Universally Composable Quantum Multi-party Computation*

Dominique Unruh

Saarland University

Abstract. The Universal Composability model (UC) by Canetti (FOCS 2001) allows for secure composition of arbitrary protocols. We present a quantum version of the UC model which enjoys the same compositionality guarantees. We prove that in this model statistically secure oblivious transfer protocols can be constructed from commitments. Furthermore, we show that every statistically classically UC secure protocol is also statistically quantum UC secure. Such implications are not known for other quantum security definitions. As a corollary, we get that quantum UC secure protocols for general multi-party computation can be constructed from commitments.

1 Introduction

Since the inception of quantum key distribution by Bennett and Brassard [4], it has been known that quantum communication permits to achieve protocol tasks that are impossible given only a classical channel. For example, a quantum key distribution scheme [4] permits to agree on a secret key that is statistically secret, using only an authenticated but not secret channel. (By statistical security we mean security against computationally unbounded adversaries, also known as information-theoretical security.) In contrast, when using only classical communication, it is easy to see that such a secret key can always be extracted by a computationally sufficiently powerful adversary. Similarly, based on an idea by Wiesner [25], Bennett, Brassard, Crépeau, and Skubiszewska [5] presented a protocol that was supposed to construct a statistically secure oblivious transfer[1] protocol from a commitment, another feat that is easily seen to be impossible classically.[2] Oblivious transfer, on the other hand, has been recognized by Kilian [15] to securely evaluate arbitrary functions. Unfortunately, the protocol of Bennett et al. could, at the time, not be proven secure, and the first complete proof

* Funded by the Cluster of Excellence "Multimodal Computing and Interaction".

[1] In an oblivious transfer protocol, Alice holds two bitstrings m_0, m_1, and Bob a bit c. Bob is supposed to get m_c but not m_{1-c}, and Alice should not learn c.

[2] We remark that, on the other hand, Mayers [16] shows that also in the quantum case, constructing a statistically secure commitment scheme *without any additional assumption* is impossible. However, under additional assumptions like in the quantum bounded storage model by Damgård, Fehr, Salvail, and Schaffner [10], statistically secure bit commitment is possible. See Section 1.1 for a discussion of the implications of Mayers' impossibility result for our result.

H. Gilbert (Ed.): EUROCRYPT 2010, LNCS 6110, pp. 486–505, 2010.

of (a variant of) that protocol was given almost two decades later by Damgård, Fehr, Lunemann, Salvail, and Schaffner [9].

Yet, although the oblivious transfer protocol satisfies the intuitive secrecy requirements of oblivious transfer, in certain cases the protocol might lose its security when used in a larger context. In other words, there are limitations on how the protocol can be composed. For example, no security guarantee is given when several instances of the protocol are executed concurrently (see the full version [21] for a more detailed explanations of the various restrictions).

The problem of composability has been intensively studied by the classical cryptography community (here and in the following, we use the word classical as opposed to quantum). To deal with this problem in a general way, Canetti [7] introduced the notion of Universal Composability, UC for short (Pfitzmann and Waidner [19] independently introduced the equivalent Reactive Simulatability framework). The UC framework allows to express the security of a multitude of protocol tasks in a unified way, and any UC-secure protocol automatically enjoys strong composability guarantees (so-called universal composability). In particular, such a protocol can be run concurrently with others, and it can be used as a subprotocol of other protocols in a general way. Ben-Or and Mayers [3] and Unruh [20] have shown that the idea of UC-security can be easily adapted to the quantum setting and have independently presented quantum variants of the UC notion. These notions enjoy the same strong compositionality guarantees. Shortly afterwards, Ben-Or, Horodecki, Leung, Mayers, and Oppenheim [2] showed that many quantum key distribution protocols are quantum-UC-secure.

Our contribution. In this work, we use the UC framework to show the existence of a statistically secure and universally composable oblivious transfer protocol that uses only a commitment scheme. Towards this goal, we first present a new definition of quantum-UC-security. In our opinion, our notion is technically simpler than the notions of Ben-Or and Mayers [3] and Unruh [20]. We believe that this may also help to increase the popularity of this notion in the quantum cryptography community and to show the potential for using UC-security in the design of quantum protocols. Second, we show that a variant of the protocol by Bennett et al. [5] is indeed a UC-secure oblivious transfer protocol. By composing this protocol with a UC-secure protocol for general multi-party computations by Ishai, Prabhakaran, and Sahai [13], we get UC-secure protocols for general multi-party computations using only commitments and a quantum channel – this is easily seen to be impossible in a purely classical setting.

UC-secure quantum oblivious transfer. The oblivious transfer (OT) protocol used in this paper is essentially the same as the protocol proposed by Damgård et al. [9] which in turn is based on a protocol by Bennett et al. [5]. The basic idea of the protocol is that Alice encodes a random sequence $\tilde{x}$ of bits as a quantum state, each bit randomly either in the computational basis or in

the diagonal basis.[3] Then Bob is supposed to measure all qubits, this time in random bases of his choosing. Then Alice sends the bases she used to Bob. Let $I_=$ denote the set of indices of the bits $\tilde{x}_i$ where Alice and Bob chose the same basis, and $I_\neq$ the set of indices of the bits where Alice and Bob chose different bases. Assume that Bob wants to receive the message m_c out of Alice's messages m_0, m_1. Then Bob sets $I_c := I_=$ and $I_{1-c} := I_\neq$ and sends (I_0, I_1) to Alice. Alice will not know which of these two sets is which and hence does not learn c. Bob will know the bits $\tilde{x}_i$ at indices $i \in I_c$. But even a dishonest Bob, assuming that he measured the whole quantum state, will not know the bits at indices $i \in I_{1-c}$ since he used the wrong bases for these bits. Thus Alice uses the bits at I_0 to mask her message m_0, and the bits at I_1 to mask her message m_1. Then Bob can recover m_c but not m_{1-c}. (To deal with the fact that a malicious Bob might have partial knowledge about the bits at I_{1-c}, we use so-called privacy amplification to extract a near uniformly mask from these bits.)

The problem with this analysis is that we have assumed that a malicious Bob measures the whole quantum state upon reception. But instead, Bob could store the quantum state until he learns the bases that Alice used, and then use these bases to measure all bits $\tilde{x}_i$ accurately. Hence, we need to force a dishonest Bob to measure all bits before Alice sends the bases. The idea of Bennett et al. [5] is to introduce the following test: Bob has to commit to the bases he used and to his measurement outcomes. Then Alice picks a random subset of the bits, and Bob opens the commitments on his bases and outcomes corresponding to this subset of bits. Alice then checks whether Bob's measurement outcomes are consistent with what Alice sent. If Bob does not measure enough bits, then he will commit to the wrong values in many of the commitments, and there will be a high probability that Alice detects this.

It was a long-standing open problem what kind of a commitment needs to be used in order for this protocol to be secure. Damgård et al. [9] give criteria for the commitment scheme under which the OT protocol can be proven to have so-called stand-alone security; stand-alone security, however, does not give as powerful compositionality guarantees as UC-security. In order to achieve UC-security, we assume that the commitment is given as an ideal functionality. Then we have to show UC-security in the case of a corrupted Alice, and UC-security in the case of a corrupted Bob. The case of a corrupted Alice is simple, as one can easily see that no information flows from Bob to Alice (the commitment functionality does, by definition, not leak any information about the committed values). The case of a corrupted Bob is more complex and requires a careful analysis about the amount of information that Bob can retrieve about Alice's bits. Such an analysis has already been performed by Damgård et al. [9] in their setting. Fortunately, we do not need to repeat the analysis. We show that under certain special conditions, stand-alone security already implies UC-security. Since in the case of a corrupted Bob,

[3] If we were to use photons for transmission, in the computational basis we might encode the bit 0 as a vertically polarized photon and the bits 1 as a horizontally polarized photon. In the diagonal basis we might encode the bit 0 as a 45°-polarized photon, and the bit 1 as a 135°-polarized photon.

these conditions are fulfilled, we get the security in the case of a corrupted Bob as a corollary from the work by Damgård et al. [9].

In Section 4, we show that the OT protocol by Damgård et al. [9], when using an ideal functionality for the commitment, is statistically quantum-UC-secure. Furthermore, the universal composition theorem guarantees that we can replace the commitment functionality by any quantum-UC-secure commitment protocol.

Quantum lifting and multi-party computation. We are now equipped with a statistically quantum-UC-secure OT protocol π_{QOT} in the commitment-hybrid model. As noted first by Kilian [15], OT can be used for securely evaluating arbitrary functions, short, OT is complete for multi-party computation. Furthermore, Ishai, Prabhakaran, and Sahai [13] showed that for any functionality $\mathcal{G}$ (even interactive functionalities that proceed in several rounds), there is a classical protocol $\rho^{\mathcal{F}_{\mathrm{OT}}}$ in the OT-hybrid model that statistically classical-UC-emulates $\mathcal{G}$. Thus, to get a protocol for $\mathcal{G}$ in the commitment-hybrid model, we simply replace all invocations to $\mathcal{F}_{\mathrm{OT}}$ by invocations of the subprotocol π_{QOT}, resulting in a protocol $\rho^{\pi_{\mathrm{QOT}}}$. We then expect that the security of $\rho^{\pi_{\mathrm{QOT}}}$ follows directly using the universal composition theorem (in its quantum variant). There is, however, one difficulty: To show that $\rho^{\pi_{\mathrm{QOT}}}$ statistically quantum-UC-emulates $\mathcal{G}$, the universal composition theorem requires that the following premises are fulfilled: π_{QOT} statistically quantum-UC-emulates $\mathcal{F}_{\mathrm{OT}}$, and $\rho^{\mathcal{F}_{\mathrm{OT}}}$ statistically quantum-UC-emulates $\mathcal{G}$. But from the result of Ishai et al. [13] we only have that $\rho^{\mathcal{F}_{\mathrm{OT}}}$ statistically *classical*-UC-emulates $\mathcal{G}$. Hence, we first have to show that the same result also holds with respect to quantum-UC-security. Fortunately, we do not have to revisit the proof of Ishai et al., because we show the following general fact:

Theorem 1 (Quantum lifting theorem – informal). *If the protocols π and ρ are classical protocols, and π statistically classical-UC-emulates ρ, then π statistically quantum-UC-emulates ρ.*

Combining this theorem with the universal composition theorem, we immediately get that $\rho^{\pi_{\mathrm{QOT}}}$ statistically quantum-UC-emulates $\mathcal{G}$. In other words, any multi-party computation can be performed securely using only a commitment and a quantum-channel. In contrast, we show that in the classical setting a commitment is not even sufficient to compute the AND-function.

We stress that a property like the quantum lifting theorem should not be taken for granted. For example, for the so-called stand-alone model as considered by Fehr and Schaffner [11], no corresponding property is known. A special case of security in the stand-alone model is the zero-knowledge property: The question whether protocols that are statistical zero-knowledge with respect to classical adversaries are also zero-knowledge with respect to quantum adversaries has been answered positively by Watrous [23] for particular protocols, but is still open in the general case.

1.1 How to Interpret Our Result

We show that we can perform arbitrary statistically UC-secure multi-party computations, given a quantum channel and a commitment. However, Mayers [16] has

shown that, even in the quantum setting, statistically secure commitment schemes do not exist, not even with respect to security notions much weaker than quantum-UC-security. In the light of this result, the reader may wonder whether our result is not vacuous. To illustrate why our result is useful even in the light of Mayers' impossibility result, we present four possible application scenarios.

Weaker computational assumptions. The first application of our result would be to combine our protocols with a commitment scheme that is only *computationally* quantum-UC-secure. Of course, the resulting multi-party computation protocol would then not be *statistically* secure any more. However, since commitment intuitively seems to be a simpler task than oblivious transfer, constructing a computationally quantum-UC-secure commitment scheme might be possible using simpler computational assumptions, and our result then implies that the same computational assumptions can be used for general multi-party computation.

Physical setup. One might seek a direct physical implementation of a commitment, such as a locked strongbox (or an equivalent but technologically more advanced construct). With our result, such a physical implementation would be sufficient for general multi-party computation. In contrast, in a classical setting one would be forced to try to find physical implementations of OT. It seems that a commitment might be a simpler physical assumption than OT (or at least an incomparable one). So our result reduces the necessary assumptions when implementing general multi-party computation protocols based on physical assumptions. Also, Kent [14] proposes to build commitments based on the fact that the speed of light is bounded. Although it is not clear whether his schemes are UC-secure (and in particular, how to model his physical assumptions in the UC framework), his ideas might lead to a UC-secure commitment scheme that then, using our result, gives general UC-secure multi-party computation based on the limitation of the speed of light.

Theoretical separation. Our result can also be seen from the purely theoretical point of view. It gives a separation between the quantum and the classical setting by showing that in the quantum setting, commitment is complete for general statistically secure multi-party computation, while in the classical world it is not. Such separations – even without practical applications – may increase our understanding of the relationship between the classical and the quantum setting and are therefore arguably interesting in their own right.

Long-term security. Müller-Quade and Unruh [17] introduce the concept of long-term UC-security. In a nutshell, long-term UC-security is a strengthening of computational UC-security that guarantees that a protocol stays secure even if the adversary gets unlimited computational power after the protocol execution. This captures the fact that, while we might confidently judge today's technology, we cannot easily make predictions about which computational problems will be hard in the future. Müller-Quade and Unruh show that (classically) long-term UC-secure commitment protocols exist given certain practical infrastructure assumptions, so-called signature cards. It is, however, likely that their results

cannot be extended to achieve general multi-party computation. Our result, on the other hand, might allow to overcome this limitation: Assume that we show that the commitment protocol of Müller-Quade and Unruh is also secure in a quantum variant of long-term UC-security. Then we could compose that commitment protocol with the protocols presented here, leading to long-term UC-secure general multi-party protocols from signature cards.

1.2 Related Work

Security models. General quantum security models based on the stand-alone model have first been proposed by van de Graaf [22]. His model comes without a composition theorem. The notion has been refined by Wehner and Wullschleger [24] and by Fehr and Schaffner [11] who also prove sequential composition theorems. Quantum security models in the style of the UC model have been proposed by Ben-Or and Mayers [3] and by Unruh [20]. The original idea behind the UC framework in the classical setting was independently discovered by Canetti [7] and by Pfitzmann and Waidner [19] (the notion is called Reactive Simulatability in the latter paper).

Quantum protocols. The idea of using quantum communication for cryptographic purposes seems to originate from Wiesner [25]. The idea gained widespread recognition with the BB84 quantum key-exchange protocol by Bennett and Brassard [4]. A statistically hiding and binding commitment scheme was proposed by Brassard, Crépeau, Jozsa, and Langlois [6]. Unfortunately, the scheme was later found to be insecure; in fact, Mayers [16] showed that statistically hiding and binding quantum commitments are impossible without using additional assumptions. Kent [14] circumvents this impossibility result by proposing a statistically hiding and binding commitment scheme that is based on the limitation of the speed of light. Bennett, Brassard, Crépeau, and Skubiszewska [5] present a protocol for statistically secure oblivious transfer in the quantum setting. They prove their protocol secure under the assumption that the adversary cannot store qubits and measures each qubit individually. They also sketch an extension that uses a commitment scheme to make their OT protocol secure against adversaries that can store and compute on quantum states. The protocol analyzed in the present paper is, in its basic idea, that extension. Yao [26] gave a partial proof of the extended OT protocol. His proof, however, is incomplete and refers to a future complete paper which, to the best of our knowledge, never appeared. As far as we know, the first complete proof of a variant of that OT protocol has been given by Damgård, Fehr, Lunemann, Salvail, and Schaffner [9]; their protocol is secure in the stand-alone model. Hofheinz and Müller-Quade [12] conjectured that the extended OT protocol by Bennett et al. [5] is indeed UC-secure; in the present paper we prove this claim. Damgård, Fehr, Salvail, and Schaffner [10] have presented OT and commitment protocols which are statistically secure under the assumption that the adversary has a bounded quantum storage capacity. [1] (extended abstract only) give a protocol for performing quantum-UC multi-party computation given an honest majority. Their protocol even allows to compute functions which have quantum output.

Classical vs. quantum security. To the best of our knowledge, van de Graaf [22] was the first to notice that even statistically secure classical protocols are not necessarily secure in a quantum setting. The reason is that the powerful technique of rewinding the adversary is not available in the quantum setting. Watrous [23] showed that in particular cases, a technique similar to classical rewinding can be used. He uses this technique to construct quantum zero-knowledge proofs. No general technique relating classical and quantum security is known; to the best of our knowledge, our quantum lifting theorem is the first such result (although restricted to the statistical UC model).

Miscellaneous. Kilian [15] first noted that OT is complete for general multiparty computation. Ishai, Prabhakaran, and Sahai [13] prove that this also holds in the UC setting. Computationally secure UC commitment schemes have been presented by Canetti and Fischlin [8].

1.3 Preliminaries

General. A nonnegative function μ is called negligible if for all $c > 0$ and all sufficiently large k, $\mu(k) < k^{-c}$. A nonnegative function f is called overwhelming if $f \geq 1 - \mu$ for some negligible μ. Keywords in typewriter font (e.g., `environment`) are assumed to be fixed but arbitrary distinct non-empty words in $\{0,1\}^*$. $\varepsilon \in \{0,1\}^*$ denotes the empty word. Given a sequence $x = x_1, \ldots, x_n$, and a set $I \subseteq \{1, \ldots, n\}$, $x_{|I}$ denote the sequence x restricted to the indices $i \in I$.

Quantum systems. We can only give a terse overview over the formalism used in quantum computing. For a thorough introduction, we recommend the textbook by Nielsen and Chuang [18, Chap. 1–2]. A (pure) state in a quantum system is described by a vector $|\psi\rangle$ in some Hilbert space $\mathcal{H}$. In this work, we only use Hilbert spaces of the form $\mathcal{H} = \mathbb{C}^N$ for some countable set N, usually $N = \{0,1\}$ for qubits or $N = \{0,1\}^*$ for bitstrings. We always assume a designated orthonormal basis $\{|x\rangle : x \in N\}$ for each Hilbert space, called the computational basis. The basis states $|x\rangle$ represent classical states (i.e., states without superposition). Given several separate subsystems $\mathcal{H}_1 = \mathbb{C}^{N_1}, \ldots, \mathcal{H}_n = \mathbb{C}^{N_n}$, we describe the joint system by the tensor product $\mathcal{H}_1 \otimes \cdots \otimes \mathcal{H}_n = \mathbb{C}^{N_1 \times \cdots \times N_n}$. We write $\langle\Psi|$ for the linear transformation mapping $|\Phi\rangle$ to the scalar product $\langle\Psi|\Phi\rangle$. Consequently, $|\Psi\rangle\langle\Psi|$ denotes the orthogonal projector on $|\Psi\rangle$. We set $|0\rangle_+ := |0\rangle$, $|1\rangle_+ := |1\rangle$, $|0\rangle_\times := \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, and $|1\rangle_\times := \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. For $x \in \{0,1\}^n$ and $\theta \in \{+,\times\}^n$, we define $|x\rangle_\theta := |x_1\rangle_{\theta_1} \otimes \cdots \otimes |x_n\rangle_{\theta_n}$.

Mixed states. If a system is not in a single pure state, but instead is in the pure state $|\Psi_i\rangle \in \mathcal{H}$ with probability p_i (i.e., it is in a mixed state), we describe the system by a density operator $\rho = \sum_i p_i |\Psi_i\rangle\langle\Psi_i|$ over $\mathcal{H}$. This representation contains all physically observable information about the distribution of states, but some distributions are not distinguishable by any measurement and thus are represented by the same mixed state. The set of all density operators is the set of all positive[4] operators $\mathcal{H}$ with trace 1, and is denoted $\mathcal{P}(\mathcal{H})$. Composed systems

[4] We call an operator positive if it is Hermitean and has only nonnegative eigenvalues.

are descibed by operators in $\mathcal{P}(\mathcal{H}_1 \otimes \cdots \otimes \mathcal{H}_n)$. In the following, when speaking about (quantum) states, we always mean mixed states in the density operator representation. A mapping $\mathcal{E} : \mathcal{P}(\mathcal{H}_1) \to \mathcal{P}(\mathcal{H}_2)$ represents a physically possible operation (realizable by a sequence of unitary transformations, measurements, and initializations and removals of qubits) iff it is a completely positive trace preserving map.[5] We call such mappings superoperators. The superoperator $\mathcal{E}^m_{init}$ on $\mathcal{P}(\mathcal{H})$ with $\mathcal{H} := \mathbb{C}^{\{0,1\}^*}$ and $m \in \{0,1\}^*$ is defined by $\mathcal{E}^m_{init}(\rho) := |m\rangle\langle m|$ for all ρ.

Composed systems. Given a superoperator $\mathcal{E}$ on $\mathcal{P}(\mathcal{H}_1)$, the superoperator $\mathcal{E} \otimes id$ operates on $\mathcal{P}(\mathcal{H}_1 \otimes \mathcal{H}_2)$. Instead of saying "we apply $\mathcal{E} \otimes id$", we say "we apply $\mathcal{E}$ to $\mathcal{H}_1$". If we say "we initialize $\mathcal{H}$ with m", we mean "we apply $\mathcal{E}^m_{init}$ to $\mathcal{H}$". Given a state $\rho \in \mathcal{P}(\mathcal{H}_1 \otimes \mathcal{H}_2)$, let $\rho_x := (|x\rangle\langle x| \otimes id)\rho(|x\rangle\langle x| \otimes id)$. Then the outcome of measuring $\mathcal{H}_1$ in the computational basis is x with probability $\operatorname{tr} \rho_x$, and after measuring x, the quantum state is $\frac{\rho_x}{\operatorname{tr} \rho_x}$. Since we will only perform measurements in the computational basis in this work, we will omit the qualification "in the computational basis". The terminology in this paragraph generalizes to systems composed of more than two subsystems.

Classical states. Classical probability distributions $P : N \to [0,1]$ over a countable set N are represented by density operators $\rho \in \mathcal{P}(\mathbb{C}^N)$ with $\rho = \sum_{x \in N} P(x)|x\rangle\langle x|$ where $\{|x\rangle\}$ is the computational basis. We call a state classical if it is of this form. We thus have a canonical isomorphism between the classical states over $\mathbb{C}^N$ and the probability distributions over N. We call a superoperator $\mathcal{E} : \mathcal{P}(\mathbb{C}^{N_1}) \to \mathcal{P}(\mathbb{C}^{N_2})$ classical iff if there is a randomized function $F : N_1 \to N_2$ such that $\mathcal{E}(\rho) = \sum_{x \in N_1, y \in N_2} \Pr[F(x) = y] \cdot \langle x|\rho|x\rangle \cdot |y\rangle\langle y|$. Classical superoperators describe what can be realized with classical computations. An example of a classical superoperator on $\mathcal{P}(\mathbb{C}^N)$ is $\mathcal{E}_{class} : \rho \mapsto \sum_x \langle x|\rho|x\rangle \cdot |x\rangle\langle x|$. Intuitively, $\mathcal{E}_{class}$ measures ρ in the computational basis and then discards the outcome, thus removing all superpositions from ρ.

2 Quantum Universal Composability

We now present our quantum-UC-framework. The basic idea of our definition is the same as that underlying Canetti's UC-framework [7]. The main change is that we allow all machines to perform quantum computations and to send quantum states as messages. For a gentler introduction into the ideas and intuitions underlying the UC-framework, we refer to [7].

Machine model. A machine M is described by an identity id_M in $\{0,1\}^*$ and a sequence of superoperators $\mathcal{E}^{(k)}_M$ ($k \in \mathbb{N}$) on $\mathcal{H}^{state} \otimes \mathcal{H}^{class} \otimes \mathcal{H}^{quant}$ with $\mathcal{H}^{state}, \mathcal{H}^{class}, \mathcal{H}^{quant} := \mathbb{C}^{\{0,1\}^*}$ (the *state transition operators*). The index k in $\mathcal{E}^{(k)}_M$ denotes the security parameter. The Hilbert space $\mathcal{H}^{state}$ represents the state kept by the machine between invocations, and $\mathcal{H}^{class}$ and $\mathcal{H}^{quant}$ are used

[5] A map $\mathcal{E}$ is completely positive iff for all Hilbert spaces $\mathcal{H}'$, and all positive operators ρ on $\mathcal{H}_1 \otimes \mathcal{H}'$, $(\mathcal{E} \otimes id)(\rho)$ is positive.

both for incoming and outgoing messages. Any message consists of a classical part stored in $\mathcal{H}^{class}$ and a quantum part stored in $\mathcal{H}^{quant}$. If a machine id_{sender} wishes to send a message with classical part m and quantum part $|\Psi\rangle$ to a machine id_{rcpt}, the machine id_{sender} initializes $\mathcal{H}^{class}$ with $(id_{sender}, id_{rcpt}, m)$ and $\mathcal{H}^{quant}$ with $|\Psi\rangle$. (See the definition of the network execution below for details.) The separation of messages into a classical and a quantum part is for clarity only, all information could also be encoded directly in a single register. If a machine does not wish to send a message, it initializes $\mathcal{H}^{class}$ and $\mathcal{H}^{quant}$ with ε.

A network $\mathbf{N}$ is a set of machines with pairwise distinct identities containing a machine $\mathcal{Z}$ with $id_{\mathcal{Z}} = \texttt{environment}$. We write $ids_{\mathbf{N}}$ for the set of the identities of the machines in $\mathbf{N}$.

We call a machine M quantum-polynomial-time if there is a uniform[6] sequence of quantum circuits C_k such that for all k, the circuit C_k implements the superoperator $\mathcal{E}_M^{(k)}$.

Network execution. The state space $\mathcal{H}_{\mathbf{N}}$ of a network $\mathbf{N}$ is defined as $\mathcal{H}_{\mathbf{N}} := \mathcal{H}^{class} \otimes \mathcal{H}^{quant} \otimes \bigotimes_{id \in ids_{\mathbf{N}}} \mathcal{H}_{id}^{state}$ with $\mathcal{H}_{id}^{state}, \mathcal{H}^{class}, \mathcal{H}^{quant} := \mathbb{C}^{\{0,1\}^*}$. Here $\mathcal{H}_{id}^{state}$ represents the local state of the machine with identity id and $\mathcal{H}^{class}$ and $\mathcal{H}^{quant}$ represent the state spaces used for communication. ($\mathcal{H}^{class}$ and $\mathcal{H}^{quant}$ are shared between all machines. Since only one machine is active at a time, no conflicts occur.)

A step in the execution of $\mathbf{N}$ is defined by a superoperator $\mathcal{E} := \mathcal{E}_{\mathbf{N}}^{(k)}$ operating on $\mathcal{H}_{\mathbf{N}}$. This superoperator performs the following steps: First, $\mathcal{E}$ measures $\mathcal{H}^{class}$ in the computational basis and parses the outcome as $(id_{sender}, id_{rcpt}, m)$. Let M be the machine in $\mathbf{N}$ with identity id_{rcpt}. Then $\mathcal{E}$ applies $\mathcal{E}_M^{(k)}$ to $\mathcal{H}_{id_{rcpt}}^{state} \otimes \mathcal{H}^{class} \otimes \mathcal{H}^{quant}$. Then $\mathcal{E}$ measures $\mathcal{H}^{class}$ and parses the outcome as $(id'_{sender}, id'_{rcpt}, m')$. If the outcome could not be parsed, or if $id'_{sender} \neq id_{rcpt}$, initialize $\mathcal{H}^{class}$ with $(\varepsilon, \texttt{environment}, \varepsilon)$ and $\mathcal{H}^{quant}$ with ε. (This ensures that the environment is activated if a machine sends no or an ill-formed message.)

The output of the network $\mathbf{N}$ on input z and security parameter k is described by the following algorithm: Let $\rho \in \mathcal{P}(\mathcal{H}_{\mathbf{N}})$ be the state that is initialized to $(\varepsilon, \texttt{environment}, z)$ in $\mathcal{H}^{class}$, and to the empty word ε in all other registers. Then repeat the following indefinitely: Apply $\mathcal{E}_{\mathbf{N}}^{(k)}$ to ρ. Measure $\mathcal{H}^{class}$. If the outcome is of the form $(\texttt{environment}, \varepsilon, out)$, return out and terminate. Otherwise, continue the loop. The probability distribution of the return value out is denoted by $\mathrm{Exec}_{\mathbf{N}}(k, z)$.

Corruptions. To model corruptions, we introduce *corruption parties*, special machines that follow the instructions given by the adversary. When invoked, the corruption party P_{id}^C with identity id measures $\mathcal{H}^{class}$ and parses the outcome as $(id_{sender}, id_{rcpt}, m)$. If $id_{sender} = \texttt{adversary}$, $\mathcal{H}^{class}$ is initialized with m. (In this case, m specifies both the message and the sender/recipient. Thus the

[6] A sequence of circuits C_k is uniform if a deterministic Turing machine can output the description of C_k in time polynomial in k.

adversary can instruct a corruption party to send to arbitrary recipients.) Otherwise, $\mathcal{H}^{class}$ is initialized with $(id, \texttt{adversary}, (id_{sender}, id_{rcpt}, m))$. (The message is forwarded to the adversary.) Note that, since P^C_{id} does not touch the $\mathcal{H}^{quant}$, the quantum part of the message is forwarded. Given a network $\mathbf{N}$, and a set of identities C, we write $\mathbf{N}^C$ for the set resulting from replacing each machine $M \in \mathbf{N}$ with identity $id \in C$ by P^C_{id}.

Security model. A protocol π is a set of machines with `environment`, `adversary` $\notin ids(\pi)$. We assume a set of identities $parties_\pi \subseteq ids(\pi)$ to be associated with π. $parties_\pi$ denotes which of the machines in the protocol are actually protocol parties (as opposed to incorruptible entities such as ideal functionalities).

An *environment* is a machine with identity `environment`, an *adversary* or a *simulator* is a machine with identity `adversary` (there is no formal distinction between adversaries and simulators, the terms refer to different intended roles of a machine). We call two networks $\mathbf{N}, \mathbf{N}'$ *indistinguishable* if there is a negligible function μ such that for all $z \in \{0,1\}^*$ and $k \in \mathbb{N}$, $|\Pr[\text{Exec}_{\mathbf{N}}(k,z) = 1] - \Pr[\text{Exec}_{\mathbf{N}'}(k,z) = 1]| \leq \mu(k)$. We speak of *perfect indistinguishability* if $\mu = 0$.

Definition 2 (Statistical quantum-UC-security). *Let protocols π and ρ be given. We say π* statistically quantum-UC-emulates *ρ iff for every set $C \subseteq parties_\pi$ and for every adversary* Adv *there is a simulator* Sim *such that for every environment $\mathcal{Z}$, the networks $\pi^C \cup \{\text{Adv}, \mathcal{Z}\}$ (called the real model) and $\rho^C \cup \{\text{Sim}, \mathcal{Z}\}$ (called the ideal model) are indistinguishable. We furthermore require that if* Adv *is quantum-polynomial-time, so is* Sim.

Definition 3 (Computational quantum-UC-security). *Let protocols π and ρ be given. We say π* computationally quantum-UC-emulates *ρ iff for every set $C \subseteq parties_\pi$ and for every quantum-polynomial-time adversary* Adv *there is a quantum-polynomial-time simulator* Sim *such that for every quantum-polynomial-time environment $\mathcal{Z}$, the networks $\pi^C \cup \{\text{Adv}, \mathcal{Z}\}$ and $\rho^C \cup \{\text{Sim}, \mathcal{Z}\}$ are indistinguishable.*

Note that although $\text{Exec}_{\pi^C \cup \{\text{Adv}, \mathcal{Z}\}}(k,z)$ may return arbitrary bitstrings, we only compare whether the return value of $\mathcal{Z}$ is 1 or not. This effectively restricts $\mathcal{Z}$ to returning a single bit. This can be done without loss of generality (see [7] for a discussion of this issue; their arguments also apply to the quantum case) and simplifies the definition.

In our framework, any communication between two parties is perfectly secure since the network model guarantees that they are delivered to the right party and not leaked to the adversary. To model a protocol with insecure channels instead, one would explicitly instruct the protocol parties to send all messages through the adversary. Authenticated channels can be realized by introducing an ideal functionality (see the next section) that realizes an authenticated channel. For simplicity, we only consider protocols with secure channels in this work.

Ideal functionalities. In most cases, the behavior of the ideal model is described by a single machine $\mathcal{F}$, the so-called ideal functionality. We can think

of this functionality as a trusted third party that perfectly implements the desired protocol behavior. For example, the functionality $\mathcal{F}_{\mathrm{OT}}$ for oblivious transfer would take as input from Alice two bitstrings m_0, m_1, and from Bob a bit c, and send to Bob the bitstring m_c. Obviously, such a functionality constitutes a secure oblivious transfer. We can thus define a protocol π to be a secure OT protocol if π quantum-UC-emulates $\mathcal{F}_{\mathrm{OT}}$ where $\mathcal{F}_{\mathrm{OT}}$ denotes the protocol consisting only of one machine, the functionality $\mathcal{F}_{\mathrm{OT}}$ itself. There is, however, one technical difficulty here. In the real protocol π, the bitstring m_c is sent to the environment $\mathcal{Z}$ by Bob, while in the ideal model, m_c is sent by the functionality. Since every message is tagged with the sender of that message, $\mathcal{Z}$ can distinguish between the real and the ideal model merely by looking at the sender of m_c. To solve this issue, we need to ensure that $\mathcal{F}$ sends the message m_c in the name of Bob (and for analogous reasons, that $\mathcal{F}$ receives messages sent by $\mathcal{Z}$ to Alice or Bob). To achieve this, we use so-called dummy-parties [7] in the ideal model. These are parties with the identities of Alice and Bob that just forward messages between the functionality and the environment.

Definition 4 (Dummy-party). *Let a machine P and a functionality $\mathcal{F}$ be given. The dummy-party $\tilde{P}$ for P and $\mathcal{F}$ is a machine that has the same identity as P and has the following state transition operator: Let $id_{\mathcal{F}}$ be the identity of $\mathcal{F}$. When activated, measure $\mathcal{H}^{class}$. If the outcome of the measurement is of the form $(\texttt{environment}, id_P, m)$, initialize $\mathcal{H}^{class}$ with $(id_P, id_{\mathcal{F}}, m)$. If the outcome is of the form $(id_{\mathcal{F}}, id_P, m)$, initialize $\mathcal{H}^{class}$ with $(id_P, \texttt{environment}, m)$. In all cases, the quantum communication register is not modified (i.e., the message in that register is forwarded).*

Note the strong analogy to the corruption parties (page 494).

Thus, if we write π quantum-UC-emulates $\mathcal{F}$, we mean that π quantum-UC-emulates $\rho_{\mathcal{F}}$ where $\rho_{\mathcal{F}}$ consists of the functionality $\mathcal{F}$ and the dummy-parties corresponding to the parties in π. More precisely:

Definition 5. *Let π be a protocol and $\mathcal{F}$ be a functionality. We say that π statistically/computationally quantum-UC-emulates $\mathcal{F}$ if π statistically/computationally quantum-UC-emulates $\rho_{\mathcal{F}}$ where $\rho_{\mathcal{F}} := \{\tilde{P} : P \in parties_{\pi}\} \cup \{\mathcal{F}\}$.*

For more discussion of dummy-parties and functionalities, see [7].

Using the concept of an ideal functionality, we can specify a range of protocol tasks by simply defining the corresponding functionality. Below, we give the definitions of various functionalities. All these functionalities are classical, we therefore do not explicitly describe when the registers $\mathcal{H}^{class}$ and $\mathcal{H}^{quant}$ are measured/initialized but instead describe the functionality in terms of the messages sent and received.

Definition 6 (Commitment). *Let A and B be two parties. The functionality $\mathcal{F}_{\mathrm{COM}}^{B\to A,\ell}$ behaves as follows: Upon (the first) input $(\texttt{commit}, x)$ with $x \in \{0,1\}^{\ell(k)}$ from B, send $\texttt{committed}$ to A. Upon input $\texttt{open}$ from B send $(\texttt{open}, x)$ to A. All communication/input/output is classical. We call B the sender and A the recipient.*

Definition 7 (Oblivious transfer (OT)). *Let A and B be two parties. The functionality $\mathcal{F}_{\mathrm{OT}}^{A\to B,\ell}$ behaves as follows: When receiving input (s_0, s_1) from A with $s_0, s_1 \in \{0,1\}^{\ell(k)}$ and $c \in \{0,1\}$ from B, send $s := s_c$ to B. All communication/input/output is classical. We call A the sender and B the recipient.*[7]

Definition 8 (Randomized oblivious transfer (ROT)). *Let A and B be two parties. The functionality $\mathcal{F}_{\mathrm{ROT}}^{A\to B,\ell}$ behaves as follows: If A is uncorrupted, when receiving input $c \in \{0,1\}$ from B, choose $s_0, s_1 \in \{0,1\}^{\ell(k)}$ uniformly and send (s_0, s_1) to A and $s := s_c$ to B. If A is corrupted, when receiving input (s_0, s_1) from A with $s_0, s_1 \in \{0,1\}^{\ell(k)}$ and $c \in \{0,1\}$ from B, send $s := s_c$ to B. All communication/input/output is classical.*

Dummy-adversary. In the definition of UC-security, we have three entities interacting with the protocol: the adversary, the simulator, and the environment. Both the adversary and the environment are all-quantified, hence we would expect that they do, in some sense, work together. This intuition is backed by the following fact which was first noted by Canetti [7]: Without loss of generality, we can assume an adversary that is completely controlled by the environment. This so-called dummy-adversary only forwards messages between the environment and the protocol. The actual attack is then executed by the environment.

Definition 9 (Dummy-adversary Adv_{dummy}). *When activated, the dummy-adversary Adv_{dummy} measures $\mathcal{H}^{class}$; call the outcome m. If m is of the form $(\mathtt{environment}, \mathtt{adversary}, m')$, initialize $\mathcal{H}^{class}$ with m'. Otherwise initialize $\mathcal{H}^{class}$ with $(\mathtt{adversary}, \mathtt{environment}, m)$. In all cases, the quantum communication register is not modified (i.e., the message in that register is forwarded).*

Note the strong analogy to the dummy-parties (Definition 4) and the corruption parties (page 494).

Lemma 10 (Completeness of the dummy-adversary). *Assume that π quantum-UC-emulates ρ with respect to the dummy-adversary (i.e., instead of quantifying over all adversaries Adv, we fix $\mathrm{Adv} := \mathrm{Adv}_{dummy}$). Then π quantum-UC-emulates ρ. This holds both for statistical and computational quantum-UC-security.*

The proof of Lemma 10 is very similar to that given in [7] and given in the full version [21].

Universal composition. For some protocol σ, and some protocol π, by σ^π we denote the protocol where σ invokes (up to polynomially many) instances of π. That is, in σ^π the machines from σ and from π run together in one network, and the machines from σ access the inputs and outputs of π. (That is, σ plays the role of the environment from the point of view of π. In particular, $\mathcal{Z}$ then

[7] We used A as the sender in the description of the OT functionality, and as the recipient in the description of the commitment functionality. We do so to simplify notation later; our protocol for OT from A to B will use a commitment from B to A.

talks only to σ and not to the subprotocol π directly.) A typical situation would be that $\sigma^{\mathcal{F}}$ is some protocol that makes use of some ideal functionality $\mathcal{F}$, say a commitment functionality, and then σ^{π} would be the protocol resulting from implementing that functionality with some protocol π, say a commitment protocol. (We say that $\sigma^{\mathcal{F}}$ is a protocol in the $\mathcal{F}$-hybrid model.) One would hope that such an implementation results in a secure protocol σ^{π}. That is, we hope that if π quantum-UC-emulates $\mathcal{F}$ and $\sigma^{\mathcal{F}}$ quantum-UC-emulates $\mathcal{G}$, then σ^{π} quantum-UC-emulates $\mathcal{G}$. Fortunately, this is the case:

Theorem 11 (Universal Composition Theorem). *Let π, ρ, and σ be quantum-polynomial-time protocols. Assume that π quantum-UC-emulates ρ. Then σ^{π} quantum-UC-emulates σ^{ρ}. This holds both for statistical and computational quantum-UC-security.*

If we additionally have that σ quantum-UC-emulates $\mathcal{G}$, from the transitivity of quantum-UC-emulation (shown in the full version [21]), it immediately follows that σ^{π} quantum-UC-emulates $\mathcal{G}$.

The proof of Theorem 11 is very similar to that given in [7] and given in the full version [21].

3 Relating Classical and Quantum-UC

We call a machine classical if its state transition operator is classical. A protocol is classical if all its machines are classical.

Using this definition we can reformulate the definition of statistical classical UC in our framework.

Definition 12 (Statistical classical-UC-security). *Let protocols π and ρ be given. We say π* statistically classical-UC-emulates *ρ iff for every set $C \subseteq \text{parties}_{\pi}$ and for every classical adversary* Adv *there is a classical simulator* Sim *such that for every classical environment $\mathcal{Z}$, $\pi^{C} \cup \{\text{Adv}, \mathcal{Z}\}$ and $\rho^{C} \cup \{\text{Sim}, \mathcal{Z}\}$ are indistinguishable. We furthermore require that if* Adv *is probabilistic-polynomial-time, so is* Sim.

Note that classical statistical UC is essentially the same as the notion of statistical UC-security defined by Canetti [7]. Thus, known results for statistical UC-security carry over to the setting of Definition 12.

The next theorem guarantees that if a classical protocol is statistically classical UC-secure, then it is also statistically quantum-UC-secure. This allows, e.g., to first prove the security of a protocol in the (usually much simpler) classical setting, and then to compose it with quantum protocols using the universal composition theorem (Theorem 11).

Theorem 13 (Quantum lifting theorem). *Let π and ρ be classical protocols. Assume that π statistically classical-UC-emulates ρ. Then π statistically quantum-UC-emulates ρ.*

Proof. Given a machine M, let $\mathcal{C}(M)$ denote the machine which behaves like M, but measures incoming messages in the computational basis before processing them, and measures outgoing messages in the computational basis. More precisely, the superoperator $\mathcal{E}^{(k)}_{\mathcal{C}(M)}$ first invokes $\mathcal{E}_{class}$ on $\mathcal{H}^{class} \otimes \mathcal{H}^{quant}$, then invokes $\mathcal{E}^{(k)}_{M}$ on $\mathcal{H}^{state} \otimes \mathcal{H}^{class} \otimes \mathcal{H}^{quant}$, and then again invokes $\mathcal{E}_{class}$ on $\mathcal{H}^{class} \otimes \mathcal{H}^{quant}$. Since it is possible to simulate quantum Turing machines on classical Turing machines (with an exponential overhead), for every machine M, there exists a classical machine M' such that $\mathcal{C}(M)$ and M' are perfectly indistinguishable.[8]

We define the classical dummy-adversary $\mathrm{Adv}^{class}_{dummy}$ to be the classical machine that is defined like Adv_{dummy} (Definition 9), except that in each invocation, it first measures $\mathcal{H}^{class}$, $\mathcal{H}^{quant}$, and $\mathcal{H}^{state}$ in the computational basis (i.e., it applies $\mathcal{E}_{class}$ to $\mathcal{H}^{state} \otimes \mathcal{H}^{class} \otimes \mathcal{H}^{quant}$) and then proceeds as does Adv_{dummy}. Note that $\mathrm{Adv}^{class}_{dummy}$ is probabilistic-polynomial-time.

By Lemma 10, we only need to show that for any set C of corrupted parties, there exists a quantum-polynomial-time machine Sim such that for every machine $\mathcal{Z}$ the real model $\pi^C \cup \{\mathcal{Z}, \mathrm{Adv}_{dummy}\}$ and the ideal model $\rho^C \cup \{\mathcal{Z}, \mathrm{Sim}\}$ are indistinguishable.

The protocol π is classical, thus π^C is classical, too, and thus all messages forwarded by Adv_{dummy} from π^C to $\mathcal{Z}$ have been measured in the computational basis by π^C, and all messages forwarded by Adv_{dummy} from $\mathcal{Z}$ to π^C will be measured by π^C before being used. Thus, if Adv would additionally measure all messages it forwards in the computational basis, the view of $\mathcal{Z}$ would not be modified. More formally, $\pi^C \cup \{\mathcal{Z}, \mathrm{Adv}_{dummy}\}$ and $\pi^C \cup \{\mathcal{Z}, \mathrm{Adv}^{class}_{dummy}\}$ are perfectly indistinguishable. Furthermore, since both π^C and $\mathrm{Adv}^{class}_{dummy}$ measure all messages upon sending and receiving, $\pi^C \cup \{\mathcal{Z}, \mathrm{Adv}^{class}_{dummy}\}$ and $\pi^C \cup \{\mathcal{C}(\mathcal{Z}), \mathrm{Adv}^{class}_{dummy}\}$ are perfectly indistinguishable. Since it is possible to simulate quantum machines on classical machines (with an exponential overhead), there exists a classical machine $\mathcal{Z}'$ that is perfectly indistinguishable from $\mathcal{C}(\mathcal{Z})$. Then $\pi^C \cup \{\mathcal{C}(\mathcal{Z}), \mathrm{Adv}^{class}_{dummy}\}$ and $\pi^C \cup \{\mathcal{Z}', \mathrm{Adv}^{class}_{dummy}\}$ are perfectly indistinguishable. Since $\mathrm{Adv}^{class}_{dummy}$ and $\mathcal{Z}'$ are classical and $\mathrm{Adv}^{class}_{dummy}$ is polynomial-time, there exists a classical probabilistic-polynomial-time simulator Sim (whose construction is independent of $\mathcal{Z}'$) such that $\pi^C \cup \{\mathcal{Z}', \mathrm{Adv}^{class}_{dummy}\}$ and $\rho^C \cup \{\mathcal{Z}', \mathrm{Sim}\}$ are indistinguishable.

Then $\rho^C \cup \{\mathcal{Z}', \mathrm{Sim}\}$ and $\rho^C \cup \{\mathcal{C}(\mathcal{Z}), \mathrm{Sim}\}$ are perfectly indistinguishable by construction of $\mathcal{Z}'$. And since both ρ^C and Sim measure all messages they send and receive, $\rho^C \cup \{\mathcal{C}(\mathcal{Z}), \mathrm{Sim}\}$ and $\rho^C \cup \{\mathcal{Z}, \mathrm{Sim}\}$ are perfectly indistinguishable.

Summarizing, we have that $\pi^C \cup \{\mathcal{Z}, \mathrm{Adv}_{dummy}\}$ and $\rho^C \cup \{\mathcal{Z}, \mathrm{Sim}\}$ are indistinguishable for all quantum-polynomial-time environments $\mathcal{Z}$. Furthermore, Sim is classical probabilistic-polynomial-time and hence quantum-polynomial-time and its construction does not depend on the choice of $\mathcal{Z}$. Thus π statistically quantum-UC-emulates ρ. □

[8] More precisely, for any set of machines N, the networks $N \cup \{M\}$ and $N \cup \{\mathcal{C}(M)\}$ are perfectly indistinguishable.

Parameters: Integers n, $m > n$, ℓ, a family $\mathbf{F}$ of universal hash functions.
Parties: The sender Alice and the recipient Bob.
Inputs: Alice gets no input, Bob gets a bit c.

1. Alice chooses $\tilde{x}^A \in \{0,1\}^m$ and $\tilde{\theta}^A \in \{+,\times\}^m$ and sends $|\tilde{x}^A\rangle_{\tilde{\theta}^A}$ to Bob.
2. Bob receives the state $|\Psi\rangle$ sent by the sender. Then Bob chooses $\tilde{\theta}^B \in \{+,\times\}^m$ and measures the qubits of $|\Psi\rangle$ in the bases $\tilde{\theta}^B$. Call the result $\tilde{x}^B$.
3. For each i, Bob commits to $\tilde{\theta}^B_i$ and $\tilde{x}^B_i$ using one instance of $\mathcal{F}_{\mathrm{COM}}^{B\to A,1}$ each.
4. Alice chooses a set $T \subseteq \{1,\dots,m\}$ of size $m-n$ and sends T to Bob.
5. Bob opens the commitments of $\tilde{\theta}^B_i$ and $\tilde{x}^B_i$ for all $i \in T$.
6. Alice checks $\tilde{x}^A_i = \tilde{x}^B_i$ for all i with $i \in T$ and $\tilde{\theta}^A_i = \tilde{\theta}^B_i$. If this test fails, Alice aborts.
7. Let x^A be the n-bit string resulting from removing the bits at positions $i \in T$ from $\tilde{x}^A$. Define θ^A, x^B, and θ^B analogously.
8. Alice sends θ^A to Bob.
9. Bob sets $I_c := \{i : \theta^A_i = \theta^B_i\}$ and $I_{1-c} := \{i : \theta^A_i \neq \theta^B_i\}$. Then Bob sends (I_0, I_1) to Alice.
10. Alice chooses $s_0, s_1 \in \{0,1\}^{\ell(k)}$ and $f_0, f_1 \in \mathbf{F}$, output (s_0, s_1), and computes $m_j := s_j \oplus f_j(x^A|_{I_j})$ for $j = 0, 1$. Then Alice sends f_0, f_1, m_0, m_1 to Bob.
11. Bob outputs $s := m_c \oplus f_c(x^B|_{I_c})$.

Fig. 1. Protocol π_{QROT} for randomized oblivious transfer

4 Oblivious Transfer

Definition 14 (OT protocols). *The protocol π_{QROT} is defined in Figure 1. Fix a commitment scheme* com*. The protocol $\pi^{\mathrm{com}}_{\mathrm{QROT}}$ is defined like π_{QROT}, but instead of using the functionality $\mathcal{F}_{\mathrm{COM}}$, the commitment scheme* com *is used. The protocol π_{QOT} is defined like π_{QROT}, with the following modifications: Alice takes as input two $\ell(k)$-bit strings v_0, v_1. In Step 10, Alice additionally sends t_0, t_1 with $t_i := s_i \oplus v_i$. Bob outputs $s \oplus t_c$ instead of s in Step 11.*

We first analyze π_{QROT} and will then deduce the security of π_{QOT} from that of π_{QROT}.

4.1 Corrupted Alice

Lemma 15. *The protocol π_{QROT} statistically quantum-UC-emulates $\mathcal{F}_{\mathrm{ROT}}^{A\to B,\ell}$ in the case of corrupted Alice.*

Proof. First, we describe the structure of the real and ideal model in the case that the party A (Alice) is corrupted:

In the real model, we have the environment $\mathcal{Z}$, the adversary Adv, the corruption party A^C, the honest party B (Bob), and the $2m$ instances of the commitment functionality $\mathcal{F}_{\mathrm{COM}}$. The adversary controls the corruption party A^C, so effectively he controls the communication with Bob and the inputs of $\mathcal{F}_{\mathrm{COM}}$. Bob's input (a choice bit c) is chosen by the environment, and the environment also gets Bob's output (a bitstring $s \in \{0,1\}^\ell$). See Figure 2(a).

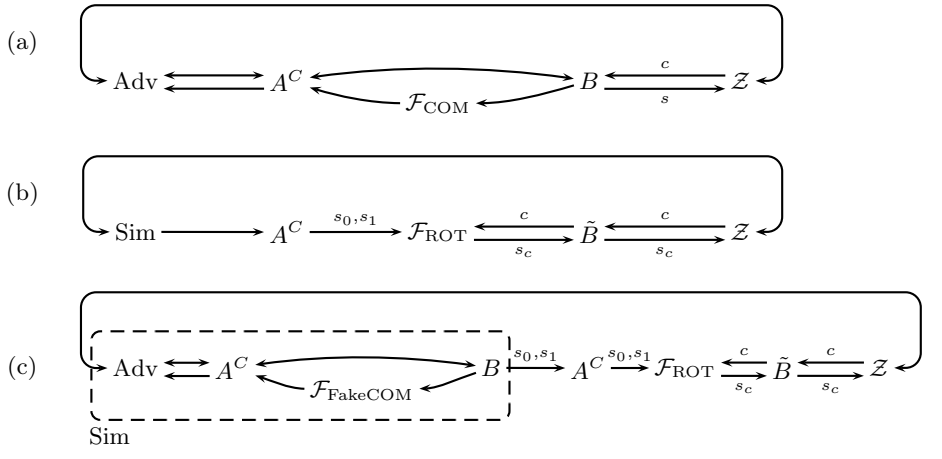

Fig. 2. Networks occurring in the proof of Lemma 15. The dashed box represents the machine Sim that internally simulates Adv, A^C, $\mathcal{F}_{\mathrm{FakeCOM}}$ and B.

In the ideal model, we have the environment $\mathcal{Z}$, the simulator Sim (to be defined below), the corruption party A^C, the dummy-party $\tilde{B}$, and the randomized OT functionality $\mathcal{F}_{\mathrm{ROT}}$. The simulator Sim controls the corruption party A^C and hence effectively chooses the inputs s_0, s_1 of $\mathcal{F}_{\mathrm{ROT}}$.[9] The input c of $\mathcal{F}_{\mathrm{ROT}}$ is chosen by the dummy-party $\tilde{B}$ and thus effectively by the environment $\mathcal{Z}$. The output $s := s_c$ of $\mathcal{F}_{\mathrm{ROT}}$ is given to the dummy-party $\tilde{B}$ and thus effectively to the environment $\mathcal{Z}$. See Figure 2(b).

To show Lemma 15, we need to find a simulator Sim such that, for any environment $\mathcal{Z}$, the real model and the ideal model are indistinguishable. To do so, we start with the real model, and change the machines in the real model step-by-step until we end up with the ideal model containing a suitable simulator Sim (which we define below in the description of Game 6). In each step, we show that network before and after the step are perfectly indistinguishable.

Game 1. We replace $\mathcal{F}_{\mathrm{COM}}$ by a commitment functionality $\mathcal{F}_{\mathrm{FakeCOM}}$ in which Bob (the sender) can cheat. That is, in the commit phase, $\mathcal{F}_{\mathrm{FakeCOM}}$ expects a message `commit` from B (instead of $(\texttt{commit}, x)$), and in the open phase, $\mathcal{F}_{\mathrm{FakeCOM}}$ expects a message $(\texttt{open}, x)$ (instead of `open`) and then sends $(\texttt{open}, x)$ to Alice. We also change Bob's implementation accordingly, i.e., when Bob should commit to a bit b, he stores that bit b and gives it to $\mathcal{F}_{\mathrm{FakeCOM}}$ when opening the commitment. Obviously, this change leads to a perfectly indistinguishable network (since Bob still opens the commitment in the same way).

Game 2. Since Bob uses $\mathcal{F}_{\mathrm{FakeCOM}}$ instead of $\mathcal{F}_{\mathrm{COM}}$, he does not use the outcomes $\tilde{x}_i^B$ of his measurements before Step 5 (for $i \in T$) or Step 11 (for $i \notin T$) of the protocol. Thus, we modify Bob so that he performs the measurements

[9] Remember that, if Alice is corrupted, $\mathcal{F}_{\mathrm{ROT}}$ behaves like $\mathcal{F}_{\mathrm{OT}}$ and takes inputs s_0, s_1 from Alice.

with outcomes $\tilde{x}_i^B$ ($i \in T$) in Step 5 (in particular, after learning T), and the measurements with outcomes x_i^B in Step 11. Delaying the measurements leads to a perfectly indistinguishable network.

Game 3. The bits x_i^B with $i \in I_{1-c}$ are never used by Bob. Thus we can modify Bob to use the bases θ_i^A instead of θ_i^B for these bits without changing the output of $\mathcal{Z}$. Furthermore, since $\theta_i^A = \theta_i^B$ for $i \in I_c$, we can modify Bob to also use the bases θ_i^A instead of θ_i^B when measuring x_i^B with $i \in I_c$. Summarizing, we modify Bob to use θ^A instead of θ^B, and we get a perfectly indistinguishable network.

Game 4. The bases θ^B are chosen randomly by Bob, and they are only used to compute the sets I_0 and I_1. We change Bob to instead pick (I_0, I_1) as a random partition of $\{1, \dots, n\}$. Since this leads to the same distribution of (I_0, I_1) and since θ^B is not used elsewhere, this leads to a perfectly indistinguishable network.

Game 5. In Step 11, we change Bob to compute $s_i := m_i \oplus f_i(x^B|_{I_i})$ for $i = 0, 1$ and to output $s := s_c$. This leads to the same value of s as the original computation $s := m_c \oplus f_c(x^B|_{I_c})$, hence the resulting network is perfectly indistinguishable from the previous one. Note that now, Bob only uses the choice bit c to pick which of the two values s_0, s_1 to output.

Game 6. We now construct a machine Sim that internally simulates the machines Adv, A^C, $\mathcal{F}_{\text{FakeCOM}}$, and Bob. We let Sim run with an (external) corruption party A^C, and when (the simulated) Bob computes s_0, s_1 in Step 11, Sim instructs the (external) corruption party A^C to input s_0, s_1 into $\mathcal{F}_{\text{ROT}}$ (instead of letting Bob output $s = s_c$). Then $\mathcal{F}_{\text{ROT}}$ will, given input c from the dummy-party $\tilde{B}$, output s_c to the dummy-party $\tilde{B}$. The dummy-party $\tilde{B}$ then forwards s_c to the environment $\mathcal{Z}$. See Figure 2(c). The only difference with respect to the previous network (besides a regrouping of machines) is that now s_c is computed by $\mathcal{F}_{\text{ROT}}$ from s_0, s_1. However, $\mathcal{F}_{\text{ROT}}$ computes s_c in the same way as Bob would have done. Thus, the resulting network is perfectly indistinguishable from the previous one.

Since the network from Game 6 (Figure 2(c)) is identical to the ideal model (Figure 2(b)), and since the real model is perfectly indistinguishable from the network from Game 6, we have that the real and the ideal network are perfectly indistinguishable.

Furthermore, Sim is quantum-polynomial-time if Adv is, and the construction of Sim does not depend on the choice of the environment $\mathcal{Z}$. Thus the protocol π_{QROT} statistically quantum-UC-emulates $\mathcal{F}_{\text{ROT}}^{A\to B,\ell}$ in the case of corrupted Alice. □

Theorem 16. *Fix constants $0 < \alpha < 1$ and $0 < \lambda < \frac{1}{4}$. Let $m := \lceil n/(1-\alpha) \rceil$ and $\ell := \lfloor \lambda n \rfloor$ and assume that n grows at least linearly in the security parameter. Then the protocol π_{QROT} statistically quantum-UC-emulates $\mathcal{F}_{\text{ROT}}^{A\to B,\ell}$.*

For the case of corrupted Alice, this is shown in Lemma 15. The cases where both parties are honest or both parties are corrupted are trivial. Thus for Theorem 16 we are left to analyze the case where Bob is corrupted. This case needs a considerably more involved analysis than the case of corrupted Alice because we

have to consider the fact that Bob may succeed in Step 6 of π_{QROT} but still have a certain amount of information about the bits $x^A|_{I_{1-c}}$. A very similar analysis has already been performed by Damgård, Fehr, Lunemann, Salvail, and Schaffner [9] in the so-called stand-alone model. Fortunately, we do not need to redo their analysis; it turns out that – although the stand-alone model is weaker than the quantum-UC-model – the particular simulator constructed by Damgård et al. is already strong enough to be used as a simulator in the quantum-UC-model. Thus we can reuse the result of Damgård et al. in our setting and get Theorem 16 without re-analyzing π_{QROT}.[10]

The full proof of Theorem 16 is given in the full version [21].

Theorem 17. *Let $0 < \alpha < 1$ and $0 < \lambda < \frac{1}{4}$ be constants. Assume $m = \lceil n/(1-\alpha) \rceil$ and $\ell = \lfloor \lambda n \rfloor$ and that n grows at least linearly in the security parameter. Then the protocol π_{QOT} (Def. 14) statistically quantum-UC-emulates $\mathcal{F}_{\mathrm{OT}}^{A\to B,\ell}$.*

Proof. Consider the following protocol π'_{QOT} in the $\mathcal{F}_{\mathrm{ROT}}$-hybrid model. Given inputs $v_0, v_1 \in \{0,1\}^{\ell(k)}$ for Alice and a bit c for Bob, Bob invokes $\mathcal{F}_{\mathrm{ROT}}$ with input c. Then Alice gets random $s_0, s_1 \in \{0,1\}^{\ell(k)}$, and Bob gets $s = s_c$. Then Alice sends t_0, t_1 with $t_i := v_i \oplus s_i$ to Bob. And Bob outputs $s \oplus t_c$. It is easy to see that π'_{QOT} statistically classical-UC-emulates $\mathcal{F}_{\mathrm{OT}}$. Hence, by the quantum lifting theorem (Theorem 13), π'_{QOT} statistically quantum-UC-emulates $\mathcal{F}_{\mathrm{OT}}$. Note that the protocol π_{QOT} is the protocol resulting from replacing, in π'_{QOT}, calls to $\mathcal{F}_{\mathrm{ROT}}$ by calls to the subprotocol π_{QROT}. Furthermore, π_{QROT} statistically quantum-UC-emulates $\mathcal{F}_{\mathrm{ROT}}$ by Theorem 16. Hence, by the composition theorem (Theorem 11), π_{QOT} statistically quantum-UC-emulates $\mathcal{F}_{\mathrm{OT}}$. □

5 Multi-party Computation

Theorem 18. *Let $\mathcal{F}$ be a classical probabilistic-polynomial-time functionality.*[11] *Then there exists a protocol π in the $\mathcal{F}_{\mathrm{COM}}$-hybrid model that statistically quantum-UC-emulates $\mathcal{F}$. (Assuming the number of protocol parties does not depend on the security parameter.)*

Proof. Ishai, Prabhakaran, and Sahai [13] prove the existence of a protocol $\rho^{\mathcal{F}_{\mathrm{OT}}}$ in the $\mathcal{F}_{\mathrm{OT}}$-hybrid model that statistically classical-UC-emulates $\mathcal{F}$ (assuming

[10] One major difference between the UC-model and the stand-alone model is that in the first, the honest parties' inputs may depend on messages the adversary intercepts during the protocol run. A simulator constructed for the stand-alone model usually is not able to cope with such dependencies. Thus, it turns out to be important that we first considered the randomized OT protocol π_{QROT} and not immediately the OT protocol π_{QOT}. In π_{QROT}, Alice gets no input, and in particular her inputs may not depend on messages intercepted by the adversary.

[11] Subject to certain technical restrictions stemming from the proof by Ishai et al. [13]: Whenever the functionality gets an input, the adversary is informed about the length of that input. Whenever the functionality makes an output, the adversary is informed about the length of that output and may decide when this output is to be scheduled.

a constant number of parties). By the quantum lifting theorem (Theorem 13), $\rho^{\mathcal{F}_{\mathrm{OT}}}$ statistically quantum-UC-emulates $\mathcal{F}$. By Theorem 17, π_{QOT} statistically quantum-UC-emulates $\mathcal{F}_{\mathrm{OT}}$. Let $\pi := \rho^{\pi_{\mathrm{QOT}}}$ be the result of replacing invocations to $\mathcal{F}_{\mathrm{OT}}$ in $\rho^{\mathcal{F}_{\mathrm{OT}}}$ by invocations of the subprotocol π_{QOT} (as described before Theorem 11). Then by the universal composition theorem (Theorem 11), π statistically quantum-UC-emulates $\rho^{\mathcal{F}_{\mathrm{OT}}}$. Using the fact that quantum-UC-emulation is transitive (shown in the full version [21]), it follows that π statistically quantum-UC-emulates $\mathcal{F}$. □

We proceed to show that the result from Theorem 18 is possible only in the quantum setting. That is, we show that there is a natural functionality that cannot be statistically classical-UC-emulated in the commitment-hybrid model.

Definition 19 (AND). *The functionality $\mathcal{F}_{\mathrm{AND}}$ expects an input $a \in \{0,1\}$ from Alice and $b \in \{0,1\}$ from Bob. Then it sends $a \cdot b$ to Alice and Bob.*

Theorem 20 (Impossibility of classical multi-party computation). *There is no classical probabilistic-polynomial-time protocol π in the $\mathcal{F}_{\mathrm{COM}}$-hybrid model such that π statistically classical-UC-emulates $\mathcal{F}_{\mathrm{AND}}$.*

The proof is given in the full version [21].

Acknowledgements. I thank Jörn Müller-Quade for the original inspiration for this work and Christian Schaffner for valuable discussions.

References

1. Ben-Or, M., Crépeau, C., Gottesman, D., Hassidim, A., Smith, A.: Secure multiparty quantum computation with (only) a strict honest majority. In: FOCS 2006, pp. 249–260. IEEE Computer Society, Los Alamitos (2006)
2. Ben-Or, M., Horodecki, M., Leung, D.W., Mayers, D., Oppenheim, J.: The universal composable security of quantum key distribution. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 386–406. Springer, Heidelberg (2005); Preprint at arXiv:quant-ph/0409078v1
3. Ben-Or, M., Mayers, D.: General security definition and composability for quantum & classical protocols. arXiv:quant-ph/0409062v2 (September 2004)
4. Bennett, C.H., Brassard, G.: Quantum cryptography: Public-key distribution and coin tossing. In: IEEE International Conference on Computers, Systems and Signal Processing 1984, pp. 175–179. IEEE Computer Society, Los Alamitos (1984)
5. Bennett, C.H., Brassard, G., Crépeau, C., Skubiszewska, M.-H.: Practical quantum oblivious transfer. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 351–366. Springer, Heidelberg (1992)
6. Brassard, G., Crépeau, C., Jozsa, R., Langlois, D.: A quantum bit commitment scheme provably unbreakable by both parties. In: FOCS 1993, Los Alamitos, CA, USA, pp. 362–371. IEEE Computer Society, Los Alamitos (1993)
7. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: FOCS 2001, pp. 136–145. IEEE Computer Society, Los Alamitos (2001); Full and revised version is IACR ePrint 2000/067

8. Canetti, R., Fischlin, M.: Universally composable commitments. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 19–40. Springer, Heidelberg (2001); Full version is IACR ePrint 2001/055
9. Damgård, I., Fehr, S., Lunemann, C., Salvail, L., Schaffner, C.: Improving the security of quantum protocols. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 408–427. Springer, Heidelberg (2009)
10. Damgård, I., Fehr, S., Salvail, L., Schaffner, C.: Cryptography in the bounded quantum-storage model. In: FOCS 2005, pp. 449–458 (2005); Full version is arXiv:quant-ph/0508222v2
11. Fehr, S., Schaffner, C.: Composing quantum protocols in a classical environment. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 350–367. Springer, Heidelberg (2009)
12. Hofheinz, D., Müller-Quade, J.: A paradox of quantum universal composability. In: 4th European QIPC Workshop, poster (2003), `http://www.quiprocone.org/Hot%20Topics%20posters/muellerquade_poster.pdf`
13. Ishai, Y., Prabhakaran, M., Sahai, A.: Founding cryptography on oblivious transfer – efficiently. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 572–591. Springer, Heidelberg (2008)
14. Kent, A.: Unconditionally secure bit commitment. PRL 83(7), 1447–1450 (1999)
15. Kilian, J.: Founding cryptography on oblivious transfer. In: STOC 1988, pp. 20–31. ACM Press, New York (1988)
16. Mayers, D.: Unconditionally Secure Quantum Bit Commitment is Impossible. Physical Review Letters 78(17), 3414–3417 (1997); Preprint at arXiv:quant-ph/9605044v2
17. Müller-Quade, J., Unruh, D.: Long-term security and universal composability. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 41–60. Springer, Heidelberg (2007)
18. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2000)
19. Pfitzmann, B., Waidner, M.: A model for asynchronous reactive systems and its application to secure message transmission. In: 22nd IEEE Symposium on Security & Privacy, pp. 184–200 (2001)
20. Unruh, D.: Simulatable security for quantum protocols (September 2004), arXiv:quant-ph/0409125v2
21. Unruh, D.: Universally composable quantum multi-party computation (October 2009), arXiv:0910.2912 [quant-ph], Full version of this paper
22. van de Graaf, J.: Towards a formal definition of security for quantum protocols. PhD thesis, Département d'informatique et de r.o., Université de Montréal (1998), `http://www.cs.mcgill.ca/~crepeau/PS/these-jeroen.ps`
23. Watrous, J.: Zero-knowledge against quantum attacks. In: STOC 2006, pp. 296–305. ACM, New York (2006)
24. Wehner, S., Wullschleger, J.: Composable security in the bounded-quantum-storage model. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part II. LNCS, vol. 5126, pp. 604–615. Springer, Heidelberg (2008); Full version is arXiv:0709.0492v1 [quant-ph]
25. Wiesner, S.: Conjugate coding. SIGACT News 15(1), 78–88 (1983) (manuscript written ca. 1970)
26. Yao, A.C.-C.: Security of quantum protocols against coherent measurements. In: STOC 1995, pp. 67–75. ACM, New York (1995)