One-Way Functions Imply Secure Computation in a Quantum World

James Bartusek* Andrea Coladangelo† Dakshita Khurana‡ Fermi Ma§

Abstract

We prove that quantum-hard one-way functions imply *simulation-secure* quantum oblivious transfer (QOT), which is known to suffice for secure computation of arbitrary quantum functionalities. Furthermore, our construction only makes *black-box* use of the quantum-hard one-way function.

Our primary technical contribution is a construction of *extractable and equivocal* quantum bit commitments from quantum-hard one-way functions in the standard model. Instantiating the Bennet-Brassard-Crépeau-Skubiszewska (CRYPTO 91) framework with these commitments yields simulation-secure quantum oblivious transfer.

*UC Berkeley. Email: bartusek.james@gmail.com
†UC Berkeley. Email: andrea.coladangelo@gmail.com
‡UIUC. Email: dakshita@illinois.edu
§Princeton University and NTT Research. Email: fermima@alum.mit.edu

Contents

1 Introduction **3**
- 1.1 Our Results . . . 4
- 1.2 Additional Related Work . . . 5

2 Technical Overview **6**
- 2.1 Recap: BBCS Quantum Oblivious Transfer from Commitments . . . 6
- 2.2 Our Construction: A High-Level Overview . . . 7
- 2.3 Making Any Quantum (or Classical) Commitment Equivocal . . . 8
- 2.4 An Extractability Compiler for Equivocal Commitments . . . 10
- 2.5 Putting it Together: From Commitments to Secure Computation. . . . 11

3 Preliminaries **12**
- 3.1 Bit Commitments . . . 12
- 3.2 Oblivious Transfer with Quantum Communication . . . 14
- 3.3 Quantum Rewinding Lemma . . . 14
- 3.4 Quantum Entropy and Leftover Hashing . . . 15

4 A Quantum Equivocality Compiler **15**

5 Quantum Extractable Commitments **19**

6 Quantum Oblivious Transfer from Extractable and Equivocal Commitments **22**

7 Acknowledgement **26**

A A Quantum Sampling Lemma **29**
- A.1 The [BF10] EPR Protocol . . . 29

1 Introduction

The complexity of cryptographic primitives has been central to the study of cryptography. Much of the work in the field focuses on establishing *reductions* between different primitives, typically building more sophisticated primitives from simpler ones. Reductions imply relative measures of complexity among different functionalities, and over the years have resulted in an expansive hierarchy of assumptions and primitives, as well as separations between them.

At the center of cryptographic complexity lie one-way functions (OWFs): their existence is the minimal assumption necessary for nearly all classical cryptography [LR86, IL89, ILL89]. One-way functions are equivalent to so-called "minicrypt" primitives like pseudorandom generators, pseudorandom functions and symmetric encryption; but provably cannot imply key exchange when used in a black-box way [IR90, BM09][1]. Thus, the existence of key exchange is believed to be a stronger assumption than the existence of one-way functions. Oblivious transfer (OT) — equivalently, secure computation — is believed to be *even stronger*: it implies key exchange, but cannot be obtained from black-box use of a key exchange protocol [MMP14].

The importance of OT stems from the fact that it can be used to achieve secure computation, which is a central cryptographic primitive with widespread applications. In a nutshell, secure computation allows mutually distrusting participants to compute any public function over their joint private inputs while revealing no private information beyond the output of the computation.

The Quantum Landscape. The landscape of cryptographic possibilities changes significantly when participants have quantum computation and communication capabilities. For one, *unconditionally* secure key distribution — commonly known as *quantum key distribution* (QKD) — becomes possible [BB84, May96]. Moreover, *quantum* oblivious transfer (QOT) is known to be achievable from special types of commitments, as we discuss next.

Bennett, Brassard, Crepeau and Skubiszewska [BBCS92] first proposed a protocol for QOT by using quantum bit commitments. The central idea in these QKD and QOT protocols is the use of (what are now known as) "BB84 states". These are single qubit states encoding either 0 or 1 in either the computational or Hadamard basis. Crucially, measuring (or essentially attempting to copy the encoded bit) in the wrong basis completely destroys information about the encoded bit. The original [BBCS92] paper heuristically argued security of the proposed OT protocol, and subsequently, Mayers and Salvail [MS94] proved that the protocol from [BBCS92] is secure against a restricted class of attackers that only perform single-qubit measurements. This was later improved by Yao [Yao95], who extended the [MS94] result to handle general quantum adversaries.

By an unfortunate historical accident, the aforementioned security proofs claimed the [BBCS92] QOT could be *information-theoretically secure*, since at the time it was believed that information-theoretic quantum bit commitment was possible [BCJL93]. Several years later, Mayers [May97] and Lo and Chau [LC97] independently proved the impossibility of information-theoretic quantum bit commitment, and as a consequence, the precise security of [BBCS92] QOT was once again unclear. This state of affairs remained largely unchanged until 2009, when Damgard, Fehr, Lunemann, Salvail, and Schaffner [DFL+09] proved that bit commitment schemes satisfying certain additional properties, namely *extraction and equivocation*, suffice to obtain QOT [BBCS92]. [DFL+09] called their commitments *dual-mode* commitments, and provided a construction based on the quantum hardness of the learning with errors (QLWE) assumption. We remark that assumptions about the hardness of specific problems like QLWE are qualitatively even worse than general assumptions like QOWFs and QOT. Thus, the following basic question remains:

Do quantum-hard one-way functions suffice for quantum oblivious transfer?

[1] In particular, [IR90, BM09] showed that there cannot exist a key exchange protocol that only has oracle access to the input/output behavior of a one-way function, and makes no additional assumptions. Then [Dac16] ruled out the possibility of certain types of key exchange protocols that also make use of the *code* of a one-way function. Constructions of key exchange from one-way functions have eluded researchers for decades. This, combined with the aforementioned negative results, is considered to be evidence that key exchange is a qualitatively stronger primitive than one-way functions in the classical regime. In fact, Impagliazzo [Imp95] stipulates that we live in one of five possible worlds, of which *Minicrypt* is one where classical one-way functions exist but classical public-key cryptographic primitives do not.

Quantum OT: The Basis of Secure Quantum Computation. There is a natural extension of secure computation to the quantum world, where Alice and Bob wish to compute a *quantum* circuit on (possibly entangled) *quantum* input states. This setting, usually referred to as secure *quantum* computation, has been previously studied and in fact has a strong tradition in the quantum cryptography literature.

Secure quantum computation was first studied by [CGS02, BCG+06], who obtained unconditional maliciously-secure general *multi-party* quantum computation with honest majority. The setting where half (or more) of the players are malicious requires computational assumptions due to the impossibility of unconditionally secure quantum bit commitment [May97, LC97].

In this computational setting, [DNS10, DNS12] showed the feasibility of two-party quantum computation (2PQC) assuming post-quantum OT. More recently, [DGJ+20] constructed maliciously-secure general multi-party quantum computation (MPQC) secure against a *dishonest* majority from any maliciously-secure post-quantum multi-party computation (MPC) protocol for classical functionalities, which can itself be obtained from post-quantum OT [ABG+20].

Nevertheless, the following natural question has remained unanswered:

Can secure (quantum) computation be obtained from quantum-hard one-way functions?

1.1 Our Results

Our main result is the following:

Quantum oblivious transfer can be based on the assumption that quantum-hard one-way functions exist.

In fact, we prove a stronger result: we show that quantum oblivious transfer can be based on the *black-box use of quantum-hard one-way functions.* This in turn implies secure two-party computation of classical functionalities, in the presence of quantum computation and communication capabilities, from (black-box use of) quantum-hard one-way functions [Kil88]. The latter can then be used to obtain secure two-party *quantum* computation, by relying on the work of [DNS12]. QOT can also be used to obtain *multi-party* secure computation of all classical functionalities [IPS08], in the presence of quantum computation and communication capabilities, and additionally assuming the existence of authenticated channels. By relying on [DGJ+20], this also implies multi-party secure *quantum* computation.

In summary, our main result implies that: (1) 2PQC can be obtained from (black-box use of) quantum-hard OWFs and (2) assuming the existence of authenticated channels, MPQC can be obtained from (black-box use of) quantum-hard OWFs.

This provides a further potential separation between the complexity of cryptographic primitives in the classical and quantum worlds. In the former, (two-party) secure computation provably cannot be based on black-box used of quantum-hard one-way functions. It is only known from special types of enhanced public-key encryption schemes or from the hardness of specific problems, both of which are believed to be much stronger assumptions than one-way functions. But in the quantum world, prior to our work, (two-party) secure computation was only known from the special commitments required in the protocol of [DFL+09], which can be based on QLWE following [DFL+09], or post-quantum OT (implicit in [HSS11, BS20, ABG+20]) — but were not known to be achievable from quantum-hard one-way functions.

Primary Tool: Stand-alone Extractable and Equivocal Commitments. As discussed earlier in the introduction, [DFL+09] show that simulation-secure OT can be obtained from commitments satisfying certain properties, namely *extraction* and *equivocation*.

- Extraction informally requires that there exist an efficient quantum "extractor", that with access to a quantum committer, is able to extract its committed value.
- Equivocality informally requires that there exist an efficient quantum simulator, that with access to a quantum receiver, is able to open a commitment to any value of its choice.

The two properties are crucial for proving simulation security of the [BBCS92] OT protocol: extraction for receiver security, and equivocality for sender security. Our key technical contribution, which may be of independent interest, is the following:

Extractable and equivocal commitments can be based on the black-box use of quantum-hard one-way functions.

We obtain this result via the following technical steps.

- *Quantum Equivocal Commitments from Quantum-Hard One-Way Functions.* We describe a generic unconditional compiler from any commitment to an equivocal commitment, in the plain model. This compiler makes black-box use of the underlying commitment. By applying our compiler to Naor's statistically binding commitment [Nao91], which can be based on quantum-hard one-way functions, we obtain a statistically binding, equivocal commitment.

- *Quantum Extractable Commitments from Quantum Equivocal Commitments.* We show that the [BBCS92, DFL+09, BF10] framework can be used to obtain an extractable commitment that leverages quantum communication, and can be based on the existence of any quantum equivocal commitment. This combined with the previous step implies the existence of quantum extractable commitments based on the existence of quantum-hard one-way functions.

  This is in contrast to existing approaches (eg., [HSS11]) that require classical communication but rely on qualitatively stronger assumptions like classical OT with post-quantum security.

- *From Extractable Commitments to Extractable and Equivocal Commitments.* We apply the black-box equivocality compiler from the first step to the quantum extractable commitment obtained above, to produce an extractable and equivocal commitment.

  We note that the need to apply our compiler to a protocol with quantum communication meant that we could not directly rely on simple equivocality compilers in the classical cryptography literature. Indeed, generic compilers in the classical setting, in addition to not necessarily being quantum secure, break down in the setting of quantum communication due to the need to make non-black-box use of the underlying commitment.

Plugging our quantum extractable and equivocal commitments into the [BBCS92] framework yields a final QOT protocol with an interaction pattern that readers familiar with [BB84, BBCS92] may find interesting: the sender sends the receiver several BB84 states, after which the receiver proves to the sender that it has honestly measured the sender's BB84 states by *generating more BB84 states of their own and asking the sender to prove that they have measured the receiver's BB84 states.* An intriguing open question is whether obtaining QOT from one-way functions *requires* this type of two-way quantum communication or, alternatively, quantum memory.[2]

1.2 Additional Related Work

The protocol in [DFL+09] can be instantiated [BF10, FUYZ20] with weaker types of commitments (in particular, statistically binding, computationally hiding commitments) to obtain a weaker version of OT, which only satisfies *indistinguishability-based* security, and not the standard notion of *simulation-based* security. Such commitments can be obtained from quantum-hard one-way functions. But this weaker notion is not typical in the classical OT literature and falls short of guaranteeing that the view of a quantum polynomial-time adversary can be efficiently simulated given the input and/or output of the protocol. More crucially, this notion is not known to imply standard (simulation-based) notions of secure multi-party computation. Our focus in this work is on achieving the (standard) simulation-based notion of security for OT – this suffices to instantiate the aforementioned compilers and achieve standard simulation-based secure quantum

[2] Naive approaches to removing one direction of quantum communication appear to require the honest parties to be entangled and subsequently perform quantum teleportation.

computation. Achieving simulation-based OT, specifically one that admits an *efficient* (quantum) simulator involves overcoming multiple technical hurdles, that we discuss in Section 2.

In the setting of Universal Composability (UC), Unruh [Unr07] obtained quantum oblivious transfer from quantum UC-secure commitments in the common reference string (CRS) model. Unfortunately, quantum (and classical) UC-secure commitments also assume the existence of public-key encryption, and are not known to be realizable from quantum-hard OWFs. By contrast, in this work, we focus on the standalone setting in the plain/standard model, and we do not assume the existence of trusted setup or a common reference string. In this setting, we obtain QOT from quantum-hard one-way functions.

2 Technical Overview

This work establishes that (1) black-box use of post-quantum one-way functions suffices for post-quantum *extractable and equivocal* commitment schemes and moreover, that (2) [BBCS92] quantum oblivious transfer instantiated with such commitments is a standard *simulation-secure* oblivious transfer. Crucially, the standard notion of simulation-secure (quantum) oblivious transfer that we achieve is sequentially composable and suffices to achieve general-purpose secure quantum computation.

Before we explain our technical approach, we provide a complete review of the original [BBCS92] protocol.

2.1 Recap: BBCS Quantum Oblivious Transfer from Commitments

In quantum oblivious transfer (QOT), a quantum sender holding two classical messages m_0, m_1 engages in an interactive protocol over a quantum channel with a quantum receiver holding a classical choice bit b. Correctness requires the receiver to learn m_b by the end of the protocol. Informally, security demands that a malicious receiver only learn information about one of m_0, m_1, and that a malicious sender learn nothing about b. Somewhat more formally, as discussed earlier, our focus is on the standard simulation-based notion of security. This stipulates the existence of an efficient quantum simulator that generates the view of an adversary (sender/receiver) when given access to an ideal OT functionality. In particular, when simulating the view of a malicious sender, this simulator must extract the sender's inputs (m_0, m_1) without knowledge of the receiver's input b. And when simulating the view of a malicious receiver, the simulator must extract the receiver's input b, and then simulate the receiver's view given just m_b.

We recall the construction of quantum oblivious transfer due to [BBCS92] (henceforth BBCS), which combines the information theoretic quantum key distribution protocol of [BB84] (henceforth BB84) with cryptographic bit commitments.

BBCS First Message. The first message of the BBCS protocol exactly follows the beginning of the BB84 protocol. For classical bits y, z, let $|y\rangle_z$ denote $|y\rangle$ if $z = 0$, and $(|0\rangle + (-1)^y |1\rangle)/\sqrt{2}$ if $z = 1$, i.e. the choice of z specifies whether to interpret y as a computational or Hadamard basis vector. Let λ denote the security parameter. The sender samples two random 2λ-bit strings x and θ, and constructs "BB84 states" $|x_i\rangle_{\theta_i}$ for $i \in [2\lambda]$. The sender forwards these 2λ BB84 states $(|x_i\rangle_{\theta_i})_{i\in[2\lambda]}$ to the receiver. Next, the receiver samples a 2λ-bit string $\hat{\theta}$, measures each $|x_i\rangle_{\theta_i}$ in the basis specified by $\hat{\theta}_i$, and obtains a 2λ-bit measurement result string $\hat{x}$.

BBCS Measurement-Check Subprotocol. At this point, the BBCS and BB84 protocols diverge. Since the BB84 protocol is an interaction between two *honest* parties, it assumes the parties comply with the protocol instructions. However, in the BBCS protocol, a malicious receiver who does not measure these BB84 states will be able to compromise sender privacy later in the protocol. Therefore, the next phase of BBCS is a measurement-check subprotocol designed to catch a malicious receiver who skips the specified measurements. This subprotocol requires the use of a quantum-secure classical commitment scheme; for the purposes of this recap, one should imagine a commitment with idealized hiding and binding properties. The subprotocol proceeds as follows:

- For each $i \in [2\lambda]$, the receiver commits to $(\hat{\theta}_i, \hat{x}_i)$.
- Next, the sender picks a random set T of λ indices from $[2\lambda]$, and challenges the receiver to open the corresponding commitments.
- The receiver sends $(\hat{\theta}_i, \hat{x}_i)$ along with the corresponding opening for each $i \in T$.
- The sender verifies each commitment opening, and furthermore checks that $\hat{x}_i = x_i$ for each $i \in T$ where $\hat{\theta}_i = \theta_i$. If any of these checks fail, the sender aborts.

The rough intuition for the subprotocol is simple: from the receiver's point of view, the BB84 states are maximally mixed and therefore completely hide x_i and θ_i. For any index i that the receiver does not measure, it must guess $\hat{x}_i$. From the receiver's perspective, the sender checks $\hat{x}_i$ against x_i if two 1/2-probability events occur: (1) i is included in T, and (2) $\hat{\theta}_i = \theta_i$. This means a malicious receiver who skips a significant number of measurements will be caught with overwhelming probability.

BBCS Privacy Amplification. If all the subprotocol checks pass, the sender continues to the final stage of the BBCS protocol. For convenience, relabel the λ indices in $[2\lambda] \setminus T$ from 1 to λ; all indices corresponding to opened commitments are discarded for the remainder of the protocol.

For each $i \in [\lambda]$, the sender reveals the correct measurement basis θ_i. The receiver then constructs the index set I_b — where b is its choice bit for the oblivious transfer — as the set of all $i \in [\lambda]$ where $\theta_i = \hat{\theta}_i$. It sets I_{1-b} to be the remaining indices, and sends (I_0, I_1) to the sender. Note that by the hiding property of the commitments, the sender should not be able to deduce b from (I_0, I_1); furthermore, I_0 and I_1 will both be close to size $\lambda/2$, since for each $i \in [\lambda]$, the receiver committed to $\hat{\theta}_i$ before obtaining θ_i.

On receiving I_0, I_1, the sender sets $x_0 \coloneqq (x_i)_{i \in I_0}$ and $x_1 \coloneqq (x_i)_{i \in I_1}$. The intuition is that if a receiver honestly constructs (I_0, I_1), it will only have information about x_b corresponding to its choice bit b. However, it turns out that even if the receiver maliciously constructs (I_0, I_1), at least one of x_0 and x_1 will have high min-entropy from its point of view. Thus, by standard privacy amplification techniques, the sender can complete the oblivious transfer as follows. It samples two universal hash functions h_0 and h_1, both with ℓ-bit outputs, and uses $h_0(x_0)$ to mask the ℓ-bit message m_0, and uses $h_1(x_1)$ to mask m_1. That is, the sender sends $(h_0, h_1, h_0(x_0) \oplus m_0, h_1(x_1) \oplus m_1)$ to the receiver, who can then use x_b to recover m_b. Since x_{1-b} will have high entropy, the leftover hash lemma implies that $h_{1-b}(x_{1-b})$ is statistically close to uniform, which hides m_{1-b} from the receiver.

The BBCS QOT protocol can be shown to satisfy simulation-based security when instantiated with an *extractable and equivocal* commitment [DFL+09]. With this in mind, we now turn to constructing such commitments from quantum-hard one-way functions.

2.2 Our Construction: A High-Level Overview

The rest of this technical overview describes our *black-box* construction of simultaneously *extractable and equivocal* quantum bit commitments from any quantum-hard one-way function.

The ingredients for our construction are the following:

- A general-purpose "equivocality compiler" that turns any bit commitment scheme — classical or quantum — into an *equivocal* quantum commitment scheme. Moreover, if the original commitment scheme is *extractable*, this compiler outputs an *extractable and equivocal* commitment scheme.
- A general-purpose "extractability compiler" that turns any *equivocal* bit commitment scheme — classical or quantum — into an *extractable but not equivocal* commitment scheme.

Both of these compilers require no additional computational assumptions beyond those of the original commitment schemes. Given these compilers, we build extractable and equivocal commitments via the following steps:

1. Begin with Naor's statistically-binding and computationally hiding commitment scheme [Nao91]. Naor's construction makes black-box use of one-way functions and achieves post-quantum computational hiding assuming post-quantum security of the one-way function.[3]

2. Plug Naor's commitments into our equivocality compiler to obtain an *equivocal* quantum bit commitment scheme.

3. Feed the resulting equivocal quantum bit commitments into our extractability compiler to obtain an *extractable but not equivocal* quantum bit commitment.

4. Run the equivocality compiler *a second time*, but now starting with the extractable commitments produced by the previous step. This gives the desired *extractable and equivocal* quantum bit commitments.

We briefly remark that if one did not care about obtaining a construction that only makes *black-box* use of one-way functions, it would only be necessary to invoke the equivocality compiler once. In slightly more detail, we could design an alternative extractability compiler that works with any (not necessarily equivocal) bit commitment scheme by additionally relying on post-quantum zero-knowledge arguments [Wat06]; this approach would still give a construction of quantum oblivious transfer from post-quantum one-way functions, but the zero-knowledge arguments would be used to prove statements involving the *description* of some one-way function, which is a non-black-box use of the primitive. We stress that constructing QOT from one-way functions is interesting regardless of whether or not the construction is black-box. However, since black-box constructions are generally preferred over non-black-box ones (e.g. for conceptual simplicity, efficiency, etc.) our focus will be on presenting a black-box construction.

Organization. We describe our equivocality compiler in Section 2.3 and our extractability compiler in Section 2.4. In Section 2.5, we briefly discuss implications for secure computation in a quantum world.

2.3 Making Any Quantum (or Classical) Commitment Equivocal

Roughly speaking, a quantum commitment protocol is *equivocal* if an efficient quantum algorithm called the *equivocator*, with access to the receiver, can generate commitments that can be opened to any value. More precisely, for any receiver (modeled as an efficient malicious quantum algorithm), there must exist an equivocator who can generate a computationally indistinguishable commitment that the equivocator can later open arbitrarily.

In this subsection, we describe a *black-box* compiler for a fairly general task (which may be of independent interest): making any *classical or quantum* commitment equivocal. Recall from Section 2.2 that we will need to invoke our equivocality compiler *twice*, once on a classical bit commitment scheme, and once on an extractable quantum bit commitment scheme; in the latter case, our compiler will need to preserve the extractability of the original commitment. Since classical commitments are a subclass of quantum commitments, our exposition will focus on challenges unique to the quantum setting.

Why Existing Classical Solutions Are Insufficient. Let us briefly relax our goal of using one-way functions in an exclusively black-box way. Then there is a simple equivocality compiler that applies to any (statistically-binding and computationally-hiding) classical commitment. Recall that quantum-hard one-way functions imply post-quantum zero-knowledge arguments for NP [Wat06]. Now, in the opening phase of the commitment, to open to a bit b, the committer will not send the randomness used to commit to b in the clear. Instead, the committer sends a zero-knowledge proof for the NP statement that there exist randomness consistent with the commitment, and opening to b. Equivocation is achieved by simulating the

[3] In slightly more detail, Naor's commitment scheme makes black-box use of any pseudo-random generator. It is straightforward to verify that if the pseudorandom generator is post-quantum secure, the commitment satisfies computational hiding against quantum attackers. A black-box construction of pseudo-random generators from one-way functions is due to [HILL99]; Aaronson [Aar09] and Zhandry [Zha12] observed that [HILL99] makes no assumptions about the computational model of the attacker and therefore applies to quantum attackers.

ZK proof. Unfortunately, this technique fails when the commitment involves *quantum* communication, since the statement to be proven is no longer an NP statement (nor a QMA statement). Therefore, we construct a compiler that only makes black-box use of the underlying commitment, and leverages Watrous's rewinding lemma in the proof of equivocality [Wat06].

Our Equivocality Compiler. In our construction, to commit to a bit b, the committer and receiver will perform λ sequential repetitions of the following subprotocol:

- The (honest) committer samples 2 uniformly random bits u_0, u_1, and commits to each one *twice* using the base commitment scheme. Let the resulting commitments be $\mathbf{c}_0^{(0)}, \mathbf{c}_0^{(1)}, \mathbf{c}_1^{(0)}, \mathbf{c}_1^{(1)}$, where the first two are to u_0 and the second two are to u_1. Note that since the base commitment scheme can be an arbitrary quantum interactive commitment, each commitment $\mathbf{c}_{b_1}^{(b_2)}$ corresponds to the receiver's quantum state after the commitment phase of the base commitment.
- The receiver sends the committer a random challenge bit β.
- The committer opens the two base commitments $\mathbf{c}_\beta^{(0)}, \mathbf{c}_\beta^{(1)}$. If the openings are invalid or the revealed messages are different, the receiver aborts the entire protocol.

If these λ executions pass, the receiver is convinced that a majority of the committer's remaining 2λ unopened commitments are honestly generated, i.e. most pairs of commitments are to the same bit.

Rewriting the (honest) committer's unopened bits as $u_1, \ldots, u_\lambda$, the final step of the commitment phase is for the committer to send $h_i \coloneqq u_i \oplus b$ for each $i \in [\lambda]$ (recall that b is the committed bit).

To decommit, the committer reveals each u_i by picking one of the two corresponding base commitments at random, and opening it. The receiver accepts if each one of the base commitment openings is valid, and the opened u_i satisfies $h_i \oplus u_i = b$ for every i.

The (statistical) binding property of the resulting commitment can be seen to follow from the (statistical) binding of the underlying commitment. For any commitment, define the unique committed value as the majority of $(h_i \oplus u_i)$ values in the unopened commitments, setting u_i to $\perp$ if both committed bits in the i^{th} session differ. Due to the randomized checks by the receiver, any committer that tries to open to a value that differs from the unique committed value will already have been caught in the commit phase, and the commitment will have been rejected with overwhelming probability. A similar argument also allows us to establish that this transformation preserves extractability of the underlying commitment. We now discuss why the resulting commitment is *equivocal*.

Quantum Equivocation. The natural equivocation strategy should have the equivocator (somehow) end up with λ pairs of base commitments where for each $i \in [\lambda]$, the pair of commitments is to u_i and $1 - u_i$ for some random bit u_i. This way, it can send an appropriately distributed string $h_1, \cdots, h_\lambda$, and later open to any b by opening the commitment to $b \oplus h_i$ for each i.

We construct our equivocator using Watrous's quantum rewinding lemma [Wat06] (readers familiar with Watrous's technique may have already noticed our construction is tailored to enable its use).

We give a brief, intuition-level recap of the rewinding technique as it pertains to our equivocator. Without loss of generality, the malicious quantum receiver derives its challenge bit β by performing some binary outcome measurement on the four quantum commitments it has just received (and on any auxiliary states). Our equivocator succeeds (in one iteration) if it can prepare four quantum commitments $\mathbf{c}_0^{(0)}, \mathbf{c}_0^{(1)}, \mathbf{c}_1^{(0)}, \mathbf{c}_1^{(1)}$ where:

1. $\mathbf{c}_\alpha^{(0)}, \mathbf{c}_\alpha^{(0)}$ are commitments to the same random bit,
2. $\mathbf{c}_{1-\alpha}^{(0)}, \mathbf{c}_{1-\alpha}^{(0)}$ are commitments to a random bit and its complement,
3. on input $\mathbf{c}_0^{(0)}, \mathbf{c}_0^{(1)}, \mathbf{c}_1^{(0)}, \mathbf{c}_1^{(1)}$, the receiver produces challenge bit $\beta = \alpha$.

That is, the equivocator is successful if the receiver's challenge bit β corresponds to the bit α that it can open honestly. Watrous's [Wat06] rewinding lemma applies if the distribution of β is *independent* of the receiver's choice of α, which is guaranteed here by the hiding of the base commitments. Thus, the rewinding lemma yields a procedure for obtaining an honest-looking interaction where all three properties above are met. Given the output of the rewinding process, the equivocator has successfully "fooled" the committer on this interaction and proceeds to perform this for all λ iterations. As described above, fooling the committer on all λ iterations enables the equivocator to later open the commitment arbitrarily.

2.4 An Extractability Compiler for Equivocal Commitments

In this subsection, we compile any classical or quantum *equivocal* bit commitment into a quantum *extractable* bit commitment. We stress that even though this compiler is applied to equivocal bit commitments, the resulting commitment is *not* guaranteed to be simultaneously *extractable and equivocal*; we refer the reader to Section 2.2 for details on how this compiler fits into our final construction.

Recall that a commitment scheme is said to be *extractable* if for any adversarial quantum committer that successfully completes the commitment phase, there exists an efficient quantum algorithm (called the extractor) which outputs the committed bit.

Construction. The committer, who intends to commit to a classical bit b, begins by sampling 2λ-bit strings x and θ. It generates the corresponding 2λ BB84 states $|x_i\rangle_{\theta_i}$ and sends this to the receiver. The receiver picks 2λ random measurement bases $\hat{\theta}_i$, and measures each $|x_i\rangle_{\theta_i}$ in the corresponding basis, obtaining outcomes $\hat{x}_i$.

Next, the receiver and committer engage in a BBCS-style measurement-check subprotocol. That is, they temporarily switch roles (for the duration of the subprotocol), and perform the following steps:

1. The receiver (acting as a committer in the subprotocol), commits to each $\hat{\theta}_i$ and $\hat{x}_i$ (for each $i \in [2\lambda]$) with an *equivocal* commitment.

2. The committer (acting as a receiver in the subprotocol), asks the receiver to open the equivocal commitments for all $i \in T$, where $T \subset [2\lambda]$ is a random set of size λ.

3. The receiver (acting as a committer in the subprotocol) opens the λ commitments specified by T.

Provided the receiver passes the measurement-check subprotocol, the committer generates the final message of the commitment phase as follows:

- Discard the indices in T and relabel the remaining λ indices from 1 to λ.
- Partition $\{x_1, \ldots, x_\lambda\}$ into $\sqrt{\lambda}$ strings $\vec{x}_1, \ldots, \vec{x}_{\sqrt{\lambda}}$ each of length $\sqrt{\lambda}$.
- Sample $\sqrt{\lambda}$ universal hash functions $h_1, \ldots, h_{\sqrt{\lambda}}$ each with 1-bit output.
- Finally, send
$$(\theta_i)_{i\in[\lambda]}, (h_j, h_j(\vec{x}_j) \oplus b)_{j\in[\sqrt{\lambda}]}.$$

This concludes the commitment phase.

To decommit, the committer reveals b and $(\vec{x}_1, \ldots, \vec{x}_{\sqrt{\lambda}})$. The receiver accepts if (1) for each j, the bit b and the value $\vec{x}_j$ are consistent with the claimed value of $h_j(\vec{x}_j) \oplus b$ from the commit phase, and (2) for each index $i \in [\lambda]$ where $\theta_i = \hat{\theta}_i$, the x_i from the opening is consistent with $\hat{x}_i$.

Extraction. The use of equivocal commitments in the measurement-check subprotocol makes extraction simple. Given any malicious committer, we construct an extractor as follows.

The extractor plays the role of the receiver and begins an interaction with the malicious committer. But once the committer sends its 2λ BB84 states, the extractor skips the specified measurements, instead leaving these states unmeasured. Next, instead of performing honest commitments to each $\hat{\theta}_i, \hat{x}_i$, the extractor invokes (for each commitment) the equivocator algorithm of the underlying equivocal commitment scheme. Since the equivocator is guaranteed to produce an indistinguishable commitment from the point of view of any malicious receiver for the equivocal commitment, this dishonest behavior by the extractor will go undetected.

When the malicious committer responds with a challenge set $T \subset [2\lambda]$, the extractor samples uniformly random bases $\hat{\theta}_i$ for each $i \in T$, measures the corresponding BB84 states to obtain $\hat{x}_i$ values, and sends $(\hat{\theta}_i, \hat{x}_i)_{i \in T}$. Moreover, the equivocator (for each commitment) will enable the extractor to generate valid-looking openings for all of these claimed values.

Thus, the malicious committer proceeds with the commitment protocol, and sends

$$(\theta_i)_{i \in [\lambda]}, (h_j, h_j(\vec{x}_j) \oplus b)_{j \in [\sqrt{\lambda}]}$$

to the extractor. These correspond to the λ BB84 states that the extractor has not yet measured, so it can simply read off the bases θ_i, perform the specified measurements, and extract the committer's choice of b.

Statistical Hiding. Intuitively, statistical hiding of the above commitment protocol follows because the measurement-check subprotocol forces the receiver to measure states in arbitrary bases, which destroys information about the corresponding x_i values whenever $\widehat{\theta}_i \neq \theta_i$. The formal argument is a straightforward application of a quantum sampling lemma of [BF10], and we defer further details to the body of the paper.

2.5 Putting it Together: From Commitments to Secure Computation.

Plugging the compilers of Sections 2.3 and 2.4 into the steps described in Section 2.2 yields a black-box construction of simultaneously extractable and equivocal quantum bit commitments from quantum-hard one-way functions. Following [DFL+09], these commitments can be plugged into BBCS to obtain maliciously simulation-secure QOT (see Section 6 for further details). Finally, going from QOT to arbitrary secure computation (in a black-box way) follows from a number of prior works, which we briefly survey here for the sake of completeness (see also the discussion in Section 1.1).

- **Secure Computation of Classical Functionalities.** It is well known that maliciously simulation-secure (classical) oblivious transfer can be used in a black-box way to build two-party (classical) computation [Kil88]. This also extends to the multi-party setting tolerating upto all-but-one corruptions [IPS08]. Since the constructions of [Kil88] and [IPS08] only make black-box use of the underlying oblivious transfer and the simulators are straight-line, their correctness/security guarantees continue to hold if parties are quantum and the oblivious transfer uses quantum communication. Therefore, if all parties are quantum and all pairs of parties are connected via authenticated quantum channels, they can securely compute any classical functionality assuming quantum-hard one way functions (and authenticated channels in the multi-party setting).

- **Secure Computation of Quantum Functionalities.** [DNS12] constructed a secure two-party quantum computation protocol assuming black-box access to any secure two-party classical computation protocol. [DGJ+20] proved the analogous statement in the multi-party setting: secure multi-party classical computation implies (in a black-box way) secure multi-party quantum computation. Therefore, by invoking the results in the previous bullet, quantum-hard one-way functions also suffice for arbitrary secure quantum computation (additionally assuming authenticated channels in the multi-party setting).

While the results in the second bullet technically subsume the first, stating them separately enables direct comparison with the classical setting. In particular, one takeaway is that while oblivious transfer is necessary and sufficient for secure computation in the classical world, quantum-hard one-way functions suffice for *exactly the same task* when parties (and their communication channels) are quantum.

3 Preliminaries

Notation. We will write density matrices/quantum random variables (henceforth, QRVs) in lowercase bold font, e.g. $\mathbf{x}$. A quantum register X will be written in uppercase (grey) serif font. A collection of (possibly entangled) QRVs will be written as $(\mathbf{x}, \mathbf{y}, \mathbf{z})$.

Throughout this paper, λ will denote a cryptographic security parameter. We say that a function $\mu(\lambda)$ is *negligible* if $\mu(\lambda) = 1/\lambda^{\omega(1)}$.

The trace distance between two QRVs $\mathbf{x}$ and $\mathbf{y}$ will be written as $\|\mathbf{x}-\mathbf{y}\|_1$. Recall that the trace distance captures the maximum probability that two QRVs can be distinguished by any (potentially inefficient) procedure. We therefore say that two infinite collections of QRVs $\{\mathbf{x}_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\mathbf{y}_\lambda\}_{\lambda\in\mathbb{N}}$ are *statistically indistinguishable* if there exists a negligible function $\mu(\lambda)$ such that $||\mathbf{x}_\lambda-\mathbf{y}_\lambda||_1 \leq \mu(\lambda)$, and we will frequently denote this with the shorthand $\{\mathbf{x}_\lambda\}_{\lambda\in\mathbb{N}} \approx_s \{\mathbf{y}_\lambda\}_{\lambda\in\mathbb{N}}$.

Non-Uniform Quantum Advice. We will consider non-uniform quantum polynomial-time (QPT) algorithms *with quantum advice*, denoted by $\mathcal{A} = \{\mathcal{A}_\lambda, \boldsymbol{\rho}_\lambda\}_{\lambda\in\mathbb{N}}$, where each $\mathcal{A}_\lambda$ is the classical description of a poly(λ)-size quantum circuit, and each $\boldsymbol{\rho}_\lambda$ is some (not necessarily efficiently computable) non-uniform poly(λ)-qubit quantum advice. We remark that "non-uniform quantum polynomial-time algorithms" often means non-uniform *classical advice*, but the cryptographic applications in this work will require us to explicitly consider quantum advice.

Therefore, *computational indistinguishability* will be defined with respect to non-uniform QPT distinguishers with quantum advice. That is, two infinite collections of QRVs $\{\mathbf{x}_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\mathbf{y}_\lambda\}_{\lambda\in\mathbb{N}}$ are computationally indistinguishable if there exists a negligible function $\mu(\cdot)$ such that for all QPT distinguishers $\mathcal{D} = \{\mathcal{D}_\lambda, \boldsymbol{\sigma}_\lambda\}_{\lambda\in\mathbb{N}}$,

$$|\Pr[\mathcal{D}_\lambda(\boldsymbol{\sigma}_\lambda, \mathbf{x}_\lambda) = 1] - \Pr[\mathcal{D}_\lambda(\boldsymbol{\sigma}_\lambda, \mathbf{y}_\lambda) = 1]| \leq \mu(\lambda).$$

We will frequently denote this with the shorthand $\{\mathbf{x}_\lambda\}_{\lambda\in\mathbb{N}} \approx_c \{\mathbf{y}_\lambda\}_{\lambda\in\mathbb{N}}$.

3.1 Bit Commitments

We define bit commitments with quantum players and quantum communication. First, we fix some notation.

A bit commitment scheme is a two-phase interactive protocol between a quantum interactive committer $\mathcal{C} = (\mathcal{C}_{\mathsf{com}}, \mathcal{C}_{\mathsf{open}})$ and a quantum interactive receiver $\mathcal{R} = (\mathcal{R}_{\mathsf{com}}, \mathcal{R}_{\mathsf{open}})$. In the commit phase, $\mathcal{C}_{\mathsf{com}}(1^\lambda, b)$ for bit $b \in \{0,1\}$ interacts with $\mathcal{R}_{\mathsf{com}}(1^\lambda)$, after which $\mathcal{C}_{\mathsf{com}}$ outputs a state $\mathbf{x}_{\mathsf{com}}$ and $\mathcal{R}_{\mathsf{com}}$ outputs a state $\mathbf{y}_{\mathsf{com}}$. We denote this interaction by $(\mathbf{x}_{\mathsf{com}}, \mathbf{y}_{\mathsf{com}}) \leftarrow \langle \mathcal{C}(1^\lambda, b), \mathcal{R}(1^\lambda)\rangle$.

In the open phase, $\mathcal{C}_{\mathsf{open}}(\mathbf{x}_{\mathsf{com}})$ interacts with $\mathcal{R}_{\mathsf{open}}(\mathbf{y}_{\mathsf{com}})$, after which $\mathcal{R}_{\mathsf{open}}$ either outputs a bit b' or $\perp$. We will denote this receiver's output by $\mathsf{OUT}_{\mathcal{R}}\langle \mathcal{C}_{\mathsf{open}}(\mathbf{x}_{\mathsf{com}}), \mathcal{R}_{\mathsf{open}}(\mathbf{y}_{\mathsf{com}})\rangle$.

First, we give the standard notions of hiding and binding.

Definition 3.1 (Hiding Commitment)**.** A bit commitment scheme is computationally *hiding* if the following holds. For any polynomial-size receiver $\mathcal{R}^*_{\mathsf{com}} = \{\mathcal{R}^*_{\mathsf{com},\lambda}, \boldsymbol{\rho}_\lambda\}$ interacting in the commit phase of the protocol, let $\mathsf{OUT}_{\mathcal{R}}\langle \mathcal{C}_{\mathsf{com}}(1^\lambda, b), \mathcal{R}^*_{\mathsf{com},\lambda}(\boldsymbol{\rho}_\lambda)\rangle$ denote a bit output by $\mathcal{R}^*_{\mathsf{com},\lambda}$ after interaction with an honest $\mathcal{C}_{\mathsf{com}}$ committing to message b. Then for every polynomial-size receiver $\mathcal{R}^*_{\mathsf{com}} = \{\mathcal{R}^*_{\mathsf{com},\lambda}, \boldsymbol{\rho}_\lambda\}$, there exists a negligible function $\nu(\cdot)$ such that

$$\left|\Pr[\mathsf{OUT}_{\mathcal{R}}\langle \mathcal{C}_{\mathsf{com}}(1^\lambda, 0), \mathcal{R}^*_{\mathsf{com},\lambda}(\boldsymbol{\rho}_\lambda)\rangle = 1] - \Pr[\mathsf{OUT}_{\mathcal{R}}\langle \mathcal{C}_{\mathsf{com}}(1^\lambda, 1), \mathcal{R}^*_{\mathsf{com},\lambda}(\boldsymbol{\rho}_\lambda)\rangle = 1]\right| = \nu(\lambda).$$

Definition 3.2 (Binding Commitment)**.** A bit commitment scheme is computationally (resp. statistically) *binding* if for every polynomial-size (resp. unbounded-size) committer $\mathcal{C}^* = \{\mathcal{C}^*_{\mathsf{com},\lambda}, \mathcal{C}^*_{\mathsf{open},\lambda}, \boldsymbol{\rho}_\lambda\}$, there exists a negligible function $\nu(\cdot)$ such that with probability $\geq 1 - \nu(\lambda)$ over $(\mathbf{x}_{\mathsf{com}}, \mathbf{y}_{\mathsf{com}}) \leftarrow \langle \mathcal{C}^*_{\mathsf{com}}(\boldsymbol{\rho}), \mathcal{R}(1^\lambda)\rangle$ (where the indexing by λ is omitted for convenience), there exists a bit $b \in \{0,1\}$ such that

$$\Pr[\mathsf{OUT}_{\mathcal{R}}\langle \mathcal{C}^*_{\mathsf{open}}(\mathbf{x}_{\mathsf{com}}), \mathcal{R}_{\mathsf{open}}(\mathbf{y}_{\mathsf{com}})\rangle = b] \leq \nu(\lambda).$$

Post-quantum bit commitments satisfying computational hiding and statistical binding can be constructed from any post-quantum pseudorandom generator (PRG) [Nao91].[4] Watrous's paper [Wat06] considers PRGs built from (post-quantum) one-way *permutations* via a "quantum Goldreich-Levin Theorem" of [AC02]. However, Aaronson [Aar09] and Zhandry [Zha12] later pointed out that the original [HILL99] construction of PRGs from one-way *functions* makes no assumptions on the computational model of the attacker, and therefore immediately extends to non-uniform quantum adversaries. Thus, post-quantum computationally hiding and statistically binding commitments are known from post-quantum one-way functions.

In addition, we will now define notions of extractability and equivocation for commitments. Extractability is a strengthening of (computational) binding, and equivocation is a strengthening of (computational) hiding.

Definition 3.3 (Extractable Commitment)**.** A bit commitment scheme is *extractable* if for every polynomial-size quantum adversarial committer $\mathcal{C}^* = \{\mathcal{C}^*_{\mathsf{com},\lambda}, \mathcal{C}^*_{\mathsf{open},\lambda}, \boldsymbol{\rho}_\lambda\}$, there exists a QPT extractor $\mathcal{E}_{\mathcal{C}^*}$ such that for every polynomial-size quantum distinguisher $\mathcal{D}^* = \{\mathcal{D}^*_\lambda, \boldsymbol{\sigma}_\lambda\}$, there exists a negligible function $\nu(\cdot)$ such that:

$$\left|\Pr[\mathcal{D}^*_\lambda(\boldsymbol{\sigma}_\lambda, \mathsf{Real}) = 1] - \Pr[\mathcal{D}^*_\lambda(\boldsymbol{\sigma}_\lambda, \mathsf{Ideal}) = 1]\right| = \nu(\lambda)$$

for Real and Ideal distributions defined below (where we omit indexing by λ for the sake of presentation).

- Denote by Real the distribution consisting of $(\mathbf{x}_{\mathsf{final}}, b)$ where $\mathbf{x}_{\mathsf{final}}$ denotes the final state of $\mathcal{C}^*_{\mathsf{open}}$ and $b \in \{0,1,\perp\}$ denotes the output by $\mathcal{R}_{\mathsf{open}}$ after the open phase.
- To define the distribution Ideal, first run the extractor $\mathcal{E}_{\mathcal{C}^*}$ on input ρ to obtain outputs $(\mathbf{x}_{\mathsf{com}}, \mathbf{y}_{\mathsf{com}}, b^*)$, where $\mathbf{x}_{\mathsf{com}}$ and $\mathbf{y}_{\mathsf{com}}$ are the final states of committer and receiver respectively after the commit phase. Then run $\mathcal{C}^*(\mathbf{x}_{\mathsf{com}})$ and $\mathcal{R}(\mathbf{y}_{\mathsf{com}})$ to produce $(\mathbf{x}_{\mathsf{final}}, b)$, where $\mathbf{x}_{\mathsf{final}}$ is the final state of the committer. If $b \notin \{\perp, b^*\}$, output FAIL and otherwise output $(\mathbf{x}_{\mathsf{final}}, b)$.

Definition 3.4 (Equivocal Commitments)**.** A bit commitment scheme is *equivocal* if for every polynomial-size quantum adversarial receiver $\mathcal{R}^* = \{\mathcal{R}^*_{\mathsf{com},\lambda}, \mathcal{R}^*_{\mathsf{open},\lambda}, \boldsymbol{\rho}_\lambda\}$ there exists a QPT equivocator $\mathcal{Q}_{\mathcal{R}^*} = (\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}, \mathcal{Q}_{\mathcal{R}^*,\mathsf{open}})$ such that for every polynomial-size quantum distinguisher $\mathcal{D}^* = \{\mathcal{D}^*_\lambda, \boldsymbol{\sigma}_\lambda\}$ there exists a negligible function $\nu(\cdot)$ such that for every bit $b \in \{0,1\}$,

$$\left|\Pr[\mathcal{D}^*(\boldsymbol{\sigma}_\lambda, \mathsf{Real}_b) = 1] - \Pr[\mathcal{D}^*(\boldsymbol{\sigma}_\lambda, \mathsf{Ideal}_b) = 1]\right| = \nu(\lambda)$$

for Real_b and Ideal_b distributions as defined below (where we omit indexing by λ for the sake of presentation).

- Real_b: Let $\mathcal{R}^*$ interact with $\mathcal{C}_{\mathsf{com}}(1^\lambda, b)$ in the commit phase, after which $\mathcal{C}_{\mathsf{com}}$ obtains state $\mathbf{x}_{\mathsf{com}}$ and $\mathcal{R}^*$ obtains state $\mathbf{y}_{\mathsf{com}}$. Next, obtain output signal bit $s \in \{0,1\}$ from $\mathcal{R}^*$, specifying whether or not the commit phase aborts. If $s = 0$, output $\mathbf{y}_{\mathsf{com}}$. Otherwise, output the final state $\mathbf{y}_{\mathsf{final}}$ of $\mathcal{R}^*$ after it interacts with $\mathcal{C}_{\mathsf{open}}(\mathbf{x}_{\mathsf{com}})$ in the open phase.
- Ideal_b: To define Ideal_b, first run the equivocator $\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}(\boldsymbol{\rho})$ to obtain state $(\mathbf{z}, \mathbf{y}_{\mathsf{com}})$ and signal bit $s \in \{0,1\}$. If $s = 0$, output $\mathbf{y}_{\mathsf{com}}$. Otherwise, obtain and output $\mathbf{y}_{\mathsf{final}} \leftarrow \mathcal{Q}_{\mathcal{R}^*,\mathsf{open}}(b, \mathbf{z}, \mathbf{y}_{\mathsf{com}})$.

Definition 3.5 (Equivocal and Extractable Commitments)**.** We say that a commitment scheme is equivocal and extractable if it satisfies Definitions 3.1, 3.3, and 3.4.

[4] In Naor's protocol, the receiver first sends a uniformly random $u \leftarrow \{0,1\}^{\lambda^3}$, and the committer commits to bit b by sending $(b \cdot u) \oplus G(s)$, where $s \leftarrow \{0,1\}^\lambda$ and G is a length-tripling PRG.

3.2 Oblivious Transfer with Quantum Communication

An oblivious transfer with quantum communication is a protocol between a quantum interactive sender $\mathcal{S}$ and a quantum interactive receiver $\mathcal{R}$, where the sender $\mathcal{S}$ has input $m_0, m_1 \in \{0,1\}^\lambda$ and the receiver $\mathcal{R}$ has input $b \in \{0,1\}$. After interaction the sender outputs (m_0, m_1) and the receiver outputs (b, m_b).

Let $\mathcal{F}(\cdot,\cdot)$ be the following functionality. $\mathcal{F}(b,\cdot)$ takes as input either (m_0, m_1) or abort from the sender, returns end to the sender, and outputs m_b to the receiver in the non-abort case and $\perp$ in the abort case. $\mathcal{F}(\cdot,(m_0,m_1))$ takes as input either b or abort from the receiver, returns m_b to the receiver, and returns end to the sender in the non-abort case, and returns $\perp$ to the sender in the abort case.

Definition 3.6. We let $\langle S(m_0,m_1), R(b)\rangle$ denote an execution of the OT protocol with sender input (m_0,m_1) and receiver input bit b. We denote by $\boldsymbol{\rho}_{\mathsf{out},S^*}\langle S^*(\boldsymbol{\rho}), R(b)\rangle$ and $\mathsf{OUT}_R\langle S^*(\boldsymbol{\rho}), R(b)\rangle$ the final state of a non-uniform malicious sender $S^*(\boldsymbol{\rho})$ and the output of the receiver $R(b)$ at the end of an interaction (leaving the indexing by λ implicit). We denote by $\boldsymbol{\rho}_{\mathsf{out},R^*}\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle$ and $\mathsf{OUT}_S\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle$ the final state of a non-uniform malicious receiver $R^*(\boldsymbol{\rho})$ and the output of the sender $S(m_0,m_1)$ at the end of an interaction. We require OT to satisfy the following security properties:

- **Receiver Security.** For every receiver bit $b \in \{0,1\}$ and every QPT non-uniform malicious sender $S^*(\boldsymbol{\rho})$, there exists a simulator Sim_{S^*} such that the following holds. $\mathsf{Sim}_{S^*}(\boldsymbol{\rho})$ sends inputs (m_0,m_1) or abort to the ideal functionality $\mathcal{F}_{\mathsf{OT}}(b,\cdot)$, whose output to the receiver is denoted by OUT_R. $\mathsf{Sim}_{S^*}(\boldsymbol{\rho})$ also outputs a final state $\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},S^*}$ such that

$$\left(\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},S^*}, \mathsf{OUT}_R\right) \approx_c \left(\boldsymbol{\rho}_{\mathsf{out},S^*}\langle S^*(\boldsymbol{\rho}), R(b)\rangle, \mathsf{OUT}_R\langle S^*(\boldsymbol{\rho}), R(b)\rangle\right).$$

- **Sender Security.** For every pair of sender inputs (m_0,m_1) and every QPT non-uniform malicious receiver $R^*(\boldsymbol{\rho})$, there exists a simulator Sim_{R^*} such that the following holds. $\mathsf{Sim}_{R^*}(\boldsymbol{\rho})$ sends bit b or abort to the ideal functionality $\mathcal{F}_{\mathsf{OT}}(m_0,m_1,\cdot)$, whose output to the sender is denoted by OUT_S. $\mathsf{Sim}_{R^*}(\boldsymbol{\rho})$ also outputs a final state $\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},R^*}$ such that

$$\left(\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},R^*}, \mathsf{OUT}_S\right) \approx_c \left(\boldsymbol{\rho}_{\mathsf{out},R^*}\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle, \mathsf{OUT}_S\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle\right).$$

3.3 Quantum Rewinding Lemma

We will make use of the following lemma from [Wat06].

Lemma 3.7. *Let $\mathcal{Q}$ be a general quantum circuit with n input qubits that outputs a classical bit b and m qubits. For an n-qubit state $|\psi\rangle$, let $p(|\psi\rangle)$ denote the probability that $b=0$ when executing $\mathcal{Q}$ on input $|\psi\rangle$. Let $p_0, q \in (0,1)$ and $\epsilon \in (0,1/2)$ be such that:*

- *For every n-qubit state $|\psi\rangle$, $p_0 \leq p(\psi)$,*
- *For every n-qubit state $|\psi\rangle$, $|p(\psi) - q| < \epsilon$,*
- *$p_0(1-p_0) \leq q(1-q)$,*

Then, there is a general quantum circuit $\widehat{\mathcal{Q}}$ of size $O\left(\frac{\log(1/\epsilon)}{4\cdot p_0(1-p_0)}|\mathcal{Q}|\right)$, taking as input n qubits, and returning as output m qubits, with the following guarantee. For an n qubit state $|\psi\rangle$, let $\mathcal{Q}_0(|\psi\rangle)$ denote the output of $\mathcal{Q}$ on input $|\psi\rangle$ conditioned on $b=0$, and let $\widehat{\mathcal{Q}}(|\psi\rangle)$ denote the output of $\widehat{\mathcal{Q}}$ on input $|\psi\rangle$. Then, for any n-qubit state $|\psi\rangle$,

$$\mathsf{TD}\left(\mathcal{Q}_0(|\psi\rangle), \widehat{\mathcal{Q}}(|\psi\rangle)\right) \leq 4\sqrt{\epsilon}\frac{\log(1/\epsilon)}{p_0(1-p_0)}.$$

3.4 Quantum Entropy and Leftover Hashing

Classical Min-Entropy. For a classical random variable X, its min-entropy $\mathbf{H}_\infty(X)$ is defined as

$$\mathbf{H}_\infty(X) \coloneqq -\log(\max_x \Pr[X = x]).$$

In cryptographic settings, we are often interested in the min-entropy of a random variable X sampled from a joint distribution (X, Y), where Y is side information available to an adversary/distinguisher. Following [RW05], we define the conditional min-entropy of X given Y as

$$\mathbf{H}_\infty(X \mid Y) \coloneqq -\log(\max_{x,y} \Pr[X = x \mid Y = y]).$$

That is, $\mathbf{H}_\infty(X \mid Y)$ is (the negative log of) the maximum probability of guessing the outcome of X, maximized over the possible outcomes of Y.

Quantum Conditional Min-Entropy. Let $\boldsymbol{\rho}_{\mathsf{XY}}$ denote a bipartite quantum state over registers XY. Following [Ren08, KRS09], the conditional min-entropy of $\boldsymbol{\rho}_{\mathsf{XY}}$ given Y is then defined to be

$$\mathbf{H}_\infty(\boldsymbol{\rho}_{\mathsf{XY}} \mid \mathsf{Y}) \coloneqq \sup_{\mathbf{y}} \max\{h \in \mathbb{R} : 2^{-h} \cdot I_{\mathsf{X}} \otimes \mathbf{y}_{\mathsf{Y}} - \boldsymbol{\rho}_{\mathsf{XY}} \geq 0\}.$$

In this work, we will exclusively consider the case where the $\boldsymbol{\rho}_{\mathsf{XY}}$ is a joint distribution of the form $(X, \mathbf{y})$ where X is a classical random variable. In other words, $\boldsymbol{\rho}_{\mathsf{XY}}$ can be written as

$$\sum_x \Pr[X = x] \, |x\rangle\langle x| \otimes \mathbf{y}_x.$$

In this case, we will write $\mathbf{H}_\infty(\boldsymbol{\rho}_{\mathsf{XY}} \mid \mathsf{Y})$ as $\mathbf{H}_\infty(X \mid \mathbf{y})$. We remark that in this particular setting, $\mathbf{H}_\infty(X \mid \mathbf{y})$ can be interpreted as the (negative log of) the maximum probability of guessing X given quantum state $\mathbf{y}$ [KRS09].

Leftover Hash Lemma with Quantum Side Information. We now state a generalization of the leftover hash lemma to the setting of quantum side information.

Lemma 3.8 ([RK05])**.** *Let $\mathcal{H}$ be a family of universal hash functions from $\mathcal{X}$ to $\{0,1\}^\ell$, i.e. for any $x \neq x'$, $\Pr_{h \leftarrow \mathcal{H}}[h(x) = h(x')] = 2^{-\ell}$. Then for joint random variables $(X, \mathbf{y})$ where X is a classical random variable over $\mathcal{X}$ and $\mathbf{y}$ is a quantum random variable,*

$$\|(h, h(X), \mathbf{y}) - (h, u, \mathbf{y})\|_1 \leq \frac{1}{2^{1+\frac{1}{2}(\mathbf{H}_\infty(X|\mathbf{y})-\ell)}},$$

where $h \leftarrow \mathcal{H}$ and $u \leftarrow \{0,1\}^\ell$.

4 A Quantum Equivocality Compiler

In this section, we show a generic black-box compiler that takes any quantum-secure commitment scheme and produces a quantum-secure *equivocal* commitment scheme.

The compiler is described in Protocol 1, where $(\mathsf{Commit}, \mathsf{Decommit})$ denotes some statistically binding and computationally hiding commitment scheme satisfying Definitions 3.1 and 3.2. We describe how to equivocally commit to a single bit, and note that commitment to an arbitrary length string follows by sequential repetition.

Furthermore, we show that if the underlying commitment $(\mathsf{Commit}, \mathsf{Decommit})$ is *extractable* according to Definition 3.3, then the resulting scheme is an extractable and equivocal commitment satisfying Definition 3.5.

These results are captured in the following theorems.

Protocol 1

Committer $\mathcal{C}$ Input: Bit $b \in \{0,1\}$.

The Protocol: Commit Phase

1. $\mathcal{C}$ samples uniformly random bits $d_{i,j}$ for $i \in [\lambda]$ and $j \in \{0,1\}$.
2. For every $i \in [\lambda]$, $\mathcal{C}$ and $\mathcal{R}$ sequentially perform the following steps.
   (a) $\mathcal{C}$ and $\mathcal{R}$ execute four sessions sequentially, namely:
   - $\mathbf{x}_{0,0}, \mathbf{y}_{0,0} \leftarrow \mathsf{Commit}\langle \mathcal{C}(d_{i,0}), \mathcal{R}\rangle$,
   - $\mathbf{x}_{0,1}, \mathbf{y}_{0,1} \leftarrow \mathsf{Commit}\langle \mathcal{C}(d_{i,0}), \mathcal{R}\rangle$,
   - $\mathbf{x}_{1,0}, \mathbf{y}_{1,0} \leftarrow \mathsf{Commit}\langle \mathcal{C}(d_{i,1}), \mathcal{R}\rangle$ and
   - $\mathbf{x}_{1,1}, \mathbf{y}_{1,1} \leftarrow \mathsf{Commit}\langle \mathcal{C}(d_{i,1}), \mathcal{R}\rangle$.

   (b) $\mathcal{R}$ sends a choice bit $c_i \leftarrow \{0,1\}$.
   (c) $\mathcal{C}$ and $\mathcal{R}$ execute two decommitments, obtaining the opened bits:
   - $u \leftarrow \mathsf{Decommit}\langle \mathcal{C}(\mathbf{x}_{c_i,0}), \mathcal{R}(\mathbf{y}_{c_i,0})\rangle$ and
   - $v \leftarrow \mathsf{Decommit}\langle \mathcal{C}(\mathbf{x}_{c_i,1}), \mathcal{R}(\mathbf{y}_{c_i,1})\rangle$.

   If $u \neq v$, $\mathcal{R}$ aborts. Otherwise, $\mathcal{C}$ and $\mathcal{R}$ continue.
3. For $i \in [\lambda]$, $\mathcal{C}$ sets $e_i = b \oplus d_{i,1-c_i}$ and sends $\{e_i\}_{i\in[\lambda]}$ to R.

The Protocol: Decommit Phase

1. $\mathcal{C}$ sends b to $\mathcal{R}$. In addition,
   - For $i \in [\lambda]$, $\mathcal{C}$ picks $\alpha_i \leftarrow \{0,1\}$ and sends it to $\mathcal{R}$.
   - $\mathcal{C}$ and $\mathcal{R}$ execute $\widehat{d_i} \leftarrow \mathsf{Decommit}\langle \mathcal{C}(\mathbf{x}_{1-c_i,\alpha_i}), \mathcal{R}(\mathbf{y}_{1-c_i,\alpha_i})\rangle$.
2. $\mathcal{R}$ accepts the decommitment and outputs b if for every $i \in [\lambda]$, $\widehat{d_i} = b \oplus e_i$.

Figure 1: Equivocal Bit Commitment.

Theorem 4.1. *Protocol 1 describes a quantum statistically binding and equivocal commitment (satisfying Definitions 3.2 and 3.4) whenever* Commit *is instantiated with any quantum statistically binding, computationally hiding commitment (satisfying Definitions 3.2 and 3.1).*

Theorem 4.2. *Protocol 1 describes a quantum extractable and equivocal commitment (satisfying Definition 3.5) whenever* Commit *is instantiated with any quantum extractable, computationally hiding commitment (satisfying Definitions 3.3 and 3.1).*

These theorems follows from establishing statistical binding, equivocality, and extractability of the commitment in Protocol 1.

Binding. We show that if Commit is statistically binding, then Protocol 1 is statistically binding. For any adversarial committer strategy, consider the λ unopened pairs of commitments after the commit phase. Since Commit is statistically binding, we can assume that each of the 2λ commitments is binding to a particular bit, except with negligible probability. Now, if any single pair contains binding commitments to the same bit d_i, then the committer will only be able to open its Protocol 1 commitment to the bit $d_i \oplus e_i$. Thus, to violate binding, the adversarial committer will have to have committed to different bits in each of the λ

unopened pairs. However, in this case, the committer will be caught and the receiver will abort except with probability $1/2^{\lambda}$.

Equivocality. The equivocal simulator $(\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}, \mathcal{Q}_{\mathcal{R}^*,\mathsf{open}})$ is obtained via the use of the quantum rewinding lemma (Lemma 3.7) [Wat06]. For the purposes of defining the simulation strategy, it will be sufficient (w.l.o.g.) to consider a restricted receiver $\mathcal{R}^*$ as follows, for the i^{th} sequential step of the protocol. In our simulation, the state of $\mathcal{R}^*$ will be initialized to the final state at the end of simulating the $(i-1)^{th}$ step.

1. $\mathcal{R}^*$ takes a quantum register W, representing its auxiliary quantum input. $\mathcal{R}^*$ will use two additional quantum registers that function as work space: V, which is an arbitrary (polynomial-size) register, and A, which is a single qubit register. The registers V and A are initialized to their all-zero states before the protocol begins.

2. Let M denote the polynomial-size register used by $\mathcal{C}$ to send messages to $\mathcal{R}^*$. After carrying out step 2(a) by running on registers $(\mathsf{W}, \mathsf{V}, \mathsf{A}, \mathsf{M})$, $\mathcal{R}^*$ measures the register A to obtain a bit c_i, for Step 2(b), which it sends back to $\mathcal{C}$.

3. Next, $\mathcal{R}^*$ computes the decommitment phases (with messages from $\mathcal{C}$ placed in register M) according to Step 2(c). $\mathcal{R}^*$ outputs registers $(\mathsf{W}, \mathsf{V}, \mathsf{A}, \mathsf{M})$.

Any polynomial-time quantum verifier can be modeled as a verifier of this restricted form followed by some polynomial-time post-processing of the restricted verifier's output. The same post-processing can be applied to the output of the simulator that will be constructed for the given restricted verifier.

Following [Wat06], we define a simulator that uses two additional registers, C and Z. C is a one qubit register, while Z is an auxiliary register used to implement the computation that will be described next. Consider a quantum procedure $\mathcal{Q}_{\mathsf{partial}}$ that implements the strategy described in Protocol 2 using these registers.

Protocol 2

Circuit $\mathcal{Q}_{\mathsf{partial}}$.

1. Sample a uniformly random classical bit $\widehat{c}$, and store it in register C.
2. Sample uniformly random bits (z, d).
3. If $\widehat{c} = 0$, initialize committer input as follows, corresponding to four sequential sessions:
   - For the first two sessions, set committer input to z.
   - For the third and fourth sessions, set committer input to d and $1-d$ respectively.
4. If $\widehat{c} = 1$, initialize committer input as follows, corresponding to four sequential sessions:
   - For the first and second sessions, set committer input to d and $1-d$ respectively.
   - For the last two sessions, set committer input to z.
5. Run the commitment phase interaction between the honest committer and $\mathcal{R}^*$'s sequence of unitaries on registers initialized as above.
6. Measure the qubit register A to obtain a bit c. If $c = \widehat{c}$, output 0, otherwise output 1.

Figure 2: Equivocal Simulator.

Next, we would like to apply the quantum rewinding lemma (Lemma 3.7) to the $\mathcal{Q}_{\mathsf{partial}}$ circuit. In order to do this, we will argue that the probability $p(\psi)$ that this circuit outputs 0 is such that $|p(\psi) - \frac{1}{2}| = \mathsf{negl}(\lambda)$, regardless of the auxiliary input $|\psi\rangle$ to $\mathcal{R}^*$. This follows from the fact that the commitments are (statistically/computationally) hiding. In more detail, by definition, Step 5 produces a distribution on the $\mathcal{R}^*$'s side that is identical to the distribution generated by $\mathcal{R}^*$ in its interaction with the committer. If $|p(\psi) - \frac{1}{2}|$ were non-negligible, then the sequence of unitaries applied by $\mathcal{R}^*$ could be used to distinguish commitments generated according to the case $\widehat{c} = 0$ from commitments generated according to the case $\widehat{c} = 1$, leading to a contradiction.

Now consider the state of the residual qubits of $\mathcal{Q}_{\mathsf{partial}}$ conditioned on a measurement of its output qubit being 0. The output state of the general quantum circuit $\widehat{\mathcal{Q}}$ resulting from applying quantum rewinding (Lemma 3.7) will have negligible trace distance from this state. This state is over all of the registers discussed above, so the simulator $\mathcal{Q}_{\mathsf{com},\mathcal{R}^*}$ must further process this state as follows:

- Measure the register C, obtaining challenge c.
- Compute decommitment information corresponding to challenge c, as in Step 2(c) of the protocol (recall that this information is stored in the message register M).
- Output registers $(\mathsf{W}, \mathsf{V}, \mathsf{A}, \mathsf{M})$. All remaining registers are traced out.

The simulator $\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}$ executes all i sequential interactions in this manner, and then samples $e_1, \ldots, e_\lambda \leftarrow \{0,1\}^\lambda$, as the committer messages for Step 3 of Protocol 1. It runs the receiver's unitary on the resulting protocol, and outputs the resulting registers $(\mathsf{W}, \mathsf{V}, \mathsf{A}, \mathsf{M})$. It additionally outputs private state $\mathsf{st} = (c_1, d_1, \ldots, c_\lambda, d_\lambda)$ where c_i, d_i were sampled during the ith execution of Protocol 2.

The simulator $\mathcal{Q}_{\mathcal{R}^*,\mathsf{open}}(b, \mathsf{st}, \mathbf{w}, \mathbf{v}, \mathbf{a}, \mathbf{m})$ parses st as $(c_1, d_1, \ldots, c_\lambda, d_\lambda)$. For every $i \in [\lambda]$ it does the following:

- Let $\widehat{d_i} = b \oplus e_i$.
- If $c_i = 0$, it executes the decommitment phase corresponding to the $((\widehat{d_i} \oplus d_i) + 2)^{th}$ session.
- If $c_i = 1$, it executes the decommitment phase corresponding to the $(\widehat{d_i} \oplus d_i)^{th}$ session.

$\mathcal{Q}_{\mathcal{R}^*,\mathsf{open}}$ then executes the receiver's algorithm on these decommitments and outputs the resulting state. Note that each decommitment will be to the bit $\widehat{d_i} = b \oplus e_i$.

To complete the proof of the equivocal property, we must establish that the distributions Real_b and Ideal_b, according to Definition 3.4, are indistinguishable. This follows from the (statistical/computational) hiding of the commitment scheme, via an identical argument to the one used above. In particular, if the equivocal simulator produces a distribution that is distinguishable from the real distribution, then there exists a session $i \in [\lambda]$ such that the distribution in the real and ideal experiments upto the $i-1^{th}$ session are indistinguishable, but upto the i^{th} session are distinguishable. This contradicts the above guarantee given by the quantum rewinding lemma, since for any i, the post-processed residual qubits of $\mathcal{Q}_{\mathsf{partial}}$ are indistinguishable from the state of $\mathcal{R}^*$ after the i'th sequential session in the real protocol (due to the hiding of the commitment scheme).

Extractability. Next, we prove that Protocol 1 satisfies Definition 3.3 as long as the underlying commitment $(\mathsf{Commit}, \mathsf{Decommit})$ is extractable according to Definition 3.3. Consider the following extractor $\mathcal{E}_{\mathcal{C}^*}$.

- For $i \in [\lambda]$:
  - Execute four sequential commitment sessions with $\mathcal{C}^*$, where the extractor of Commit is run on the first and third sessions, and the honest interaction is run for the second and fourth sessions. Obtain outputs $(\boldsymbol{\rho}_{\mathcal{C}^*}, \mathsf{st}_{\mathcal{R},i,0}, d'_{i,0}, \mathsf{st}_{\mathcal{R},i,0}, d'_{i,1})$, where $\boldsymbol{\rho}_{\mathcal{C}^*}$ is the final state of the committer after engaging in all four sequential sessions, and $\mathsf{st}_{\mathcal{R},i,0}$, $\mathsf{st}_{\mathcal{R},i,1}$ are receiver states output by the extractor corresponding to the first and third sessions.

- Corresponding to Step 2(b), compute and send $c_i \leftarrow \{0,1\}$.
- Execute Step 2(c) identically to Protocol 1.

- Executes Step 3 of Protocol 1, receiving bits $\{e_i\}_{i\in[\lambda]}$. Fix b^* to be the most frequently ocurring bit in $\{e_i \oplus d'_{i,1-c_i}\}_{i\in[\lambda]}$, and output the final state of $\mathcal{C}^*$, the receiver states $\{\mathsf{st}_{\mathcal{R},i,0}, \mathsf{st}_{\mathcal{R},i,1}\}_{i\in[\lambda]}$, and the extracted bit b^*.

Indistinguishability between the distributions Real and Ideal follows by definition of extractability of the underlying commitment $(\mathsf{Commit}, \mathsf{Decommit})$. In more detail, recall that Real denotes the distribution $(\rho_{\mathcal{C}^*,\mathsf{final}}, b)$ where $\rho_{\mathcal{C}^*,\mathsf{final}}$ denotes the final state of $\mathcal{C}^*$ and b the output of the receiver, and Ideal denotes the final committer state and opened bit after the opening phase of the scheme is run on the output of the extractor. Now for every commitment strategy, every $i \in [\lambda]$, the probability that $d'_{i,1-c_i}$ is not equal to the other bit committed in its pair, and yet the receiver does not abort in Step 2(c) in the i^{th} sequential repetition, is $\leq \frac{1}{2}+\mathsf{negl}(\lambda)$. Thus, by the correctness of the extractor, this implies that the probability that an adversarial committer opens to $1-b^*$ is at most $1/2^{\lambda/2}+\mathsf{negl}(\lambda) = \mathsf{negl}(\lambda)$. Finally, by indistinguishability of the extractor's and committer's states, this implies that the distributions Real and Ideal are indistinguishable.

5 Quantum Extractable Commitments

We construct extractable commitments by making use of the following building blocks.

- We let $(\mathsf{EqCommit}, \mathsf{EqDecommit})$ denote any quantum statistically binding and equivocal commitment scheme, satisfying Definition 3.2 and Definition 3.4. Such a commitment can be obtained by applying the compiler from last section to Naor's commitment scheme [Nao91].
- For a suitable polynomial $k(\cdot)$, let $h : \{0,1\}^{k(\lambda)} \times \{0,1\}^{\lambda^2} \to \{0,1\}$ be a universal hash function that is evaluated on a random seed $s \in \{0,1\}^{k(\lambda)}$ and input $x \in \{0,1\}^{\lambda^2}$.

Our extractable commitment scheme is described formally in Figure 3. It is shown how to commit to a single bit, though commitment to any arbitrary length string follows by sequential repetition. Correctness of the protocol follows by inspection. In the remainder of this section, we prove the following theorem.

Theorem 5.1. *Protocol 3 describes a quantum statistically hiding and extractable commitment (satisfying Definition 3.2 and Definition 3.3) whenever* $(\mathsf{EqCommit}, \mathsf{EqDecommit})$ *is instantiated with any quantum statistically binding and equivocal commitment scheme (satisfying Definition 3.2 and Definition 3.4).*

Proof. Throughout, we will consider non-uniform adversaries, but drop the indexing by λ.

Extractability. Consider any adversarial committer $\mathcal{C}^*$ with advice $\boldsymbol{\rho}$. The extractor $\mathcal{E}_{\mathcal{C}^*}(\boldsymbol{\rho})$ is constructed as follows.

1. Run the first message algorithm of $\mathcal{C}^*$ on input $\boldsymbol{\rho}$, obtaining message $\boldsymbol{\psi}$.
2. For $i \in [2\lambda^3]$, sequentially execute equivocal commitment sessions with the equivocal simulator $\mathcal{Q}_{R^*,\mathsf{com}}$, where R^* is the part of $\mathcal{C}^*$ that participates as receiver in the i^{th} session. Session i results in output $(\mathbf{z}_i, \mathbf{y}_{\mathsf{com},i})$, where $\mathbf{z}_i$ is stored by the extractor, and $\mathbf{y}_{\mathsf{com},i}$ is the current state of $\mathcal{C}^*$, which is fed as input into the next session.
3. Obtain T from $\mathcal{C}^*$, and sample $\widehat{\theta} \leftarrow \{+,\times\}^{2\lambda^3}$. Let $\boldsymbol{\psi}_i$ denote the i^{th} qubit of $\boldsymbol{\psi}$, and measure the qubits $\boldsymbol{\psi}_i$ for $i \in T$, each in basis $\widehat{\theta}_i$. Let $\{\widehat{x}_i\}_{i\in[T]}$ be the results of the measurements.
4. Let $\mathbf{x}_{\mathsf{com}}$ be the current state of $\mathcal{C}^*$. For each $i \in [T]$, execute $\mathcal{Q}_{R^*,\mathsf{open}}((\widehat{\theta}_i, \widehat{x}_i), \mathbf{z}_i, \mathbf{x}_{\mathsf{com}})$, where R^* is the part of $\mathcal{C}^*$ that participates in the i^{th} opening, and $\mathbf{x}_{\mathsf{com}}$ is updated to be the current state of $\mathcal{C}^*$ after each sequential session.

Protocol 3

Committer $\mathcal{C}$ Input: Bit $b \in \{0,1\}$.

The Protocol: Commit Phase.

1. $\mathcal{C}$ chooses $x \leftarrow \{0,1\}^{2\lambda^3}$, $\theta \leftarrow \{+,\times\}^{2\lambda^3}$ and sends $|x\rangle_\theta$ to $\mathcal{R}$.
2. $\mathcal{R}$ chooses $\widehat{\theta} \leftarrow \{+,\times\}^{2\lambda^3}$ and obtains $\widehat{x} \in \{0,1\}^{2\lambda^3}$ by measuring $|x\rangle_\theta$ in basis $\widehat{\theta}$.

   $\mathcal{R}$ commits to $\widehat{\theta}$ and $\widehat{x}$ position-wise: $\mathcal{R}$ and $\mathcal{C}$ execute sequentially $2\lambda^3$ equivocal commitment sessions with $\mathcal{R}$ as committer and $\mathcal{C}$ as receiver. That is, for each $i \in [2\lambda^3]$, they compute $(\mathbf{x}_{\mathsf{com},i}, \mathbf{y}_{\mathsf{com},i}) \leftarrow \mathsf{EqCommit}\langle \mathcal{R}(\widehat{\theta}_i, \widehat{x}_i), \mathcal{C}\rangle$.
3. $\mathcal{C}$ sends a random test subset $T \subset [2\lambda^3]$ of size λ^3 to $\mathcal{R}$.
4. For every $i \in T$, $\mathcal{R}$ and $\mathcal{C}$ engage in $(\widehat{\theta}_i, \widehat{x}_i) \leftarrow \mathsf{EqDecommit}\langle \mathcal{R}(\mathbf{x}_{\mathsf{com},i}), \mathcal{C}(\mathbf{y}_{\mathsf{com},i})\rangle$, and $\mathcal{C}$ aborts if any commitment fails to open.
5. $\mathcal{C}$ checks that $x_i = \widehat{x}_i$ whenever $\theta_i = \widehat{\theta}_i$. If all tests pass, $\mathcal{C}$ proceeds with the protocol, otherwise, $\mathcal{C}$ aborts.
6. The tested positions are discarded by both parties: $\mathcal{C}$ and $\mathcal{R}$ restrict x and θ, respectively $\widehat{x}$ and $\widehat{\theta}$, to the λ^3 indices $i \in \overline{T}$. $\mathcal{C}$ sends θ to $\mathcal{R}$.
7. $\mathcal{C}$ partitions the remaining λ^3 bits of x into λ different λ^2-bit strings $y_1, \ldots, y_\lambda$. For each $i \in [\lambda]$, sample a seed $s_i \leftarrow \{0,1\}^{k(\lambda)}$ and compute $d_i \coloneqq h(s_i, y_i)$. Then output $\{s_i, b \oplus d_i\}_{i\in[\lambda]}$.

The Protocol: Decommit Phase.

1. $\mathcal{C}$ sends b and $(y_1, \ldots, y_\lambda)$ to $\mathcal{R}$.
2. If either of the following fails, $\mathcal{R}$ rejects and outputs $\perp$. Otherwise, $\mathcal{R}$ accepts and outputs b.
   - Let $\{s_i, v_i\}_{i\in[\lambda]}$ be the message received by $\mathcal{R}$ in step 7. Check that for all $i \in [\lambda]$, $v_i = b \oplus h(s_i, y_i)$.
   - Let $x = (y_1, \ldots, y_\lambda)$. For each $j \in [\lambda^3]$ such that $\widehat{\theta}_j = \theta_j$, check that $\widehat{x}_j = x_j$.

Figure 3: Extractable Commitment.

5. If $\mathcal{C}^*$ aborts at any point, abort and output $\perp$, otherwise continue.
6. Discard tested positions and restrict $\widehat{\theta}$ to the indices in $\overline{T}$. Obtain $\theta \in \{+,\times\}^{\lambda^3}$ from $\mathcal{C}^*$. Measure the qubits $\boldsymbol{\psi}_i$ in basis θ_i to obtain $\widehat{x}_i$ for $i \in \overline{T}$, and then partition $\widehat{x}$ into λ different λ^2-bit strings $\widehat{y}_1, \ldots, \widehat{y}_\lambda$.
7. Obtain $\{s_i, v_i\}_{i\in[\lambda]}$ from $\mathcal{C}^*$. Let b^* be the most frequently occurring bit in $\{h(s_i, \widehat{y}_i) \oplus v_i\}_{i\in[\lambda]}$. Output $(\mathbf{x}_{\mathsf{com}}, \mathbf{y}_{\mathsf{com}}, b^*)$, where $\mathbf{x}_{\mathsf{com}}$ is the resulting state of $\mathcal{C}^*$ and $\mathbf{y}_{\mathsf{com}} = (\theta, \widehat{\theta}, \widehat{x})$.

We now prove that $\mathcal{E}_{\mathcal{C}^*}$ satisfies the indistinguishability between Real and Ideal distributions as required by Definition 3.3. We do so via a sequence of hybrids.

Hyb_1. Define distribution Hyb_1 identically to Hyb_0, except that in Step 2, for $i \in [2\lambda^3]$, sequentially execute equivocal commitment sessions using the equivocal simulator $\mathcal{Q}_{R^*,\mathsf{com}}$, as described in the extractor. In Step

4, for every $i \in T$, open the i'th commitment to $(\widehat{\theta}_i, \widehat{x}_i)$ using $\mathcal{Q}_{R^*,\mathsf{open}}$, as described in the extractor.

By the equivocal property of Commit, for any QPT distinguisher $(\mathcal{D}^*, \boldsymbol{\sigma})$, there exists a negligible function $\nu(\cdot)$ such that

$$\left| \Pr[\mathcal{D}^*(\boldsymbol{\sigma}, \mathsf{Hyb}_1) = 1] - \Pr[\mathcal{D}^*(\boldsymbol{\sigma}, \mathsf{Hyb}_0) = 1] \right| = \nu(\lambda).$$

Hyb_2**.** This is identical to Hyb_1, except that the verifier measures qubits of $|x\rangle_\theta$ *only after* obtaining a description of the set T, and *only measures* the qubits $i \in [T]$. The output of this experiment is identical to Hyb_1, therefore for any QPT distinguisher $(\mathcal{D}^*, \boldsymbol{\sigma})$,

$$\Pr[\mathcal{D}^*(\boldsymbol{\sigma}, \mathsf{Hyb}_3) = 1] = \Pr[\mathcal{D}^*(\boldsymbol{\sigma}, \mathsf{Hyb}_2) = 1].$$

Moreover, the only difference between Hyb_2 and Ideal is that Ideal outputs FAIL when the message b opened by $\mathcal{C}^*$ is not $\perp$ and differs from the one extracted by $\mathcal{E}_{\mathcal{C}^*}$. Therefore, to derive a contradiction it will suffice to prove that there exists a negligible function $\nu(\cdot)$ such that

$$\Pr[\mathsf{FAIL}|\mathsf{Ideal}] = \nu(\lambda).$$

Consider any sender $\mathcal{C}^*$ that produces a committer state $\mathbf{x}_{\mathsf{com}}$ and then decommits to message b' using strings $(y_1, \ldots, y_\lambda)$ during the decommit phase. Let $T' \subseteq [\lambda]$ denote the set of all indices $i \in [\lambda]$ such that the corresponding $y_i \neq \widehat{y}_i$, where $\widehat{y}_i$ denotes the values obtained by the extractor in Step 6. Then we have the following claim.

Claim 5.2. *There exists a negligible function $\nu(\cdot)$ such that*

$$\Pr[|T'| > \lambda/2] = \nu(\lambda)$$

where the probability is over the randomness of the extractor.

Proof. For every $i \in [\lambda]$, we have that (over the randomness of the extractor):

$$\Pr\left[\mathcal{R}_{\mathsf{open}}(\mathbf{y}_{\mathsf{com}}) \text{ outputs } \perp \text{ in } \langle \mathcal{C}^*_{\mathsf{open}}(\mathbf{x}_{\mathsf{com}}), \mathcal{R}_{\mathsf{open}}(\mathbf{y}_{\mathsf{com}}) \rangle \;\middle|\; y_i \neq \widehat{y}_i \right] \geq \frac{1}{2}.$$

Indeed, the receiver will reject if for some position i for which $y_i \neq \widehat{y}_i$, it holds that $\theta_i = \widehat{\theta}_i$. Since $\widehat{\theta}$ was sampled uniformly at random, this will occur with probability $1/2$. This implies that $\Pr[|T'| > \lambda/2] \leq \frac{1}{2^{\lambda/2}}$, and the claim follows. □

By construction of $\mathcal{E}_{\mathcal{C}^*}$, $\Pr[\mathsf{FAIL}|\mathsf{Ideal}] < \Pr[|T'| > \lambda/2]$, and therefore it follows that there exists a negligble function $\nu(\cdot)$ such that

$$\Pr[\mathsf{FAIL}|\mathsf{Ideal}] = \nu(\lambda).$$

Statistical Hiding. We now prove that the protocol in Fig. 3 is statistically hiding. This follows from the statistical binding of the commitment scheme $\mathsf{EqCommit}$.

In more detail, we show that for any (unbounded) receiver strategy $\mathcal{R}^*$, the λ bits d_i computed by $\mathcal{C}$ in Step 7, Fig. 3 are negligibly close to uniformly random bits from the perspective of $\mathcal{R}^*$. This suffices to show that the committer's bit b is statistically hidden.

To show this, we use $\mathcal{R}^*$ to construct an algorithm B that interacts in the EPR protocol (Protocol 5). From $\mathcal{R}^*$'s perspective, this interaction will be (statistically) indistinguishable from an interaction with the honest committer $\mathcal{C}$. B is constructed as follows.

- B forwards the state received from A to $\mathcal{R}^*$. B then takes the role of a receiver, interacting with $\mathcal{R}^*$ as $\mathcal{R}^*$ commits to each bit of $\widehat{\theta}, \widehat{x}$. It inefficiently (brute force) extracts $\widehat{\theta}, \widehat{x}$ from these sessions and forwards $\widehat{\theta}, \widehat{x}$ to A.

- B forwards the test subset T received from A to $\mathcal{R}^*$. B then interacts with $\mathcal{R}^*$ to open the i'th commitments for $i \in [T]$. It sends continue to A if all commitments open and the opened messages $\{\widehat{\theta}_i, \widehat{x}_i\}_{i\in[T]}$ match the previously extracted strings $\widehat{\theta}, \widehat{x}$. Otherwise, it sends abort.
- In the post-processing phase, A follows the honest committer $\mathcal{C}$ in Protocol 3. That is, it partitions x into $y_1, \ldots, y_\lambda$, samples $\{s_i\}_{i\in[\lambda]}$, computes $\{d_i := h(s_i, y_i)\}_{i\in[\lambda]}$ and outputs $\{s_i, b \oplus d_i\}_{i\in[\lambda]}$.

There are a few differences between the view of $\mathcal{R}^*$ in the above "simulated" protocol and the view of $\mathcal{R}^*$ in the real protocol.

1. In the first round, $\mathcal{R}^*$ receives halves of EPR pairs in the simulated protocol, but this is identical to the real protocol since the state sent by $\mathcal{C}$ in the first round is maximally mixed from $\mathcal{R}$'s perspective.
2. In the simulated protocol, there is an additional check performed on the messages sent by $\mathcal{R}^*$, namely, that the opened messages $\{\widehat{\theta}_i, \widehat{x}_i\}_{i\in[T]}$ match previously extracted strings $\widehat{\theta}, \widehat{x}$. Since the commitment is statistically binding, there is only a negligible probability that this check fails (and B aborts) in the simulated protocol, but $\mathcal{C}$ does not abort in the real protocol.
3. In the simulated protocol, A measures her half of the EPR pairs in T according to the basis $\widehat{\theta}$, checking that the result matches $\widehat{x}$ whenever $\widehat{\theta}_i = \theta_i$. Due to the aforementioned check on the ZK argument instance, this check is computed with respect to the same $\widehat{\theta}, \widehat{x}$ as in the real protocol. Moreover, there is no difference from $\mathcal{R}^*$'s perspective whether, for $i \in [T]$ such that $\widehat{\theta}_i = \theta_i$, it received a qubit already in basis $\widehat{\theta}_i$, or it received half of an EPR pair, the other half of which was then subsequently measured in basis $\widehat{\theta}_i$. For the indices i for which $\widehat{\theta}_i \neq \theta_i$, $\mathcal{R}^*$ simply sees a maximally mixed state in both protocols.

Thus, by Lemma A.1, each $y_1, \ldots, y_\lambda$ has min-entropy $\Omega(\lambda^2)$ conditioned on $\mathcal{R}^*$'s view (and each other), except with probability $\mathrm{negl}(\lambda^3\lambda^2/\lambda^4) = \mathrm{negl}(\lambda)$. This follows because, by Hoeffding's inequality, the number of positions that θ and $\widehat{\theta}$ differ in each of the λ subsets of size λ^2 is $\Omega(\lambda^2)$, except with negligible probability. Finally, Lemma 3.8 shows that each d_i is negligibly close to a uniformly random bit, completing the proof. □

The following corollary follows immediately from Theorems 4.1, 4.2 and 5.1.

Corollary 5.3. *Extractable and equivocal commitments satisfying Definition 3.5 can be based on black-box use of statistically binding bit commitments, or on black-box use of quantum-hard one-way functions.*

6 Quantum Oblivious Transfer from Extractable and Equivocal Commitments

We construct simulation-secure quantum oblivious transfer by making use of the following building blocks.

- Let $(\mathsf{EECommit}, \mathsf{EEDecommit})$ denote any extractable and equivocal bit commitment satisfying Definition 3.5. Such a commitment scheme may be obtained by applying the compiler from Section 4 to the extractable commitment constructed in Section 5.
- Let $\{h_p\}_{p\in[\lambda,7\lambda]}$ be a family of universal hash functions, where for suitable polynomial $k(\cdot)$, $h_p : \{0,1\}^{k(\lambda)} \times \{0,1\}^p \to \{0,1\}^\lambda$.

Our QOT protocol is described in Protocol 4, which is essentially the [BBCS92] protocol instantiated with our extractable and equivocal commitment scheme.

Theorem 6.1. *The protocol in Figure 4 is a QOT protocol satisfying Definition 3.6 whenever* $(\mathsf{EECommit}, \mathsf{EEDecommit})$ *is instantiated with any extractable and equivocal commitment that satisfies Definition 3.5.*

Proof. We prove that the resulting QOT protocol satisfies the following properties.

Protocol 4

Sender S Input: Messages $m_0, m_1 \in \{0,1\}^\lambda \times \{0,1\}^\lambda$
Receiver R Input: Bit $b \in \{0,1\}$

The Protocol:

1. S chooses $x \leftarrow \{0,1\}^{16\lambda}$ and $\theta \leftarrow \{+,\times\}^{16\lambda}$ and sends $|x\rangle_\theta$ to R.
2. R chooses $\widehat{\theta} \leftarrow \{+,\times\}^{16\lambda}$ and obtains $\widehat{x} \in \{0,1\}^{16\lambda}$ by measuring $|x\rangle_\theta$ in basis $\widehat{\theta}$. Then, S and R execute 16λ sessions of $\mathsf{EECommit}$ sequentially with R acting as committer and S as receiver. In session i, R commits to the bits $\widehat{\theta}_i, \widehat{x}_i$.
3. S sends a random test subset $T \subset [16\lambda]$ of size 8λ to R.
4. For each $i \in T$, R and S sequentially execute the i'th $\mathsf{EEDecommit}$, after which S receives the opened bits $\widehat{\theta}_i, \widehat{x}_i$.
5. S checks that $x_i = \widehat{x}_i$ whenever $\theta_i = \widehat{\theta}_i$. If all tests pass, S accepts, otherwise, S rejects and aborts.
6. The tested positions are discarded by both parties: S and R restrict x and θ, respectively $\widehat{\theta}$ and $\widehat{x}$, to the 8λ indices $i \in \overline{T}$. S sends θ to R.
7. R partitions the positions of $\overline{T}$ into two parts: the "good" subset $I_b = \{i : \theta_i = \widehat{\theta}_i\}$ and the "bad" subset $I_{1-b} = \{i : \theta_i \neq \widehat{\theta}_i\}$. R aborts if $|I_0| < \lambda$ or $|I_1| < \lambda$. Otherwise, R sends (I_0, I_1) to S.
8. S samples seeds $s_0, s_1 \leftarrow \{0,1\}^{k(\lambda)}$ and sends $\big(s_0, h_{|I_0|}(s_0, x_0) \oplus m_0, s_1, h_{|I_1|}(s_1, x_1) \oplus m_1\big)$, where x_0 is x restricted to the set of indices I_0 and x_1 is x restricted to the set of indices I_1.
9. R decrypts s_b using $\widehat{x}_b$, the string $\widehat{x}$ restricted to the set of indices I_b.

Figure 4: Quantum Oblivious Transfer.

Receiver Security. Consider any adversarial sender S^* with advice $\boldsymbol{\rho}$. The simulator $\mathsf{Sim}_{S^*}(\boldsymbol{\rho})$ is constructed as follows.

1. Run the first message algorithm of S^* on input $\boldsymbol{\rho}$ to obtain message $\boldsymbol{\psi}$.
2. Execute 16λ sequential sessions of $\mathsf{EECommit}$. In each session, run the equivocator $\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}$, where $\mathcal{R}^*$ denotes the portion of S^* that participates as receiver in the i^{th} sequential $\mathsf{EECommit}$ session.
3. Obtain test subset T of size 8λ from S^*.
4. For each $i \in T$, sample $\widehat{\theta}_i \leftarrow \{+,\times\}$. Obtain $\widehat{x}_i$ by measuring the i^{th} qubit of $\boldsymbol{\psi}$ in basis $\widehat{\theta}_i$. For each $i \in T$, sequentially execute the equivocal simulator $\mathcal{Q}_{\mathcal{R}^*,\mathsf{open}}$ on input $(\widehat{\theta}_i, \widehat{x}_i)$ and the state obtained from $\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}$.
5. If S^* continues, discard positions indexed by T. Obtain θ_i for $i \in \overline{T}$ from S^*, and compute x_i for $i \in \overline{T}$ by measuring the i^{th} qubit of $\boldsymbol{\psi}$ in basis θ_i.
6. For every $i \in \overline{T}$, sample bit $d_i \leftarrow \{0,1\}$. Partition the set $\overline{T}$ into two subsets (I_0, I_1), where for every $i \in \overline{T}$, place $i \in I_0$ if $d = 0$ and otherwise place $i \in I_1$. Abort if $|I_0| < \lambda$ or $|I_1| < \lambda$. Send (I_0, I_1) to S.

7. Obtain (y_0, y_1) from S. Set x_0 to be x restricted to the set of indices I_0 and x_1 to be x restricted to the set of indices I_1. For $b \in \{0,1\}$, parse $y_b = (s_b, t_b)$ and compute $m_b = t_b \oplus h_{|I_b|}(s_b, x_b)$.

8. If S^* aborts anywhere in the process, send abort to the ideal functionality. Otherwise, send (m_0, m_1) to the ideal functionality. Output the final state of S^*.

Next (recalling notation from Definition 3.6), we will establish receiver security by proving that for each $b \in \{0,1\}$,

$$\left(\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},S^*}, \mathsf{OUT}_R\right) \approx_c \left(\boldsymbol{\rho}_{\mathsf{out},S^*}\langle S^*(\boldsymbol{\rho}), R(b)\rangle, \mathsf{OUT}_R\langle S^*(\boldsymbol{\rho}), R(b)\rangle\right).$$

Towards a contradiction, suppose there exists a bit $b \in \{0,1\}$, a non-uniform QPT sender $(S^*, \boldsymbol{\rho})$, a non-uniform QPT distinguisher $(D^*, \boldsymbol{\sigma})$, and polynomial $\mathsf{poly}(\cdot)$ such that

$$\Big|\Pr\left[D^*\left(\boldsymbol{\sigma}, \left(\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},S^*}, \mathsf{OUT}_R\right)\right) = 1\right] - \Pr\left[D^*\left(\boldsymbol{\sigma}, \left(\boldsymbol{\rho}_{\mathsf{out},S^*}\langle S^*(\boldsymbol{\rho}), R(b)\rangle, \mathsf{OUT}_R\langle S^*(\boldsymbol{\rho}), R(b)\rangle\right)\right) = 1\right]\Big| \geq \frac{1}{\mathsf{poly}(\lambda)}.$$

Fix any such b, sender $(S^*, \boldsymbol{\rho})$ and distinguisher $(D^*, \boldsymbol{\sigma})$. We derive a contradiction via an intermediate hybrid experiment, defined as follows with respect to bit b and sender $(S^*, \boldsymbol{\rho})$.

Hyb. In this hybrid, we generate the QOT receiver commitments via the equivocal simulator $\mathcal{Q}_{\mathcal{R}^*}$ (where $\mathcal{R}^*$ is derived from the malicious QOT sender S^*), and otherwise follow the honest QOT receiver's algorithm.

1. Run the first message algorithm of S^* on input $\boldsymbol{\rho}$ to obtain message $\boldsymbol{\psi}$.

2. Choose $\widehat{\theta} \leftarrow \{+,\times\}^{16\lambda}$ and obtain $\widehat{x} \in \{0,1\}^{16\lambda}$ by measuring $\boldsymbol{\psi}$ in basis $\widehat{\theta}$. Execute 16λ sequential sessions of $\mathsf{EECommit}$. In each session, run the equivocator $\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}$, where $\mathcal{R}^*$ denotes the portion of S^* that participates as receiver in the i^{th} sequential $\mathsf{EECommit}$ session.

3. Obtain test subset T of size 8λ from S^*.

4. For each $i \in T$, sequentially execute the equivocal simulator $\mathcal{Q}_{\mathcal{R}^*,\mathsf{open}}$ on input $\widehat{\theta}_i, \widehat{x}_i$ and the state obtained from $\mathcal{Q}_{\mathcal{R}^*,\mathsf{com}}$.

5. If S^* continues, discard positions indexed by T. Obtain θ_i for $i \in \overline{T}$ from S^*.

6. Partition the set $\overline{T}$ into two subsets: the "good" subset $I_b = \{i : \theta_i = \widehat{\theta}_i\}$ and the "bad" subset $I_{1-b} = \{i : \theta_i \neq \widehat{\theta}_i\}$. Abort if $|I_0| < \lambda$ or $|I_1| < \lambda$. Otherwise, send (I_0, I_1) to S.

7. Obtain (y_0, y_1) from S. Set x_b to be $\widehat{x}$ restricted to the set of indices I_b, and compute and set $m_b = t_b \oplus h_{|I_b|}(s_b, x_b)$. If S^* aborts anywhere in the process, let $\perp$ be the output of the receiver, otherwise let m_b be the output of the receiver.

The output of Hyb is the joint distribution of the final state of S^* and the output of the receiver.

Receiver security then follows from the following two claims.

Claim 6.2.

$$\Pr\left[D^*\left(\boldsymbol{\sigma}, \left(\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},S^*}, \mathsf{OUT}_R\right)\right) = 1\right] \equiv \Pr\left[D^*(\boldsymbol{\sigma}, \mathsf{Hyb}) = 1\right].$$

Proof. The only differences in the simulated distribution are (1) that measurements of S^*'s initial message $\boldsymbol{\psi}$ are delayed (which cannot be noticed by S^*), and (2) a syntactic difference in that the ideal functionality is queried to produce the receiver's output. □

Claim 6.3. *There exists a negligible function $\nu(\cdot)$ such that*

$$\left|\Pr[D^*(\boldsymbol{\sigma}, \mathsf{Hyb}) = 1] - \Pr\left[D^*\left(\boldsymbol{\sigma}, \left(\boldsymbol{\rho}_{\mathsf{out},S^*}\langle S^*(\boldsymbol{\rho}), R(b)\rangle, \mathsf{OUT}_R\langle S^*(\boldsymbol{\rho}), R(b)\rangle\right)\right) = 1\right]\right| = \nu(\lambda).$$

Proof. The only difference between the two distributions is that in the first, the receiver generates commitments according to the honest commit algorithms of $\mathsf{EECommit}$ while in the second, commitments in step 2 are generated via the equivocal simulator $\mathcal{Q}_{\mathcal{R}^*}$ of $\mathsf{EECommit}$. Therefore, this claim follows by the equivocal property of $(\mathsf{EECommit}, \mathsf{EEDecommit})$ (Definition 3.4). □

Sender Security. Consider any adversarial receiver R^* with advice $\boldsymbol{\rho}$. The simulator $\mathsf{Sim}_{R^*}(\boldsymbol{\rho})$ is constructed as follows:

1. Choose $x \leftarrow \{0,1\}^{16\lambda}$ and $\theta \leftarrow \{+,\times\}^{16\lambda}$. Send $|x\rangle_\theta$ to R^*.
2. Execute 16λ sequential sessions of $\mathsf{EECommit}$. In the i^{th} session for $i \in [16\lambda]$, run the extractor $\mathcal{E}_{\mathcal{C}_i^*,\mathsf{com}}$ where $\mathcal{C}_i^*$ denotes the portion of R^* that participates as committer in the i^{th} sequential $\mathsf{EECommit}$ session. Obtain from $\mathcal{E}_{\mathcal{C}_i^*,\mathsf{com}}$ the extracted values $(\widehat{\theta}_i, \widehat{x}_i)$.
3. Sample and send a random test subset $T \subset [16\lambda]$ of size 8λ to R^*.
4. Execute the T opening phases $\mathsf{EEDecommit}$ with R^*, and abort if any fail. Discard the opened values.
5. Check that $x_i = \widehat{x}_i$ whenever $\theta_i = \widehat{\theta}_i$, and if not abort.
6. Restrict x and θ to $i \in \overline{T}$ and send θ to R^*.
7. Obtain R^*'s message and parse it as consisting of two sets I_0 and I_1. Let S be the set of indices in I_0 such that $\theta_i \neq \widehat{\theta}_i$. If $|S| \geq 3\lambda/2$ then set $b = 1$ and otherwise set $b = 0$.
8. Obtain the output m_b of $\mathcal{F}_{\mathsf{OT}}$ on input bit b. Set $m_{1-b} = 0^\lambda$.
9. Sample seeds $s_0, s_1 \leftarrow \{0,1\}^{k(\lambda)}$ and send $s_0, h_{|I_0|}(s_0, x_0) \oplus m_0, s_1, h_{|I_1|}(s_1, x_1) \oplus m_1$, where x_0 is x restricted to the set of indices I_0 and x_1 is x restricted to the set of indices I_1. Output the final state of R^*.

We will now establish sender security by proving that

$$\left(\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},R^*}, \mathsf{OUT}_S\right) \approx_c \left(\boldsymbol{\rho}_{\mathsf{out},R^*}\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle, \mathsf{OUT}_S\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle\right).$$

Towards a contradiction, suppose there exists a pair of messages $(m_0, m_1) \in \{0,1\}^\lambda$, a non-uniform QPT receiver $(R^*, \boldsymbol{\rho})$, a non-uniform QPT distinguisher $(D^*, \boldsymbol{\sigma})$, and a polynomial $\mathsf{poly}(\cdot)$ such that

$$\Big|\Pr\left[D^*\left(\boldsymbol{\sigma}, \left(\boldsymbol{\rho}_{\mathsf{Sim},\mathsf{out},R^*}, \mathsf{OUT}_S\right)\right) = 1\right] - \Pr\left[D^*\left(\boldsymbol{\sigma}, \left(\boldsymbol{\rho}_{\mathsf{out},R^*}\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle, \mathsf{OUT}_S\langle S(m_0,m_1), R^*(\boldsymbol{\rho})\rangle\right)\right) = 1\right]\Big| \geq \frac{1}{\mathsf{poly}(\lambda)}.$$

Fix any such (m_0, m_1), receiver $(R^*, \boldsymbol{\rho})$ and distinguisher $(D^*, \boldsymbol{\sigma})$. We derive a contradiction via an intermediate hybrid experiment, defined as follows. Like the simulator, this hybrid simulates and extracts from the receiver's commitments, but does not yet replace the sender string m_{1-b} with 0^λ.

Hyb. Consider any adversarial receiver R^* with advice $\boldsymbol{\rho}$. The output of Hyb depends on (m_0, m_1) and is generated as follows.

1. Choose $x \leftarrow \{0,1\}^{16\lambda}$ and $\theta \leftarrow \{+,\times\}^{16\lambda}$. Send $|x\rangle_\theta$ to R^*.
2. Execute 16λ sequential sessions of $\mathsf{EECommit}$. In the i^{th} session for $i \in [16\lambda]$, run the extractor $\mathcal{E}_{\mathcal{C}_i^*,\mathsf{com}}$ where $\mathcal{C}_i^*$ denotes the portion of R^* that participates as committer in the i^{th} sequential $\mathsf{EECommit}$ session. Obtain from $\mathcal{E}_{\mathcal{C}_i^*,\mathsf{com}}$ the extracted values $\widehat{\theta}_i, \widehat{x}_i$.
3. Sample and send a random test subset $T \subset [16\lambda]$ of size 8λ to R^*.

4. Execute the T opening phases $\mathsf{EEDecommit}$ with R^*, and abort if any fail. Discard the opened values.
5. Check that $x_i = \widehat{x}_i$ whenever $\theta_i = \widehat{\theta}_i$, and if not abort.
6. Obtain R^*'s message and parse it as consisting of two sets I_0 and I_1.
7. Sample seeds $s_0, s_1 \leftarrow \{0,1\}^{k(\lambda)}$ and send $s_0, h_{|I_0|}(s_0, x_0) \oplus m_0, s_1, h_{|I_1|}(s_1, x_1) \oplus m_1$, where x_0 is x restricted to the set of indices I_0 and x_1 is x restricted to the set of indices I_1.

This hybrid distribution is indistinguishable from the real distribution due to the extractability of $(\mathsf{EECommit}, \mathsf{EEDecommit})$ (Definition 3.3). Indeed, the values extracted from the simulator are used in place of the values opened by R^*. Conditioned on the opening phases succeeding, these values will be equal with all but negligible probability.

Now, the only difference between this hybrid experiment and the simulation is the value of the string m_{1-b}, where b is the bit extracted in the simulated game. Thus, to complete the proof, it suffices to argue that in the simulation, the string $h_{|I_{1-b}|}(s_{1-b}, x_{1-b})$, which is used to mask m_{1-b}, is statistically close to a uniformly random string. We will show this by using the algorithm R^* in the simulation to construct an algorithm B that interacts in the EPR protocol (Protocol 5). The output of the simulation will be identical to the output of B in the EPR protocol. B is constructed as follows.

- B forwards the state received from A to R^*.
- B runs the $\mathsf{EECommit}$ extractor as in Step 2 of the simulator, and responds to A with the extracted values $\{\widehat{\theta}_i, \widehat{x}_i\}_{i \in [16\lambda]}$.
- B receives the test subset T from A, and forwards it to R^*. Then, it executes the opening phases with R^*, and sends abort to A if any fail. Otherwise, it sends continue to A.
- In the post-processing phase, A acts as the simulator, and B as the receiver R^* in Steps 6-9.

The only difference between this EPR protocol and the simulation is that in the EPR protocol, R^* receives halves of EPR pairs in the first round, while in the simulation, R^* receives $|x\rangle_\theta$. However, from R^*'s perspective, these are both maximally mixed states.

Now, consider the two subsets $I_0, I_1 \subset [8\lambda]$ sent by R^* in the EPR protocol game. For $c \in \{0,1\}$, let $\mathsf{wt}(I_c)$ be the number of indices in I_c such that $\theta_i \neq \widehat{\theta}_i$. If $\mathsf{wt}(I_0) > 3\lambda/2$, then b will be fixed to 1, and by Lemma 3.8, $h_{|I_0|}(s_0, x_0)$ will be statistically close to a uniformly random string. Now, by Hoeffding's inequality, with overwhelming probability θ and $\widehat{\theta}$ differ in more than 3λ positions. Conditioned on this event, $\mathsf{wt}(I_0) \leq 3\lambda/2$ implies that $\mathsf{wt}(I_1) > 3\lambda/2$. If this case b will be fixed to 0, and again by Lemma 3.8, $h_{|I_1|}(s_1, x_1)$ will be statistically close to a uniformly random string. This completes the proof. □

Finally, the following corollary follows immediately from Corollary 5.3 and Theorem 6.1.

Corollary 6.4. *Quantum oblivious transfer (QOT) satisfying Definition 3.6 can be based on black-box use of statistically binding bit commitments, or on black-box use of quantum-hard one-way functions.*

7 Acknowledgement

This material is based on work supported in part by DARPA under Contract No. HR001120C0024 (for DK). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Government or DARPA.

References

[Aar09] Scott Aaronson, *Quantum copy-protection and quantum money*, 2009 24th Annual IEEE Conference on Computational Complexity (2009).

[ABG+20] Amit Agarwal, James Bartusek, Vipul Goyal, Dakshita Khurana, and Giulio Malavolta, *Post-quantum multi-party computation*, Cryptology ePrint Archive, Report 2020/1395, 2020, https://eprint.iacr.org/2020/1395.

[AC02] Mark Adcock and Richard Cleve, *A quantum goldreich-levin theorem with cryptographic applications*, Annual Symposium on Theoretical Aspects of Computer Science, Springer, 2002, pp. 323–334.

[BB84] Charles H Bennett and Gilles Brassard, *Proceedings of the ieee international conference on computers, systems and signal processing*, 1984.

[BBCS92] Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélène Skubiszewska, *Practical quantum oblivious transfer*, CRYPTO'91 (Joan Feigenbaum, ed.), LNCS, vol. 576, Springer, Heidelberg, August 1992, pp. 351–366.

[BCG+06] Michael Ben-Or, Claude Crépeau, Daniel Gottesman, Avinatan Hassidim, and Adam Smith, *Secure multiparty quantum computation with (only) a strict honest majority*, 47th FOCS, IEEE Computer Society Press, October 2006, pp. 249–260.

[BCJL93] Gilles Brassard, Claude Crépeau, Richard Jozsa, and Denis Langlois, *A quantum bit commitment scheme provably unbreakable by both parties*, 34th FOCS, IEEE Computer Society Press, November 1993, pp. 362–371.

[BF10] Niek J. Bouman and Serge Fehr, *Sampling in a quantum population, and applications*, CRYPTO 2010 (Tal Rabin, ed.), LNCS, vol. 6223, Springer, Heidelberg, August 2010, pp. 724–741.

[BM09] Boaz Barak and Mohammad Mahmoody-Ghidary, *Merkle puzzles are optimal - an $O(n^2)$-query attack on any key exchange from a random oracle*, CRYPTO 2009 (Shai Halevi, ed.), LNCS, vol. 5677, Springer, Heidelberg, August 2009, pp. 374–390.

[BS20] Nir Bitansky and Omri Shmueli, *Post-quantum zero knowledge in constant rounds*, Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020 (Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, eds.), ACM, 2020, pp. 269–279.

[CGS02] Claude Crépeau, Daniel Gottesman, and Adam Smith, *Secure multi-party quantum computation*, 34th ACM STOC, ACM Press, May 2002, pp. 643–652.

[Dac16] Dana Dachman-Soled, *Towards non-black-box separations of public key encryption and one way function*, Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part II, 2016, pp. 169–191.

[DFL+09] Ivan Damgård, Serge Fehr, Carolin Lunemann, Louis Salvail, and Christian Schaffner, *Improving the security of quantum protocols via commit-and-open*, CRYPTO 2009 (Shai Halevi, ed.), LNCS, vol. 5677, Springer, Heidelberg, August 2009, pp. 408–427.

[DGJ+20] Yfke Dulek, Alex B. Grilo, Stacey Jeffery, Christian Majenz, and Christian Schaffner, *Secure multi-party quantum computation with a dishonest majority*, EUROCRYPT 2020, Part III (Anne Canteaut and Yuval Ishai, eds.), LNCS, vol. 12107, Springer, Heidelberg, May 2020, pp. 729–758.

[DNS10] Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail, *Secure two-party quantum evaluation of unitaries against specious adversaries*, CRYPTO 2010 (Tal Rabin, ed.), LNCS, vol. 6223, Springer, Heidelberg, August 2010, pp. 685–706.

[DNS12] ———, *Actively secure two-party evaluation of any quantum operation*, CRYPTO 2012 (Reihaneh Safavi-Naini and Ran Canetti, eds.), LNCS, vol. 7417, Springer, Heidelberg, August 2012, pp. 794–811.

[FUYZ20] Junbin Fang, Dominique Unruh, Jun Yan, and Dehua Zhou, *How to base security on the perfect/statistical binding property of quantum bit commitment?*, Cryptology ePrint Archive, Report 2020/621, 2020, https://eprint.iacr.org/2020/621.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby, *A pseudorandom generator from any one-way function*, SIAM Journal on Computing **28** (1999), no. 4, 1364–1396.

[HSS11] Sean Hallgren, Adam Smith, and Fang Song, *Classical cryptographic protocols in a quantum world*, CRYPTO 2011 (Phillip Rogaway, ed.), LNCS, vol. 6841, Springer, Heidelberg, August 2011, pp. 411–428.

[IL89] Russell Impagliazzo and Michael Luby, *One-way functions are essential for complexity based cryptography (extended abstract)*, 30th FOCS, IEEE Computer Society Press, October / November 1989, pp. 230–235.

[ILL89] Russell Impagliazzo, Leonid A. Levin, and Michael Luby, *Pseudo-random generation from one-way functions (extended abstracts)*, 21st ACM STOC, ACM Press, May 1989, pp. 12–24.

[Imp95] Russell Impagliazzo, *A personal view of average-case complexity*, Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995, 1995, pp. 134–147.

[IPS08] Yuval Ishai, Manoj Prabhakaran, and Amit Sahai, *Founding cryptography on oblivious transfer - efficiently*, CRYPTO 2008 (David Wagner, ed.), LNCS, vol. 5157, Springer, Heidelberg, August 2008, pp. 572–591.

[IR90] Russell Impagliazzo and Steven Rudich, *Limits on the provable consequences of one-way permutations*, CRYPTO'88 (Shafi Goldwasser, ed.), LNCS, vol. 403, Springer, Heidelberg, August 1990, pp. 8–26.

[Kil88] Joe Kilian, *Founding cryptography on oblivious transfer*, 20th ACM STOC, ACM Press, May 1988, pp. 20–31.

[KRS09] Robert Konig, Renato Renner, and Christian Schaffner, *The operational meaning of min-and max-entropy*, IEEE Transactions on Information theory **55** (2009), no. 9, 4337–4347.

[LC97] Hoi-Kwong Lo and Hoi Fung Chau, *Is quantum bit commitment really possible?*, Physical Review Letters **78** (1997), no. 17, 3410.

[LR86] Michael Luby and Charles Rackoff, *Pseudo-random permutation generators and cryptographic composition*, 18th ACM STOC, ACM Press, May 1986, pp. 356–363.

[May96] Dominic Mayers, *Quantum key distribution and string oblivious transfer in noisy channels*, CRYPTO'96 (Neal Koblitz, ed.), LNCS, vol. 1109, Springer, Heidelberg, August 1996, pp. 343–357.

[May97] Dominic Mayers, *Unconditionally secure quantum bit commitment is impossible*, Physical review letters **78** (1997), no. 17, 3414.

[MMP14] Mohammad Mahmoody, Hemanta K. Maji, and Manoj Prabhakaran, *On the power of public-key encryption in secure computation*, TCC 2014 (Yehuda Lindell, ed.), LNCS, vol. 8349, Springer, Heidelberg, February 2014, pp. 240–264.

[MS94] Dominic Mayers and Louis Salvail, *Quantum oblivious transfer is secure against all individual measurements*, Proceedings Workshop on Physics and Computation. PhysComp'94, IEEE, 1994, pp. 69–77.

[Nao91] Moni Naor, *Bit commitment using pseudorandomness*, Journal of Cryptology **4** (1991), no. 2, 151–158.

[Ren08] Renato Renner, *Security of quantum key distribution*, International Journal of Quantum Information **6** (2008), no. 01, 1–127.

[RK05] Renato Renner and Robert König, *Universally composable privacy amplification against quantum adversaries*, TCC 2005 (Joe Kilian, ed.), LNCS, vol. 3378, Springer, Heidelberg, February 2005, pp. 407–425.

[RW05] Renato Renner and Stefan Wolf, *Simple and tight bounds for information reconciliation and privacy amplification*, ASIACRYPT 2005 (Bimal K. Roy, ed.), LNCS, vol. 3788, Springer, Heidelberg, December 2005, pp. 199–216.

[Unr07] Dominique Unruh, *Random oracles and auxiliary input*, CRYPTO 2007 (Alfred Menezes, ed.), LNCS, vol. 4622, Springer, Heidelberg, August 2007, pp. 205–223.

[Wat06] John Watrous, *Zero-knowledge against quantum attacks*, 38th ACM STOC (Jon M. Kleinberg, ed.), ACM Press, May 2006, pp. 296–305.

[Yao95] Andrew Chi-Chih Yao, *Security of quantum protocols against coherent measurements*, 27th ACM STOC, ACM Press, May / June 1995, pp. 67–75.

[Zha12] Mark Zhandry, *How to construct quantum random functions*, 53rd FOCS, IEEE Computer Society Press, October 2012, pp. 679–687.

A A Quantum Sampling Lemma

A.1 The [BF10] EPR Protocol

We describe a protocol we call the "EPR Protocol," which is essentially equivalent to the "QOT*" protocol described by [BF10]. This protocol is exclusively stated as a tool for arguing about the security of BB84-style protocols, and should not be viewed as a functional protocol.

In the EPR protocol, party A deviates from the standard BB84-style template, instead generating EPR pairs and sending one half of each pair. From the other party's point of view, this message is distributed identically to random BB84 states, since in both cases each qubit is maximally mixed. Note that in a standard BB84-style protocol, A samples a random bit string x and encodes it in a random bases before interacting with B. But in the EPR protocol, A obtains x as the result of measuring her halves of the EPR pairs; this has the advantage of significantly simplifying arguments about the entropy of x from B's point of view [BF10].

This is captured in Lemma A.1, which relies heavily on quantum sampling techniques of [BF10].

Lemma A.1. *Consider the protocol in Fig. 5. Let $n \geq m/2$. For any subset of $I \subset [n]$ of size t, let $\mathsf{wt}(I)$ be the number of positions in I that θ and $\widehat{\theta}$ differ, and let x_I be the string x restricted to positions in I. Then for any $\epsilon > 0$, x_I has quantum min-entropy (see definition in Section 3.4) at least $\mathsf{wt}(I) - \epsilon t$ conditioned on B's view and on $x_{\overline{I}}$, except with probability* $\mathrm{negl}(kt^2/n^2)$.

Protocol 5

Parameters: $m = n + k$.

1. A prepares m EPR pairs and sends one of each pair to B. A also samples $\theta \leftarrow \{+, \times\}^m$.
2. B responds with $\widehat{\theta} \in \{+, \times\}^m, \widehat{x} \in \{0, 1\}^m$.
3. A sends a random test subset $T \subset [m]$ of size k.
4. B responds with either continue or abort.
5. A measures each of her qubits in T in the corresponding basis of $\widehat{\theta}$. For any index i such that $\widehat{\theta}_i = \theta_i$, A checks that the measured x_i matches $\widehat{x}_i$. If not, A aborts. If all checks pass, A restricts her qubits and θ to subset $\overline{T}$ of size n. Then she measures her qubits in basis θ to produce a string x, and sends θ to B.
6. *Post-processing*: A and B engage in some protocol using the strings x and θ, and $\widehat{x}, \widehat{\theta}$ restricted to positions in $\overline{T}$.

Figure 5: EPR Protocol.

Proof. (Sketch) We argue for the case where the strings $\widehat{\theta}, \widehat{x}$ returned by B are both the all-zeros strings; all other possibilities can be handled by applying the same arguments after an appropriate change of bases. Now consider the (purified) joint state $|\psi_{\mathsf{A},\mathsf{B}}\rangle$ of A and B in the middle of step 5, after A has restricted her state to n qubits. The check at the beginning of step 5 is essentially establishing that A's part of $|\psi_{\mathsf{A},\mathsf{B}}\rangle$ is close to the all zeros state $|0\rangle^{\otimes n}$. Indeed, applying [BF10, Theorem 3] (plus analysis in [BF10, Appendix B.4] bounding the error probability of the corresponding classical sampling strategy) with some $\delta > 0$ shows that, conditioned on A not aborting after this check, A's part of the state is within trace distance $\mathrm{negl}(k\delta^2)$ of a superposition of states with Hamming weight at most δn. That is, we can assume that the state $|\psi_{\mathsf{A},\mathsf{B}}\rangle$ is a superposition of vectors that only contain at most δn 1s among the n positions in A. Now, letting $\delta = \epsilon' t/n$ for a small enough constant $\epsilon' > 0$ and applying [BF10, Corollary 1] shows that the conditional min-entropy (conditioned on R's portion of the state) of x_I is at least $\mathsf{wt}(I) - \epsilon t$.

□