tag{52}$$

We define the set $\Omega := \{[|+_{\theta'}\rangle\langle+_{\theta'}|] \mid \theta' \in \{0, \pi/4, ..., 7\pi/4\}\}$. For simplicity, we will denote in the following $[\theta] = [|+_\theta\rangle\langle+_\theta|]$.

In the remaining of the proof, we are going to use the converters $\mathcal{A}$ and $\mathcal{P}$ together with the ideal resource $\mathcal{S}_{UBQC1}$, to construct a 2-party setting that would achieve signaling, which would end our contradiction proof. More specifically, we will define a converter D running on the right interface of $\mathcal{S}_{UBQC1}$ which will manage to recover the ϕ_0 chosen randomly by $\mathcal{A}$.
As shown in Fig. 11, if we define C as $C := \mathcal{A}\mathcal{S}_{UBQC1}$ and D the converter described above, then the setting can be seen equivalently as: C chooses as random ϕ_0 and D needs to output $\phi_0 \bmod \pi$. This is however impossible, as no message is sent from $\mathcal{S}_{UBQC1}$ to its right interface (as seen in Fig. 11) (and thus no message from C to D), and therefore guessing ϕ_0 is forbidden by the no-signaling principle [GRW80].

We define $\mathcal{P}'$ as the converter that, given $[\tilde{\rho}]$ from the outer interface of $\mathcal{P}$ computes $[\tilde{\phi}] = \mathrm{Round}_\Omega([\tilde{\rho}])$ and outputs $\tilde{\phi}_\pi = \tilde{\phi} \bmod \pi$ (as depicted in Fig. 11). We will now prove that $\tilde{\phi}_\pi = \phi_0 \bmod \pi$ with overwhelming probability.

All elements in Ω are different pure states, and in finite number, so there exist a constant $\eta > 0$ respecting the first condition of Lemma 4.7. Moreover from Eq. (52) we have that $\mathcal{S}_{UBQC1}$ is ε-describable with $\varepsilon = \mathsf{negl}(n)$, so we also have (for large enough n), $\eta > 6\sqrt{\varepsilon}$. Therefore, from Lemma 4.7, we have that:

$$\Pr_{([\rho],[\tilde{\rho}])\leftarrow\mathcal{A}\mathcal{S}_{UBQC1}\mathcal{P}}[\,\mathrm{Round}_\Omega([\tilde{\rho}]) = [\rho]\,] \geq 1-\mathsf{negl}(n) \tag{53}$$

But using the definition of converter $\mathcal{A}$, we have: $[\rho] = [\phi']$, where $\phi' = \phi_0 + \bar{s}\pi$, and hence $\phi' \bmod \pi = \phi_0 \bmod \pi$. Then, using the definition of $\mathcal{P}'$, the Eq. (53) is equivalent to:

$$\Pr_{([\phi'],\tilde{\phi}_\pi)\leftarrow\mathcal{A}\mathcal{S}_{UBQC1}\mathcal{P}\mathcal{P}'}[\,\tilde{\phi}_\pi = \phi_0 \bmod \pi\,] \geq 1-\mathsf{negl}(n) \tag{54}$$

However, as pictured in Fig. 11, this can be seen as a game between $C = \mathcal{A}\mathcal{S}_{UBQC1}$ and $D = \mathcal{P}\mathcal{P}'$, where, as explained before, C picks a $\phi_0 \in \mathbb{Z}_{\frac{\pi}{2}}$ randomly, and D needs to output $\phi_0 \bmod \pi$. From Eq. (54) D wins with overwhelming probability, however, we know that since there is no information transfer from C to D, the probability of winning this game better than 1/4 (guessing both the bits at random) would imply signalling. □

Remark 4.9. The guessing game described at the end of the preceding proof can be generalized to the case when some (partial) information transfer from C to D takes place. More precisely, whenever we consider a new resource together with some converters $\mathcal{A}$ and $\mathcal{Q}$, it is enough to show that this resource is not describable to prove that it is impossible to classically realize. To that purpose, it may as above be practical to

define a guessing game similar to the above one, but without the nice property that no information flows from C to D. Here, the connections with the non-local games [BCP+14] and information causality [PPK+09] could provide an upper bound on the winning probability (e.g., as a function of the conditional mutual information conditioned on the information exchanged). We leave the quantitative analysis for future work.

4.2 Impossibility of Composable $\mathsf{UBQC}_{\mathsf{CC}}$ on Any Number of Qubits

We saw in Theorem 4.2 that it is not possible to implement a composable classical-client UBQC protocol performing a computation on a single qubit. In this section, we prove that this result generalizes to the impossibility of $\mathsf{UBQC}_{\mathsf{CC}}$ on computations using an arbitrary number of qubits. The proof works by reducing the general case to the single-qubit case from the previous section.

Theorem 4.10 (No-go Composable Classical-Client UBQC). *Let (P_A, P_B) be a protocol interacting only through a classical channel $\mathcal{C}$, such that $(\theta, \rho_B) \leftarrow (P_A \mathcal{C} P_B)$ with $\theta \in \mathbb{Z}\frac{\pi}{4}$, and such that the trace distance between ρ_B and $|+_\theta\rangle\langle+_\theta|$ is negligible with overwhelming probability. Then, if we define (π_A^G, π_B^G) as the UBQC protocol on any fixed graph G (with at least one output qubit[23]), that uses (P_A, P_B) as a sub-protocol to replace the quantum channel, (π_A^G, π_B^G) is not composable, i.e. there exists no simulator σ such that:*

$$\pi_A^G \mathcal{C} \pi_B^G \approx_\varepsilon \mathcal{S}_{UBQC1} \vdash^{c=0} \tag{55}$$

$$\pi_A^G \mathcal{C} \approx_\varepsilon \mathcal{S}_{UBQC1} \sigma \tag{56}$$

for some negligible $\varepsilon = \mathsf{negl}(n)$.

Proof. To prove this statement, we just need to prove that we can come back to the setting with a single qubit, where we want to perform a computation with angle ϕ, and output one angle close to ϕ as in the proof of Theorem 4.2. Because the graph has at least one output qubit, we will denote by ω the index of the last output qubit. So the idea is to let the distinguisher choose the client input such that for any node $i \neq \omega$ in the graph, $\phi_i = 0$, and for the output qubit, $\phi_\omega = \phi$. Moreover, on the server side, the distinguisher will behave like the honest protocol π_B^G, except that it will not entangle the qubits provided by P_A, and it will deviate on the output qubit ω by not measuring it and sending $s := 0$, the qubit being rotated again with angle $-\delta_\omega$, and outputed on the outer interface, like in the one-qubit case. It is now easy to see by induction (over the index of the qubit, following the order chosen on G) that, in the real world, for all $i \neq \omega$, we always have $s_i = r_i$, therefore $\bar{s_i} = 0$. So for all nodes i, (including ω), $s_i^X = \oplus_{i \in D_i} \bar{s}_i = 0$ and $s_i^Z = \oplus_{i \in D'_i} \bar{s}_i = 0$. Thus we have on the last node:

$$\begin{aligned} \delta_\omega &= \theta_\omega + (-1)^{s_\omega^X} \phi_\omega + s_\omega^Z \pi + r_\omega \pi \\ &= \theta_\omega + \phi + r_\omega \pi \end{aligned}$$

which corresponds exactly to the single-qubit setting, shown to be impossible. □

5 Game-Based Security of QF-UBQC

While we know from Theorem 4.10 that classical-client UBQC (henceforth simply $\mathsf{UBQC}_{\mathsf{CC}}$) cannot be proven secure in a fully composable setting, there is hope that it remains possible with a weaker definition of security. And indeed, in this section we show that

[23] Note, that in $\mathsf{UBQC}_{\mathsf{CC}}$ with zero output qubits the client does not receive any results. Hence, the protocol is trivially implementable for this degenerated case.

$\mathsf{UBQC}_{\mathsf{CC}}$ is possible in the *game-based setting* by implementing it using a combination of the known quantum-client UBQC Protocol 1 [BFK09] and 8-states QFactory Protocol 3 [CCKW19]. We start with giving a formal definition of the game-based security of $\mathsf{UBQC}_{\mathsf{CC}}$.

Definition 5.1 (Blindness of $\mathsf{UBQC}_{\mathsf{CC}}$). *A $\mathsf{UBQC}_{\mathsf{CC}}$ protocol $\mathcal{P} = (P_C, P_S)$ is said to be (computationally) adaptively blind if no computationally bounded malicious server can distinguish between runs of the protocol with adversarially chosen measurement patterns on the same MBQC graph.*

In formal terms, $\mathcal{P}$ is said to be (computationally) adaptively blind if and only if for any quantum-polynomial-time adversary A it holds that

$$\Pr\left[c' = c \;\middle|\; (\phi^{(1)}, \phi^{(2)}) \leftarrow A,\, c \leftarrow_{\$} \{0,1\},\, \left\langle P_C(\phi^{(c)}), A\right\rangle,\, c' \leftarrow A\right] \leq \frac{1}{2} + \mathrm{negl}(\lambda),$$

where λ is the security parameter, and $\left\langle P_C(\phi^{(c)}), A\right\rangle$ denotes the interaction of the two algorithms $P_C(\phi^{(c)})$ and A.

Remark 5.2. Although, Definition 5.1 is written using the terminology of measurement-based model. It doesn't compromise the generality, as the model is universal and can be easily translated into a circuit model, because the measurement pattern and unitary operator have a one-to-one mapping.

5.1 Implementing Classical-Client UBQC with QFactory

The UBQC protocol from [BFK09], where the quantum interaction is replaced by a $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$ protocol, is shown in Protocol 1. In this section, we replace the $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$ protocol with the concrete protocol proposed in [CCKW19]. This protocol, known by the name of 8-states QFactory[24] and described in Protocol 3, exactly emulates the capability of $\mathsf{RSP}_{\mathsf{CC}}^{8-\mathsf{states}}$. The resulting protocol contains a QFactory instance for each qubit that would have been generated on the client's side. The keys to all QFactory instances are generated entirely independently by the client.

Unfortunately, considering the results from Section 4 there is no hope that the composable security of any $\mathsf{UBQC}_{\mathsf{CC}}$ may be achieved. Nonetheless, letting go of composability, we are able to prove the game-based security for this specific combination of protocols. This leads us to the main theorem of this section.

Theorem 5.3 (Game-based Blindness of QF-UBQC). *The protocol resulting from combining the quantum-client UBQC protocol with QFactory is a (computationally) adaptively blind implementation of $\mathsf{UBQC}_{\mathsf{CC}}$ in the game-based model according to Definition 5.1. We call this protocol QF-UBQC.*

The proof of Theorem 5.3 which will be given in the remainder of this section follows two main ideas:

1. Every angle used in the UBQC protocol has only eight possible values, and can, therefore, be described by three bits. In the protocol, the first bit is the one for which QFactory *cannot* guarantee blindness. Fortunately, the additional one-time padding in UBQC allows analyzing the blindness of the protocol independently of the blindness of exactly this first bit. Therefore, it suffices to rely on the blindness of the last two bits which is conveniently guaranteed by QFactory and the hardness of LWE.

[24] We refer here to the 8-states QFactory implementation with negligible abort probability, and superpolynomial parameters. This is necessary since our proof does not take the abort case into account for now.

2. To analyze the leakage about the last two bits during a QFactory run, it is sufficient to notice that the leakage is equal to a ciphertext under an LWE-based encryption scheme. The semantic security of this encryption scheme and the hardness assumption for LWE guarantee that this leakage is negligible and can be omitted.

In more detail, the 8-states QFactory protocol which is used here consists of two combined runs of 4-states QFactory, each contributing with a single blind bit to the three-bit angles used in the UBQC protocol. Recall from Theorem B.2 and Theorem B.6 the formulae for how these angles from the 4-states protocol are combined in the 8-states protocol. If B_1 is the hidden bit of the first 4-states QFactory instance and B'_1 the hidden bit of the second instance, then we obtain

$$L_1 = B'_2 \oplus B_2 \oplus [B_1 \cdot (s_1 \oplus s_2)],\ L_2 = B'_1 \oplus [(B_2 \oplus s_2) \cdot B_1],\ L_3 = B_1, \tag{57}$$

where $L = L_1L_2L_3 \in \{0,1\}^3$ is the description of the output state $\left|+_{L\frac{\pi}{4}}\right\rangle$, s_1, s_2 are computed by the server, and

$$B_2 = f(\mathsf{sk}, B_1, y, b), \qquad B'_2 = f(\mathsf{sk}', B'_1, y', b') \tag{58}$$

for some function f, QFactory secret keys $\mathsf{sk}, \mathsf{sk}'$, and server-chosen values y, b, y', b'.

The two 4-states QFactory instances now leak the ciphertext of B_1 and B'_1, respectively. Given the semantic security of the encryption, after a run of 8-states QFactory, L_2 and L_3 remain hidden, while the blindness of L_1 cannot be guaranteed by QFactory. This fact is going to be useful in the following proof.

5.2 Single-Qubit QF-UBQC

We first prove the security of combining QFactory with UBQC on a single qubit.

Lemma 5.4 (Blindness in the single-qubit case). *The protocol resulting from combining the quantum-client* UBQC *protocol with (8-states) QFactory is a (computationally) adaptively blind implementation of* $\mathsf{UBQC}_{\mathsf{CC}}$ *in the game-based model for MBQC computations on a single qubit.*

Proof. We start with the real protocol, describing the adaptive blindness of QFactory combined with single-qubit UBQC. In the following, we denote the set of possible angles by $M = \{j\pi/4, j = 0, \ldots, 7\}$. The encryption scheme that appears in Game 1 is the semantically secure public-key encryption scheme from [Reg09]. Note that the two key pairs are generated completely independently on the challenger's side.

GAME 1:

	Adversary		Challenger
1 :	Choose $\phi^{(1)}, \phi^{(2)} \in M$	$\phi^{(1)}, \phi^{(2)} \longrightarrow$	$c \leftarrow\$ \{0,1\}$
2 :			$B_1, B'_1 \leftarrow\$ \{0,1\}$
3 :		$\mathsf{pk}, \mathsf{pk}',$ $\mathrm{Enc}^{\mathsf{pk}}(B_1), \mathrm{Enc}^{\mathsf{pk}'}(B'_1) \longleftarrow$	Generate key pairs $(\mathsf{sk}, \mathsf{pk}), (\mathsf{sk}', \mathsf{pk}')$
4 :		$y, b, y', b', s_1, s_2 \longrightarrow$	$B_2 = f(\mathsf{sk}, B_1, y, b),\ B'_2 = f(\mathsf{sk}', B'_1, y', b')$
5 :			$L_1 = B'_2 \oplus B_2 \oplus [B_1 \cdot (s_1 \oplus s_2)]$
6 :			$L_2 = B'_1 \oplus [(B_2 \oplus s_2) \cdot B_1]$
7 :			$L_3 = B_1$
8 :			$r \leftarrow\$ \{0,1\}$
9 :		$\delta \longleftarrow$	$\delta = \phi^{(c)} + L_3\pi/4 + L_2\pi/2 + L_1\pi + r\pi$
10 :		$s \longrightarrow$	
11 :	Compute guess $c' \in \{0,1\}$	$c' \longrightarrow$	Check $c' = c$?

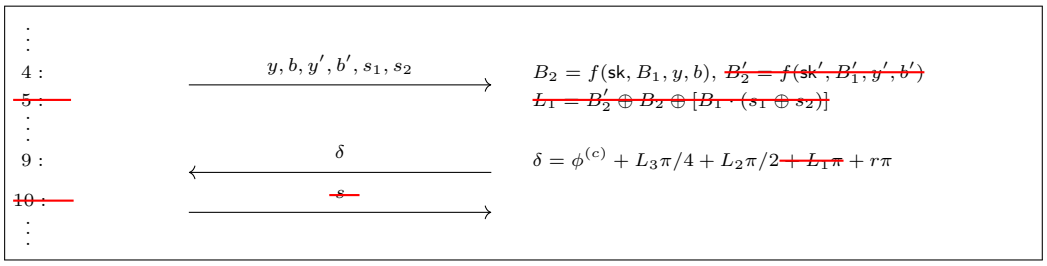
In the following, instead of repeating the redundant parts of subsequent games, we only present incremental modifications to Game 1. Every not explicitly written line is assumed to be identical to the previous game.

Clearly, since s is never used by the challenger, we can remove it from the protocol without distorting the success probability of the adversary. Next, we remove L_1 from the protocol and from the calculation of δ. L_1 is only used in the calculation of δ, which can be rewritten as

$$\delta = \phi^{(c)} + L_3\pi/4 + L_2\pi/2 + (L_1 + r)\pi. \tag{59}$$

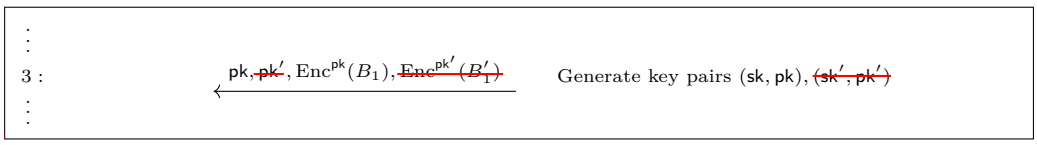
Since r is a uniform binary random variable with unique use in this line, $(L_1 + r)$ is still uniform over $\{0, 1\}$. Therefore, removing L_1 leaves the distribution of the protocol outcome unchanged.

GAME 2:



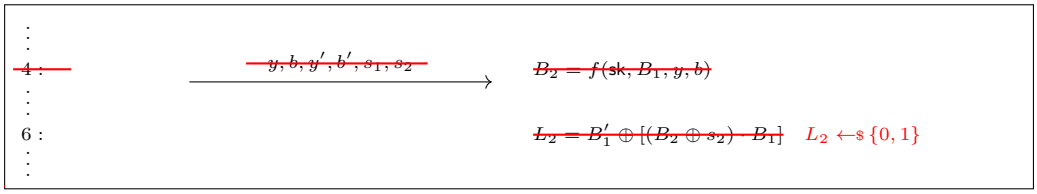
The next step introduces a (negligible) distortion to the success probability of the adversary. By the semantic security of the employed encryption scheme, no quantum-polynomial-time adversary can notice if the plaintext is replaced by pure randomness except with negligible probability, even if information about the original plaintext is leaked on the side. Therefore, replacing B_1' in the encryption by independent randomness cannot lead to a significant change of the adversary's success probability. Further, since ciphertexts of independent randomness can be equally generated by the adversary herself (being in possession of the public key), we can remove the encryption of B_1' from the protocol altogether.

GAME 3:



Next, note that B_1' perfectly one-time pads the value of L_2. This breaks the dependency of L_2 on B_2, s_2 and B_1. It does not change the distribution of L_2, if L_2 is instead directly sampled uniformly from $\{0, 1\}$. Since B_2 is unused, we remove it in the following game, and y, b, y', b', s_1, s_2 can be ignored.

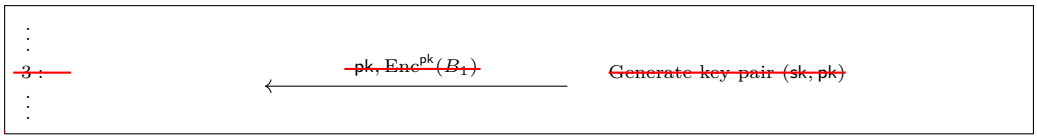
GAME 4:



By the same argument as for the transition from Game 2 to Game 3, we remove the encryption of B_1 from the following game. This introduces at most a negligible change in the success probability of the adversary.

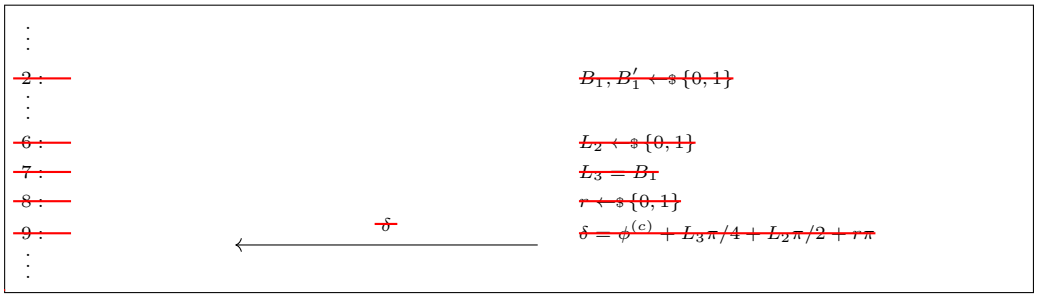
Finally, since the encryption scheme is not in use anymore, we can also remove the key generation and the message containing the public key without affecting the adversary's success probability.

GAME 5:



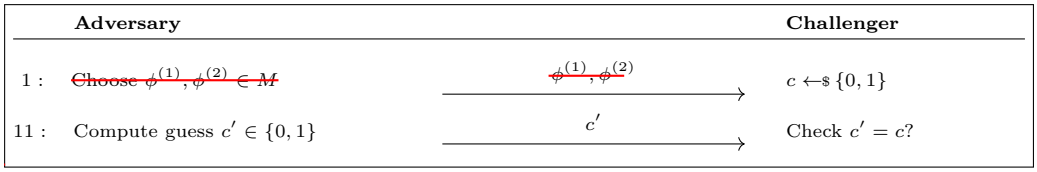
We now see that δ is a uniformly random number, L_2, L_3, and r being i.i.d. uniform bits. Therefore, the calculation and the message containing δ can be removed from the protocol without affecting the adversary.

GAME 6:



In Game 6, the inputs of the adversary are ignored by the challenger. Therefore, the computation angles $\phi^{(1)}$, $\phi^{(2)}$ can equally be removed from the protocol, leaving us with the final Game 7.

GAME 7:



Game 7 exactly describes the adversary's uninformed guess of the outcome of an independent bit flip. Therefore, by a simple information-theoretic argument, any strategy for the adversary will lead to a success probability of exactly 1/2.

We summarize:

$$\begin{aligned}
&\text{Succ-Pr}_{\text{Game1}} = \text{Succ-Pr}_{\text{Game2}}, \quad |\text{Succ-Pr}_{\text{Game2}} - \text{Succ-Pr}_{\text{Game3}}| \leq \text{negl}(\lambda),\\
&\text{Succ-Pr}_{\text{Game3}} = \text{Succ-Pr}_{\text{Game4}}, \quad |\text{Succ-Pr}_{\text{Game4}} - \text{Succ-Pr}_{\text{Game5}}| \leq \text{negl}(\lambda),\\
&\text{Succ-Pr}_{\text{Game5}} = \text{Succ-Pr}_{\text{Game6}} = \text{Succ-Pr}_{\text{Game7}} = \frac{1}{2},
\end{aligned}$$

and therefore we have $\left|\text{Succ-Pr}_{\text{Game1}} - \frac{1}{2}\right| \leq \text{negl}(\lambda)$ concluding the proof. □

subsectionGeneral QF-UBQC

We extend the security proof from Section 5.2 to UBQC on polynomially-sized graphs, i.e. MBQC computations on a polynomial number of qubits. The proof works by induction over the number n of qubits in the graph. Lemma 5.4 with $n = 1$ serves as start of the induction. We continue with proving the induction step, assuming the security of QF-UBQC on graphs of size n and showing its security for any graph of size $n + 1$. The induction step works analogously to the proof of Lemma 5.4. In this way, the security

of QF-UBQC on n qubits is reduced to the security of QF-UBQC on $n-1$ qubits, which can be reduced to the security of QF-UBQC on even one qubit less. This chain continues down to the single-qubit case whose security was already established in Lemma 5.4. Every step in this chain adds at most a negligible probability to the adversary's advantage. Therefore, also any such chain of polynomial length adds no more than a negligible probability to the adversary's advantage in the single-qubit case, thereby showing the security of the protocol on n qubits. We now provide the full details of the induction step.

Details of the proof of Theorem 5.3. The proof works by induction over the number n of qubits in the graph. Lemma 5.4 with $n = 1$ serves as start of the induction. We continue with proving the induction step, assuming the security of QF-UBQC on graphs of size n and showing its security for any graph of size $n+1$.

We first state some useful observations for the proof:

1. The existence of a *flow* on the MBQC graph induces a total order of all qubits in the graph, the order in which the qubits are measured. We subsequently assume that in the protocol the qubits are processed in exactly this order.
2. Given this order on the qubits, the dependence of the computation angles δ_i on outcomes of measurement of other qubits takes a specific form, they solely depend on previous (corrected) measurement outcomes $\{\bar{s}_j, j < i\}$, i.e. outcomes of measurements of qubits smaller in the order induced by the flow. Since the exact form of this dependence does not matter for the following proof, we denote the update of the angles in the following general way:

$$\begin{aligned}\delta_i =&(-1)^{f_1(s_1,r_1,\ldots,s_{i-1},r_{i-1})}\phi_i + \theta_1\pi/4 + \theta_2\pi/2 + \theta_3\pi \\ &+ r_i\pi + f_2(s_1, r_1, \ldots, s_{i-1}, r_{i-1})\pi,\end{aligned}$$

   with (deterministic families of) functions f_1 and f_2.
3. Given the previous observation, one can generalize the statement of the theorem to a family of protocols for any functions f_1 and f_2. For the remainder of the proof, we do hence not assume anything about these two functions, but simply take them as given. The actual statement of the theorem then follows as a special case, imposing that f_1 and f_2 describe the MBQC correction terms.

Given these observations, the rest of the proof works analogously to the proof of Lemma 5.4, removing one-by-one the ciphertexts of the two basis bits B_1, B_1' of the last QFactory instance, before removing the last measurement angle δ and reducing the protocol on $n+1$ qubits to the protocol on one qubit less. □

By the inductive nature of this proof, every qubit – and hence every QFactory instance – adds some negligible value to the success probability of the malicious adversary. This explains that the security only holds for polynomially-sized graphs. For an MBQC graph on a superpolynomial number of qubits, there are no guarantees anymore that these small errors don't add up to something constant. Having in mind that QFactory is trivially broken by exponential adversaries, it is clear that this is the best we can expect.

Acknowledgements.

The authors thank Céline Chevalier, Omar Fawzi, Daniel Jost, and Luka Music for useful discussions. LC also thanks M.T. This work has been supported in part by

grant FA9550-17-1-0055, by the European Union's H2020 Programme under grant agreement number ERC-669891, and by the French ANR Project ANR-18-CE39-0015 (CryptiQ). EK acknowledges support from the EPSRC Verification of Quantum Technology grant (EP/N003829/1), the EPSRC Hub in Quantum Computing and Simulation (EP/T001062/1), and the UK Quantum Technology Hub: NQIT grant (EP/M013243/1). LC and DL gratefully acknowledge support from the French ANR project ANR-18-CE47-0010 (QUDATA). LC, EK, and DL acknowledge funding from the EU Flagship Quantum Internet Alliance (QIA) project. AM gratefully acknowledges funding from the AFOSR MURI project "Scalable Certification of Quantum Computing Devices and Networks". This work was partly done while AM was at University of Edinburgh, UK where it was supported by EPSRC Verification of Quantum Technology grant (EP/N003829/1).

References

ABOE08. Dorit Aharonov, Michael Ben-Or, and Elad Eban. Interactive proofs for quantum computations. *arXiv preprint arXiv:0810.5375*, 2008.

ACGK19. Scott Aaronson, Alexandru Cojocaru, Alexandru Gheorghiu, and Elham Kashefi. Complexity-Theoretic Limitations on Blind Delegated Quantum Computation. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, 2019.

AFK87. Martin Abadi, Joan Feigenbaum, and Joe Kilian. On hiding information from an oracle. In *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 195–203. ACM, 1987.

BCP+14. Nicolas Brunner, Daniel Cavalcanti, Stefano Pironio, Valerio Scarani, and Stephanie Wehner. Bell nonlocality. *Reviews of Modern Physics*, 86(2):419, 2014.

BFK09. Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. Universal blind quantum computation. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pages 517–526. IEEE, 2009.

BJ15. Anne Broadbent and Stacey Jeffery. Quantum homomorphic encryption for circuits of low t-gate complexity. In *Annual Cryptology Conference*, pages 609–629. Springer, 2015.

BPW03. Michael Backes, Birgit Pfitzmann, and Michael Waidner. A composable cryptographic library with nested operations. In *Proceedings of the 10th ACM conference on Computer and communications security*, pages 220–230. ACM, 2003.

Bra18. Zvika Brakerski. Quantum fhe (almost) as secure as classical. In *Annual International Cryptology Conference*, pages 67–95. Springer, 2018.

Bro15a. Anne Broadbent. Delegating private quantum computations. *Canadian Journal of Physics*, 93(9):941–946, 2015.

Bro15b. Anne Broadbent. How to verify a quantum computation. *arXiv preprint arXiv:1509.09180*, 2015.

Can01. Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pages 136–145. IEEE, 2001.

CCKW18. Alexandru Cojocaru, Léo Colisson, Elham Kashefi, and Petros Wallden. On the possibility of classical client blind quantum computing. *arXiv preprint arXiv:1802.08759*, 2018.

CCKW19. Alexandru Cojocaru, Léo Colisson, Elham Kashefi, and Petros Wallden. Qfactory: Classically-instructed remote secret qubits preparation. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology – ASIACRYPT 2019*, pages 615–645. Springer International Publishing, 2019.

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

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

DK06. Vincent Danos and Elham Kashefi. Determinism in the one-way model. *Physical Review A*, 74(5):052310, 2006.

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

DKL12. Vedran Dunjko, Elham Kashefi, and Anthony Leverrier. Blind quantum computing with weak coherent pulses. *Physical Review Letters*, 108(20):200502, 2012.

DL70. E. B. Davies and J. T. Lewis. An operational approach to quantum probability. *Communications in Mathematical Physics*, 17(3):239–260, September 1970.

DSS16. Yfke Dulek, Christian Schaffner, and Florian Speelman. Quantum homomorphic encryption for polynomial-sized circuits. In *Annual Cryptology Conference*, pages 3–32. Springer, 2016.

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

Fit17. Joseph F Fitzsimons. Private quantum computation: an introduction to blind quantum computing and related protocols. *npj Quantum Information*, 3(1):23, 2017.

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

GKK19. Alexandru Gheorghiu, Theodoros Kapourniotis, and Elham Kashefi. Verification of quantum computation: An overview of existing approaches. *Theory of computing systems*, 63(4):715–808, 2019.

Gol01. Oded Goldreich. *Foundations of Cryptography*. Cambridge University Press, Aug 2001.

GRW80. G. C. Ghirardi, Alberto Rimini, and Tullio Weber. A general argument against superluminal transmission through the quantum mechanical measurement process. *Lettere al Nuovo Cimento (1971-1985)*, 27:293–298, 1980.

GV19. Alexandru Gheorghiu and Thomas Vidick. Computationally-secure and composable remote state preparation. *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1024–1033, 2019.

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

JM17. Daniel Jost and Ueli Maurer. Context-restricted indifferentiability: Generalizing UCE and implications on the soundness of hash-function constructions. *IACR Cryptol. ePrint Arch.*, 2017:461, 2017.

KMW17. Elham Kashefi, Luka Music, and Petros Wallden. The quantum cut-and-choose technique and quantum two-party computation. *arXiv preprint arXiv:1703.03754*, 2017.

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

KRK12. Alastair Kay, Ravishankar Ramanathan, and Dagomir Kaszlikowski. Optimal Asymmetric Quantum Cloning. *arXiv e-prints*, page arXiv:1208.5574, August 2012.

KW17. Elham Kashefi and Petros Wallden. Garbled quantum computation. *Cryptography*, 1(1):6, 2017.

Mah18a. Urmila Mahadev. Classical homomorphic encryption for quantum circuits. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 332–338. IEEE Computer Society, 2018.

Mah18b. Urmila Mahadev. Classical verification of quantum computations. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 259–267. IEEE Computer Society, 2018.

Man19. Atul Mantri. Secure delegated quantum computing, Phd thesis, 2019.

Mau11. Ueli Maurer. Constructive cryptography–a new paradigm for security definitions and proofs. In *Theory of Security and Applications*, pages 33–56. Springer, 2011.

MDF17. Atul Mantri, Tommaso F Demarie, and Joseph F Fitzsimons. Universality of quantum computation with cluster states and (X, Y)-plane measurements. *Scientific Reports*, 7:42861, 2017.

MDMF17. Atul Mantri, Tommaso F Demarie, Nicolas C Menicucci, and Joseph F Fitzsimons. Flow ambiguity: A path towards classically driven blind quantum computation. *Physical Review X*, 7(3):031004, 2017.

MF12. Tomoyuki Morimae and Keisuke Fujii. Blind topological measurement-based quantum computation. *Nature Communications*, 3:1036, 2012.

MK13. Tomoyuki Morimae and Takeshi Koshiba. Composable security of measuring-alice blind quantum computation. *arXiv preprint arXiv:1306.2113*, 2013.

MK14. Tomoyuki Morimae and Takeshi Koshiba. Impossibility of perfectly-secure delegated quantum computing for classical client. *arXiv preprint arXiv:1407.1636*, 2014.

MP12. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. *Lecture Notes in Computer Science*, page 700–718, 2012.

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

NC00. Michael A Nielsen and Isaac Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

Nie06. Michael A Nielsen. Cluster-state quantum computation. *Reports on Mathematical Physics*, 57(1):147–161, 2006.

PPK+09. Marcin Pawłowski, Tomasz Paterek, Dagomir Kaszlikowski, Valerio Scarani, Andreas Winter, and Marek Żukowski. Information causality as a physical principle. *Nature*, 461(7267):1101–1104, 2009.

RB01. Robert Raussendorf and Hans J Briegel. A one-way quantum computer. *Physical Review Letters*, 86(22):5188, 2001.

Reg09. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34, 2009.

RUV12. Ben W Reichardt, Falk Unger, and Umesh Vazirani. A classical leash for a quantum system: Command of quantum systems via rigidity of chsh games. *arXiv preprint arXiv:1209.0448*, 2012.

TMM+18. Yuki Takeuchi, Atul Mantri, Tomoyuki Morimae, Akihiro Mizutani, and Joseph F Fitzsimons. Resource-efficient verification of quantum computing using Serfling's bound. *arXiv preprint arXiv:1806.09138*, 2018.

Vid20. Thomas Vidick. Verifying quantum computations at scale: A cryptographic leash on quantum devices. *Bulletin of the American Mathematical Society*, 57(1):39–76, 2020.

A Game-Based Security and Constructive Cryptography

The main aim of our work is to prove possibility and impossibility results in different security models. We will in this paper focus mostly on two different notions: the game-based security model and the Constructive Cryptography framework.

The definition of game-based security is pretty straightforward: we define a *game* between a challenger and an (arbitrary) adversary: a protocol is secure if no adversary can win this game with "good" probability. The problem of this approach is that one game describes only one possible attack, and it is hard to list all the possible attacks against a protocol. Therefore, a protocol that proves to be secure in a specific game might not be secure in an arbitrary environment (composed with other protocols in parallel or in series).

Composable security on the other hand takes a different approach to phrasing the guarantees achieved by a protocol. Loosely speaking, a protocol is composable when it is shown to be secure in an arbitrarily adversarial environment[25], and where secure means that it achieves a well-defined ideal (secure by definition) resource. This means the protocol retains the desired functionality even if it is composed of other instances of its own or a completely different protocol. There are several approaches which provide a general framework to study this cryptographic definitions [Can01,BPW03,MR11], but we

[25] Of course, the environment may still be limited to "efficient" computations.

will focus in this paper on Constructive Cryptography (CC) (also known under the term Abstract Cryptography (AC)). In this section, we provide relevant terminologies (mostly adapted to our protocol) required to analyse composable security in this framework, introduced by Maurer and Renner in [MR11]. For more details, we refer readers to some of the previous works [Mau11,MR11,DFPR14,DK16].

The basic elements of AC are systems: objects with well-distinguished and labeled interfaces. The system uses interfaces to exchange information with the outside world and/or other systems. Systems are grouped in distinct classes: resources, converters, filters, and distinguisher.

Resource systems (or $\mathcal{I}$-resources) are devices with several interfaces in $\mathcal{I}$, in general, each of them accessible by a single agent: each interface represents the actions that are accessible by that player. Resources are the central elements of CC, and they are used at the abstract level to specify the relevant properties of a protocol. Note that in this work we only consider resources with two interfaces $I = \{A, B\}$ because our protocol consists of two parties (one client A and one server B).

A *converter system*, on the other hand, is always limited to two interfaces, an inside and an outside one. Converters are usually attached to the interfaces of a resource (or a group of resource as already explained), and the name reflects the fact that a converter *converts* the functionality of the resource's interface it is attached to into a new functionality on the outside. A resource having a converter attached to one of its interfaces continues to qualify as a resource, possibly equipped with new functionalities. Usually, if $\alpha \in \Sigma$ is a converter and $\mathcal{R}$ a resource, we write $\alpha^i\mathcal{R}$ to denote new resource where the inner interface of α is connected to the interface i of $\mathcal{R}$, the outer interface of α being the new interface i. But because in our case we have two interfaces $\mathcal{I} = \{A, B\}$, we will put the converter on the left of the resource when it is plugged on the interface A, and we will put the converter on the right of the resource when it is plugged on the interface B: $\alpha^A\mathcal{R}$ is denoted $\alpha\mathcal{R}$ while $\alpha^B\mathcal{R}$ is $\mathcal{R}\alpha$.

A *filter* (usually denoted $\vdash$) is a special converter used to force a honest behaviour on a given interface of a resource. They are usually used to prove the correctness of a protocol, as they describe what can be done in an honest run. They are removed when we want to provide full power to a cheating adversary or to a simulator. Usually, in order to keep the filter simple, the functionality accepts as a first message a bit c which says if the party wants to behave honestly ($c = 0$) or maliciously ($c = 1$). That way, the filter $\vdash^{c=0}$ (or simply $\vdash$) just sends $c = 0$ to the resource, and then forwards all the messages between it's inner and outer interface.

A *distinguisher* helps to quantify the distance between resources. Given an n-interface resource $\mathcal{R}$, a distinguisher $D \in \mathcal{D}$ outputs a bit determined after interacting with the n interfaces of $\mathcal{R}$ (we denote by $D\mathcal{R}$ this random variable). Then the distance (actually it is a pseudo-metric) between two resources $\mathcal{R}$ and $\mathcal{S}$ is defined by the best advantage ε a distinguisher $D \in \mathcal{D}$ can achieve when trying to determine which resource it is interacting with. This leads to the following definition:

$$\mathcal{R} \approx_\varepsilon \mathcal{S} :\Longleftrightarrow \Delta^{\mathcal{D}}(\mathcal{R}, \mathcal{S}) \leq \varepsilon \tag{60}$$

with $\Delta^{\mathcal{D}}(\mathcal{R}, \mathcal{S}) = \sup_{D\in\mathcal{D}} \Delta(D\mathcal{R}, D\mathcal{S})$, where $\Delta(D\mathcal{R}, D\mathcal{S})$ is the statistical distance between the distributions $D\mathcal{R}$ and $D\mathcal{S}$. Note that $\Delta^{\mathcal{D}}$ defines a pseudo-metric: $\forall \varepsilon > 0, (\mathcal{R}, \mathcal{S}, \mathcal{T}) \in \Phi^3$,

$$\Delta^{\mathcal{D}}(\mathcal{R}, \mathcal{R}) = 0 \tag{61}$$

$$\Delta^{\mathcal{D}}(\mathcal{R}, \mathcal{S}) = \Delta^{\mathcal{D}}(\mathcal{R}, \mathcal{S}) \tag{62}$$

$$\Delta^{\mathcal{D}}(\mathcal{R}, \mathcal{S}) \leq \Delta^{\mathcal{D}}(\mathcal{R}, \mathcal{T}) + \Delta^{\mathcal{D}}(\mathcal{T}, \mathcal{R}) \tag{63}$$

In the general Constructive Cryptography framework, we do not need to specify how the different systems are constructed, we just need to have some general properties on them: we basically require $\langle \Phi, \Sigma \rangle$ to be a *cryptographic algebra*, and the pseudo-metric must be *compatible* with $\langle \Phi, \Sigma \rangle$ (see for example [Mau11, Sec. 4] for precise definitions). And as soon as $\mathcal{D}$ can "absorb" the converters and the resources, then $\mathcal{D}$ is compatible with $\langle \Phi, \Sigma \rangle$ [Mau11, Lem. 1]. That way, it is possible to derive different notions of security by just changing the sets Φ (resources), Σ (converters) and $\mathcal{D}$ (distinguisher) and making sure they respect these general properties. In the paper, we will focus mostly on two definitions (respecting the above properties): when all the systems (resources, converters, and distinguishers) are *feasible* (in our case we mean they run in polynomial time on a quantum machine), denoted as $(\Phi^f, \Sigma^f, \mathcal{D}^f)$ we will say that the security is *computational*. If the systems are unbounded $(\Phi^u, \Sigma^u, \mathcal{D}^u)$ we will refer to *information-theoretic* security.

Note that the impossibility results presented in this paper apply in both computational and information-theoretic security, and because we only focus on these two settings, $\langle \Phi, \Sigma \rangle$ will always be a cryptographic algebra, and the pseudo-metric $\Delta^{\mathcal{D}}$ is always compatible with it. When a property is valid only for one set of distinguishers, we will write this set above the $\approx$ sign, like for example $\mathcal{R} \approx_{\varepsilon}^{\mathcal{D}^u} \mathcal{S}$.
A main theorem is that any such construction achieve (general) composability:

Lemma A.1 ([MR11, Thm. 1][Mau11, Thm. 3]). *The construction* $\longrightarrow$ *is* (generally) composable*, i.e. for all* $(\varepsilon, \varepsilon') \in \mathbb{R}^+$, $(\mathcal{R}, \mathcal{S}, \mathcal{T}) \in \Phi^3$, $\boldsymbol{\pi} \in \Sigma^2$*:*

- *we have sequential composability:* $(\mathcal{R} \xrightarrow[\varepsilon]{\boldsymbol{\pi}} \mathcal{S} \wedge \mathcal{S} \xrightarrow[\varepsilon']{\boldsymbol{\pi}'} \mathcal{T}) \Rightarrow \mathcal{R} \xrightarrow[\varepsilon+\varepsilon']{\boldsymbol{\pi}\circ\boldsymbol{\pi}'} \mathcal{T}$,
- *we have parallel composability:* $(\mathcal{R} \xrightarrow[\varepsilon]{\boldsymbol{\pi}} \mathcal{S} \wedge \mathcal{R}' \xrightarrow[\varepsilon']{\boldsymbol{\pi}'} \mathcal{S}') \Rightarrow \mathcal{R}\|\mathcal{R}' \xrightarrow[\varepsilon+\varepsilon']{\boldsymbol{\pi}|\boldsymbol{\pi}'} \mathcal{S}\|\mathcal{S}'$
- $\mathcal{R} \xrightarrow[0]{\mathrm{id}} \mathcal{R}$

where $|$ *(resp.* $\circ$*) represents the parallel (resp. serial) composition of protocols,* $\|$ *is the merging of resources, and* id *is the identity converter.*

B QFactory: Remote State Preparation, Revisited

The construction of the QFactory protocol relies on a family of functions with certain cryptographic properties, specifically, a 2-regular homomorphic-hardcore family of functions. For the formal definition of these properties, see [CCKW19].

We first begin by recalling the formal description of the protocol in Appendix B.1 and then in Appendix B.2 and Appendix B.3 we present the results concerning the correctness and security of QFactory.

B.1 4-states and 8-states QFactory protocol

Protocol 2 4-states QFactory: classical delegation of the BB84 states ([CCKW19])

Requirements: Public: A 2-regular homomorphic-hardcore family $\mathcal{F}$ with respect to $\{h_k\}$ and d_0. For simplicity, we will represent the sets $\mathcal{D}'$ (respectively $\mathcal{R}$) using n (respectively m) bits strings: $\mathcal{D}' = \{0,1\}^n$, $\mathcal{R} = \{0,1\}^m$.

Stage 1: Preimages superposition

1. Client runs the algorithm $(k, t_k) \leftarrow \mathrm{Gen}_{\mathcal{F}}(1^n)$.

2. Client instructs Server to prepare one register at $\otimes^n H|0\rangle$ and second register initiated at $|0\rangle^m$.
3. Server receives k from the client and applies U_{f_k} using the first register as control and the second as target.
4. Server measures the second register in the computational basis, obtains the outcome y. The combined state is given by $(|x\rangle + |x'\rangle) \otimes |y\rangle$ with $f_k(x) = f_k(x') = y$ and $y \in \text{Im}\, f_k$.

Stage 2: Output preparation

1. Server applies U_{h_k} on the preimage register $|x\rangle + |x'\rangle$ as control and another qubit initiated at $|0\rangle$ as target. Then, measures all the qubits, but the target in the $\{\frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)\}$ basis, obtaining the outcome $b = (b_1, ..., b_n)$. Now, the Server returns both y and b to the Client.
2. Client using the trapdoor t_k computes the preimages of y:
   - if y does not have exactly two preimages x, x' (the server is cheating with overwhelming probability), defines $B_1 = d_0(t_k)$, and chooses $B_2 \in \{0, 1\}$ uniformly at random
   - if y has exactly two preimages x, x', defines $B_1 = h_k(x) \oplus h_k(x') = d_0(t_k)$, and B_2.

Output: The quantum state that the Server has generated is (with overwhelming probability [26]) the BB84 state $|\mathtt{out}\rangle = H^{B_1} X^{B_2} |0\rangle$ (see Eq. (65) and Eq. (66) for the exact value of B_1 and B_2). The output of the Server is a quantum state $|\mathtt{out}\rangle$ and the output of the Client is given by (B_1, B_2) (2 bits).

Protocol 3 8-states QFactory: classical delegation of the $|+_\theta\rangle$ states ([CCKW19])

Requirements: Same as in Protocol 2

Input: Client runs twice the algorithm $Gen_{\mathcal{F}}(1^n)$, obtaining $(k^1, t_k^1), (k^2, t_k^2)$. Client keeps t_k^1, t_k^2 private.

Protocol Steps:

1. Client runs 4-states QFactory Protocol 2 to obtain a state $|\mathtt{in}_1\rangle$ and a "rotated" 4-states QFactory to obtain a state $|\mathtt{in}_2\rangle$ (by rotated 4-states QFactory we mean a 4-states QFactory, but where the last set of measurements in the $|\pm\rangle$ basis is replaced by measurements in the $\left|\pm_{\frac{\pi}{2}}\right\rangle$ basis).
2. Client records measurement outcomes (y^1, b^1), (y^2, b^2) and computes and stores the corresponding indices of the output states of the 2 runs of 4-states QFactory protocol: (B_1, B_2) for $|\mathtt{in}_1\rangle$ and (B_1', B_2') for $|\mathtt{in}_2\rangle$.
3. Client instructs Server to apply the Merge Gadget in Fig. 12 ([CCKW19]) on the states $|\mathtt{in}_1\rangle$, $|\mathtt{in}_2\rangle$.
4. Server returns the 2 measurement results s_1, s_2.
5. Client using (B_1, B_2), (B_1', B_2'), s_1, s_2 computes the index $L = L_1 L_2 L_3 \in \{0, 1\}^3$ of the output state (see Eq. (67), Eq. (68), and Eq. (69) for the exact value of L_1, L_2, and L_3, respectively.)

Output: The output of the Server is (with overwhelming probability) a quantum state $|\mathtt{out}\rangle := |+_{L\frac{\pi}{4}}\rangle$ and the output of the Client is given by L (3 bits).

B.2 Correctness of QFactory

In an honest run, the description of the output state of the protocol depends on measurement results $y \in \text{Im}\, f_k$ and b, but also on the 2 preimages x and x' of y.

The output state of 4-states QFactory belongs to the set of states $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ and its exact description is the following:

[26] As for the previous protocol, the probability comes from the probability of $\mathcal{F}$ being a 2-regular homomorphic-hardcore family of functions

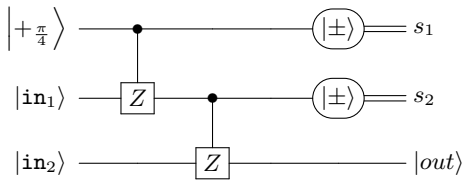


Fig. 12: Merge Gadget (Taken from [CCKW19])

Theorem B.1 (4-states QFactory is correct ([CCKW19])). *In an honest run, with overwhelming probability the output state $|\mathtt{out}\rangle$ of the 4-states QFactory Protocol 2 is a BB84 state whose basis is $B_1 = h_k(x) \oplus h_k(x') = d_0$, and:*

- *if $d_0 = 0$, then the state is $|h_k(x)\rangle$ (computational basis, also equal to $|h_k(x')\rangle$)*
- *if $d_0 = 1$, then if $\sum_i b_i \cdot (x_i \oplus x'_i) = 0 \bmod 2$, the state is $|+\rangle$, otherwise the state is $|-\rangle$ (Hadamard basis).*

i.e.

$$|\mathtt{out}\rangle = H^{B_1} X^{B_2} |0\rangle \tag{64}$$

with

$$B_1 = h_k(x) \oplus h_k(x') = d_0 \tag{65}$$
$$B_2 = (d_0 \times (b \cdot (x \oplus x'))) \oplus h(x)h(x') \tag{66}$$

(the inner product is taken modulo 2, and $x \oplus x'$ is a bitwise xor)

Theorem B.2 (8-states QFactory is correct ([CCKW19])). *In an honest run, the Output state of the 8-states QFactory Protocol is of the form $|+_{L \cdot \frac{\pi}{4}}\rangle$, where $L = L_1 L_2 L_3 \in \{0,1\}^3$, defined as:*

$$L_1 = B'_2 \oplus B_2 \oplus [B_1 \cdot (s_1 \oplus s_2)] \tag{67}$$
$$L_2 = B'_1 \oplus [(B_2 \oplus s_2) \cdot B_1] \tag{68}$$
$$L_3 = B_1 \tag{69}$$

B.3 Security of QFactory

In any run of the protocol, honest or malicious, the state that the client believes that the server has is given by Theorem B.1. Therefore, the task that a malicious server wants to achieve, is to be able to guess, as good as he can, the description of the output state that the client (based on the public communication) thinks the server has produced. In particular, in our case, the server needs to guess the bit B_1 (corresponding to the basis) of the (honest) output state.

Definition B.3 (4 states basis blindness). *We say that a protocol (π_A, π_B) achieves **basis-blindness** with respect to an ideal list of 4 states $S = \{S_{B_1,B_2}\}_{(B_1,B_2)\in\{0,1\}^2}$ if:*

- *S is the set of states that the protocol outputs, i.e.:*

$$\Pr[\,|\phi\rangle = S_{B_1 B_2} \in S \mid ((B_1, B_2), |\phi\rangle) \leftarrow (\pi_A \| \pi_B)\,] \geq 1 - \mathsf{negl}(n)$$

- *and no information is leaked about the index bit B_1 of the output state of the protocol, i.e for all QPT adversary $\mathcal{A}$:*

$$\Pr[\,B_1 = \tilde{B}_1 \mid ((B_1, B_2), \tilde{B}_1) \leftarrow (\pi_A \| \mathcal{A})\,] \leq 1/2 + \mathsf{negl}(n)$$

Theorem B.4 (4-states QFactory is secure ([CCKW19])). *Protocol 2 satisfies 4-states basis blindness with respect to the ideal list of states* $S = \{H^{B_1} X^{B_2} |0\rangle\}_{B_1,B_2} = \{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$.

Definition B.5 (8 states basis blindness). *Similarly, we say that a protocol* (π_A, π_B) *achieves* ***basis-blindness*** *with respect to an ideal list of 8 states* $S = \{S_{L_1,L_2,L_3}\}_{(L_1,L_2,L_3)\in\{0,1\}^3}$ *if:*

- S *is the set of states that the protocol outputs, i.e.:*

$$\Pr[\, |\phi\rangle = S_{L_1,L_2,L_3} \in S \mid ((L_1,L_2,L_3),|\phi\rangle) \leftarrow (\pi_A \| \pi_B)\,] = 1$$

- *and if no information is leaked about the "basis" bits* (L_2, L_3) *of the output state of the protocol, i.e for all QPT adversary* $\mathcal{A}$*:*

$$\Pr[\, L_2 = \tilde{L_2} \text{ and } L_3 = \tilde{L_3} \mid ((L_1,L_2,L_3),(\tilde{L_2},\tilde{L_3})) \leftarrow (\pi_A \| \mathcal{A})\,] \leq 1/4 + \mathsf{negl}(n)$$

Theorem B.6 (8-states QFactory is secure ([CCKW19])). *Protocol 3 satisfies 8-state basis blindness with respect to the ideal set of states* $S = \{|+_{\pi L/4}\rangle\}_{L\in\{0,\ldots,7\}} = \{|+\rangle, |+_{\frac{\pi}{4}}\rangle, .., |+_{\frac{7\pi}{4}}\rangle\}$.

C Distance Measures for Quantum States

Lemma C.1. *For any two self-adjoint trace-class operators* ρ, σ *it holds that*

$$\mathrm{Tr}(\rho\sigma) = \frac{1}{2}\left[\mathrm{Tr}(\rho^2) + \mathrm{Tr}(\sigma^2)\right] - \frac{1}{2}\|\rho - \sigma\|_{HS}^2,$$

where the Hilbert-Schmidt norm is defined as

$$\|A\|_{HS} = \sqrt{\mathrm{Tr}(A^* A)}.$$

Proof. This follows directly from the relation

$$(\rho - \sigma)^2 = \rho^2 - \rho\sigma - \sigma\rho + \sigma^2$$

and the fact that ρ and σ are self-adjoint operators. □

The following lemma formalizes the following statement: If $\mathrm{Tr}(\rho\sigma)$ is close to 1, then both ρ and σ must be almost pure, and ρ and σ must be close. Note that Lemma C.2 holds in particular for density matrices ρ and σ, despite being stated for a more general class of operators.

Lemma C.2. *Let* $\varepsilon \geq 0$ *and* $\mathrm{Tr}(\rho\sigma) \geq 1 - \varepsilon$ *for two self-adjoint, positive semi-definite operators* ρ, σ *with trace less than 1. Then, it holds that*

1. $\mathrm{Tr}\left(\rho^2\right) \geq 1 - 2\varepsilon$,
2. $\mathrm{Tr}\left(\sigma^2\right) \geq 1 - 2\varepsilon$, *and*
3. $\|\rho - \sigma\|_{HS} \leq \sqrt{2\varepsilon}$.

Proof. 1. With the formula from Lemma C.1, we infer that

$$\mathrm{Tr}(\rho\sigma) \leq \frac{1}{2}\left[\mathrm{Tr}(\rho^2) + \mathrm{Tr}(\sigma^2)\right] \leq \frac{1}{2}\left[\mathrm{Tr}(\rho^2) + 1\right],$$

using the non-negativity of the Hilbert-Schmidt norm and the fact that $\mathrm{Tr}\left(\sigma^2\right) \leq 1$. Hence,

$$\mathrm{Tr}\left(\rho^2\right) \geq 2\,\mathrm{Tr}\left(\rho\sigma\right) - 1 \geq 1 - 2\varepsilon.$$

2. Analogously to 1.
3. Using $\mathrm{Tr}\left(\rho^2\right) \leq 1$ and $\mathrm{Tr}\left(\sigma^2\right) \leq 1$, we obtain

$$\begin{aligned} \mathrm{Tr}\left(\rho\sigma\right) &\leq 1 - \frac{1}{2}\left\|\rho - \sigma\right\|_{\mathrm{HS}}^2 \\ \Rightarrow \left\|\rho - \sigma\right\|_{\mathrm{HS}}^2 &\leq 2\left(1 - \mathrm{Tr}\left(\rho\sigma\right)\right) \leq 2\varepsilon, \end{aligned}$$

which implies the claim.

□

Lemma C.3. *Let λ be a security parameter and let ρ, σ be two density matrices of finite and fixed dimension. Then, the following statements are equivalent:*

1. $\mathrm{Tr}\left(\rho^2\right) \geq 1 - \mathrm{negl}(\lambda)$, $\mathrm{Tr}\left(\sigma^2\right) \geq 1 - \mathrm{negl}(\lambda)$, *and* $\mathrm{TD}\left(\rho - \sigma\right) \leq \mathrm{negl}(\lambda)$,
2. $\mathrm{Tr}\left(\rho\sigma\right) \geq 1 - \mathrm{negl}(\lambda)$,

where TD *denotes the trace distance.*

Proof. One direction of the equivalence follows directly from Lemma C.2. The other direction follows from the formula in Lemma C.1 and the fact that in finite-dimensional spaces the trace norm is equivalent to the Hilbert-Schmidt norm. □

Lemma C.4. *Let $\varepsilon_1, \varepsilon_2 \geq 0$. Let further $\mathrm{Tr}\left(\rho_1\rho_2\right) \geq 1 - \varepsilon_1$ and $\mathrm{Tr}\left(\rho_2\rho_3\right) \geq 1 - \varepsilon_2$ for self-adjoint, positive semi-definite operators ρ_1, ρ_2, ρ_3 with trace less than 1. Then it holds that $\mathrm{Tr}\left(\rho_1\rho_3\right) \geq 1 - 3\left(\varepsilon_1 + \varepsilon_2\right)$.*

Proof. From Lemma C.2 we know that $\mathrm{Tr}\left(\rho_1^2\right) \geq 1 - 2\varepsilon_1$, $\mathrm{Tr}\left(\rho_3^2\right) \geq 1 - 2\varepsilon_2$, and

$$\left\|\rho_1 - \rho_2\right\|_{\mathrm{HS}} \leq \sqrt{2\varepsilon_1}, \quad \left\|\rho_2 - \rho_3\right\|_{\mathrm{HS}} \leq \sqrt{2\varepsilon_2}.$$

By the triangle inequality for the Hilbert-Schmidt norm, it follows readily that

$$\left\|\rho_1 - \rho_3\right\|_{\mathrm{HS}} \leq \sqrt{2\varepsilon_1} + \sqrt{2\varepsilon_2}$$

and therefore

$$\left\|\rho_1 - \rho_3\right\|_{\mathrm{HS}}^2 \leq \left(\sqrt{2\varepsilon_1} + \sqrt{2\varepsilon_2}\right)^2 = 2\varepsilon_1 + 2\varepsilon_2 + 4\sqrt{\varepsilon_1}\sqrt{\varepsilon_2} \leq 4\left(\varepsilon_1 + \varepsilon_2\right)$$

where we applied the inequality of the geometric mean to obtain the last bound. Using the formula from Lemma C.1, we then conclude that

$$\begin{aligned} \mathrm{Tr}\left(\rho_1\rho_3\right) &= \frac{1}{2}\left[\mathrm{Tr}(\rho_1^2) + \mathrm{Tr}(\rho_3^2)\right] - \frac{1}{2}\left\|\rho_1 - \rho_3\right\|_{\mathrm{HS}}^2 \\ &\geq \frac{1}{2}\left[1 - 2\varepsilon_1 + 1 - 2\varepsilon_2\right] - \frac{1}{2}4\left(\varepsilon_1 + \varepsilon_2\right) \geq 1 - 3\left(\varepsilon_1 + \varepsilon_2\right). \end{aligned}$$

□