Post-Quantum Simulatable Extraction with Minimal Assumptions: Black-Box and Constant-Round

Nai-Hui Chia[1], Kai-Min Chung[2], Xiao Liang$^{[0000-0003-0858-9289]3}$, and Takashi Yamakawa[4]

[1] Rice University, Houston, USA
nc67@rice.edu

[2] Academia Sinica, Taipei, Taiwan
kmchung@iis.sinica.edu.tw

[3] Stony Brook University, Stony Brook, USA
xiao.crypto@gmail.com

[4] NTT Social Informatics Laboratories, Tokyo, Japan
takashi.yamakawa.ga@hco.ntt.co.jp

Abstract. From the minimal assumption of post-quantum semi-honest oblivious transfers, we build the first *ε-simulatable* two-party computation (2PC) against quantum polynomial-time (QPT) adversaries that is both constant-round and black-box (for both the construction and security reduction). A recent work by Chia, Chung, Liu, and Yamakawa (FOCS'21) shows that post-quantum 2PC with standard simulation-based security is impossible in constant rounds, unless either **NP** $\subseteq$ **BQP** or relying on non-black-box simulation. The ε-simulatability we target is a relaxation of the standard simulation-based security that allows for an arbitrarily small noticeable simulation error ε. Moreover, when quantum communication is allowed, we can further weaken the assumption to post-quantum secure one-way functions (PQ-OWFs), while maintaining the constant-round and black-box property.

Our techniques also yield the following set of *constant-round and black-box* two-party protocols secure against QPT adversaries, only assuming black-box access to PQ-OWFs:

- extractable commitments for which the extractor is also an ε-simulator;
- ε-zero-knowledge commit-and-prove whose commit stage is extractable with ε-simulation;
- ε-simulatable coin-flipping;
- ε-zero-knowledge arguments of knowledge for **NP** for which the knowledge extractor is also an ε-simulator;
- ε-zero-knowledge arguments for **QMA**.

At the heart of the above results is a black-box extraction lemma showing how to efficiently extract secrets from QPT adversaries while disturbing their quantum states in a controllable manner, i.e., achieving ε-simulatability of the after-extraction state of the adversary.

1 Introduction

Extractability is an important concept in cryptography. A typical example is extractable commitments, which enable an extractor to extract a committed message from a malicious committer. Extractable commitments have played a central role in several major cryptographic tasks, including (but not limited to) secure two-party and multi-party computation (e.g., [19, 61, 36, 38]), zero-knowledge (ZK) protocols (e.g., [63, 56]), concurrent zero-knowledge protocols (e.g., [62, 59]), non-malleable commitments (e.g., [37, 52]) etc. Recently, two concurrent works by Grilo, Lin, Song, and Vaikuntanathan [40] and Bartusek, Coladangelo, Khurana, and Ma [5] (based on earlier works [22, 7, 23, 11]) demonstrate new applications of extractable commitments in quantum cryptography. They show that quantumly secure extractable commitments are sufficient for constructing maliciously secure quantum oblivious transfers (OTs), which can be compiled into general-purpose quantum MPC [49, 26].[5]

As noted in [40], it is surprisingly non-trivial to construct quantumly secure extractable commitments. The reason is that quantum extractability requires an extractor to extract the committed message *while simulating the committer's post-execution state.* However, known rewinding-based classical extraction techniques are not directly applicable as it is unclear if they could provide any simulation guarantee when used against quantum adversaries. To address this issue, recent works [40, 5] propose new polynomial-round *quantum* constructions of quantumly secure extractable commitments from post-quantum one-way functions (PQ-OWFs), which are functions efficiently computable in the classical sense but one-way against quantum polynomial-time (QPT) adversaries. Relying on assumptions stronger than PQ-OWFs, *classical* constructions of quantumly secure extractable commitments (which we call *post-quantum* extractable commitments) are known [10, 4, 9, 43, 58]. However, those constructions require (at least) the existence of OTs.

Moreover, all existing post-quantum extractable commitments make *non-black-box* use of their building-block primitives. This is not ideal as *black-box* constructions are often preferred over non-black-box ones. A black-box construction only depends on the input/output behavior of its building-block cryptographic primitive(s). In particular, such a construction is independent of the specific implementation or code of the building-block primitive. Black-box constructions enjoy certain advantages. For example, they remain valid even if the building-block primitive/oracle is based on a *physical* object such as a noisy channel or tamper-proof hardware [66, 21, 34]. Also, since the efficiency of black-box constructions does not depend on the implementation details of the primitive, their efficiency can be theoretically independent of the code of lower-level primitives. Indeed, it has been an important theme to obtain black-box constructions for major cryptographic objects, e.g., [51, 24, 46, 47, 41, 49, 61, 36, 37, 55, 52, 39, 31, 44, 32, 50, 15, 33, 29, 54, 18].

[5] They actually rely on extractable and *equivocal* commitments. However, since equivocality can be added easily, extractable commitments are the essential building block.

In the classical setting, it is well-known that constant-round extractable commitments can be obtained assuming only black-box access to OWFs [61, 25, 62, 63].[6] Therefore, it is natural to ask the following analog question in the quantum setting: Is it possible to construct constant-round post-quantum extractable commitments assuming only black-box access to PQ-OWFs? We remark that this question is open even if we do not require the scheme to be constant-round or black-box.

The Black-Box Extraction Barrier. We observe that the recent lower bound on black-box post-quantum ZK [17] suggests a negative answer to the above question. Namely, if we have constant-round post-quantum extractable commitment with black-box extraction, then we can construct constant-round post-quantum ZK arguments for **NP** with black-box simulation based on standard techniques (see [16, Appendix A] for details). However, [17] showed that such a ZK argument cannot exist unless $\mathbf{NP} \subseteq \mathbf{BQP}$, which seems unlikely.[7]

ε-Simulation Security. On the other hand, another recent work [18] showed that we can bypass the impossibility result by relaxing the requirement of ZK to the so-called ε-ZK [27, 8, 28]. The standard ZK property requires a simulator to simulate the verifier's view in a way that no distinguisher can distinguish it from the real one with non-negligible advantage. In contrast, the ε-ZK property only requires the existence of a simulator such that for any noticeable $\varepsilon(\lambda)$, the simulated view can be distinguished from the real one with advantage at most ε. As explained in [18], ε-ZK is still useful in several applications of ZK. The results in [18] suggest the possibility of post-quantum extractable commitments if we relax the simulation requirement on the extractor to a similar ε-close[8] version. We will refer to this weakened notion as *extractability with ε-simulation*.[9] It seems natural to hope that the techniques in [18] could be used in the context of extractable commitments. Indeed, by plugging the ZK argument from [18] into the OT-based construction [10, 9, 43, 58], we can obtain a non-black-box construction of constant-round post-quantum extractable commitments with ε-simulation, assuming constant-round post-quantum OTs. However, if we focus on *black-box constructions* from the *minimal assumption* of PQ-OWFs, it is unclear if the techniques in [18] would help. Therefore, we ask the following question:

> **Question 1:** *Is it possible to have constant-round post-quantum extractable commitments with ε-simulation, assuming only black-box access to PQ-OWFs?*

[6] The term "black-box" here refers to both black-box constructions and black-box extraction.

[7] A concurrent work by Lombardi, Ma, and Spooner [57] showed that the impossibility of [17] can be avoided if we consider a stronger computational model for simulators. We provide more discussion in [16, Section 1.3].

[8] Throughout this paper, "ε-close" means that the adversary's distinguishing advantage is at most ε.

[9] In the main body, we call it *strong* extractability with ε-simulation since we also define a weaker variant of that.

Table 1. Comparison of Quantumly Secure Extractable Commitment.

Reference	#Round	Cla. Const.	BB Const.	BB Ext.	Siml. Err.	Assumption
[40]	$\mathsf{poly}(\lambda)$			✓	negl	OWF
[5]	$\mathsf{poly}(\lambda)$		✓	✓	negl	OWF
[10]	$O(1)$	✓			negl	QFHE+QLWE
folklore[a]	$\mathsf{poly}(\lambda)$	✓		✓	negl	OT
folklore+[18]	$O(1)$	✓		✓	ε	$O(1)$-round OT
Ours	$O(1)$	✓	✓	✓	ε	OWF

The "Cla. Const.", "BB Const.", and "BB Ext." columns indicate if the scheme relies on classical constructions, black-box constructions, and extraction, respectively. In the "Siml. Err." column, negl and ε mean that the construction achieves the standard quantum extractability and quantum extractability with ε-simulation, respectively. In "Assumption" column, QFHE and QLWE means quantum fully homomorphic encryption and the quantum hardness of learning with errors, respectively.

[a] As noted in [9], the construction is implicit in [10, 43, 58].

In the more general context of 2PC and MPC, the implication of [17] is that to obtain constant-round constructions with post-quantum security, we have to

1. rely on non-black-box simulation, *or*
2. aim for a relaxed security notion (e.g., ε-close simulation security).

The first approach was taken in [2] (based on [10]), leading to a constant-round post-quantum MPC protocol with non-black-box simulation. On the other hand, the second approach has not been explored in the existing literature of post-qauntum 2PC or MPC (except for the special case of ZK as in [18]). It is possible to construct constant-round post-quantum 2PC with ε-close simulation by combining constant-round post-quantum semi-honest OTs and the constant-round post-quantum ε-ZK in [18]. However, the naive approach will lead to a non-black-box construction. In contrast, in the classical setting, constant-round *black-box constructions* of 2PC [61] and MPC [19, 36] are known from the minimal assumption of constant-round semi-honest OT. The above discussion suggests that one has to relax the security requirement when considering the post-quantum counterparts of these tasks. We will refer to 2PC and MPC with ε-close simulation as ε-2PC and ε-MPC respectively. Then, an interesting question is:

> **Question 2:** *Do there exist constant-round black-box post-quantum ε-2PC and ε-MPC, assuming only constant-round semi-honest OTs secure against QPT adversaries?*

1.1 Our Results

We answer **Question 1** affirmatively and address **Question 2** partially, showing a positive answer only for the two-party case. We first construct constant-round black-box post-quantum extractable commitments with ε-simulation from PQ-OWFs. See Table 1 for comparisons among quantumly secure extractable commitments. Such commitments imply new constant-round and black-box protocols for general-purpose 2PC secure against QPT adversaries. In particular, we get

- post-quantum ε-2PC from semi-honest OTs, and
- post-quantum ε-2PC from PQ-OWFs, assuming that quantum communication is possible. (Henceforth, we will use OWFs to denote PQ-OWFs.)

As an intermediate tool to achieve the above results, we construct a constant-round post-quantum ε-ZK commit-and-prove, assuming only black-box access to OWFs. Black-box zero-knowledge commit-and-prove [47, 37, 39, 45, 50, 53] is a well-studied primitive in classical cryptography; it enables a prover to commit to some message and later to prove in zero-knowledge that the committed message satisfies a given predicate in a *black-box* manner. In addition to being secure in the post-quantum setting, our construction enjoys the extra property that the commit stage is extractable (albeit with only ε-simulation of the adversary's post-extraction state). Such a constant-round ε-simulatable ExtCom-and-Prove protocol implies the following set of two-party protocols:

- constant-round black-box post-quantum coin-flipping with ε-simulation,
- constant-round black-box post-quantum ε-ZK arguments of knowledge for **NP** with ε-simulating knowledge extractor, and
- constant-round black-box ε-ZK arguments for **QMA**.

In the following, we provide more discussion about them.

Coin-Flipping. Coin-flipping is a two-party protocol to generate a uniformly random string that cannot be biased by either of parties (w.r.t. the standard simulation-based security). In the classical setting, constant-round black-box constructions from OWFs are known [61]. On the other hand, known post-quantum constructions are based on stronger assumptions (like QLWE) than OWFs, and require either polynomial rounds [58] or non-black-box simulation [2]. Our construction can be understood as the post-quantum counterpart of the classical construction by Pass and Wee [61], albeit with ε-simulation.

Arguments of Knowledge with Simulating Extractor. Arguments of knowledge intuitively require an extractor to extract a witness from any efficient malicious prover whenever it passes the verification. In the classical setting, constant-round black-box constructions from OWFs are known [61]. In the post-quantum setting, there are two existing notions of arguments of knowledge depending on whether we require the extractor to simulate the prover's post-execution state or not. For the "without-simulation" version, Unruh [64] gave a polynomial-round black-box construction from OWFs.[10] For the "with-simulation" version, all existing constructions require both polynomial rounds *and* assumptions stronger than OWFs (like QLWE) [43, 58, 3].[11] Our construction improves both the round

[10] Though Unruh originally assumes *injective* OWFs, [18] pointed out that any OWF suffices.

[11] Though not claimed explicitly, it seems also possible to obtain constant-round construction with non-black-box simulation from QLWE and QFHE based on [10].

complexity and the required assumption, at the cost of weakening ZK and extractability to their ε-simulation variants. On the other hand, we note that the construction in [3] achieves *proofs of knowledge*, while ours only achieves *arguments of knowledge*. We also note that even without knowledge extractability, our construction improves the construction in [18, Section 6], which is a *non-black-box construction* of constant-round ε-ZK arguments for **NP** from OWFs.

ZK Arguments for QMA. QMA is a quantum analog of **NP**. Known constructions of ZK proofs or arguments for **QMA** rely on either polynomial-round communication [14, 13, 12] or non-black-box simulation [10]. If we relax the ZK requirement to ε-ZK, constant-round black-box ε-ZK *proofs* were already constructed in [18]; but that construction needs to assume collapsing hash functions, which are stronger than OWFs. Our construction improves the assumption to the existence of OWFs at the cost of weakening the soundness to the computational one (i.e., an argument system).

Discussion. Due to space constraints, we provide additional discussion on minimality of assumptions, other potential applications, and a comparison with the concurrent work by Lombardi, Ma, and Spooner [57] in the full version [16, Sections 1.2 and 1.3].

2 Technical Overview

2.1 Extractable Commitment with ε-Simulation

Our main technical tool for constructing ε-simulatable extractable commitments is a generalization of the extract-and-simulate technique from [18].

Extract-and-Simulation Lemma in [18]. We briefly recall the extract-and-simulate lemma shown in [18, Lemma 4.2].[12] At a high level, that lemma can be interpreted as follows.[13] Let $\mathcal{A}$ be a quantum algorithm with an initial state ρ. Suppose that $\mathcal{A}$ outputs some *unique* classical string s^* or otherwise outputs a failure symbol Fail. Then, there exists a simulation-extractor $\mathcal{SE}$ such that for any noticeable function ε (on the security parameter), the following two experiments are ε-close:

```
$\underline{\mathsf{Exp}_{\mathsf{real}}}$

Run $\mathcal{A}(\rho)$,
If $\mathcal{A}$ outputs $\mathsf{Fail}$,
    Output $\mathsf{Fail}$
Else output $\mathcal{A}$'s final state.
```

```
$\underline{\mathsf{Exp}_{\mathsf{ext}}}$
$(s_{\mathsf{Ext}}, \rho_{\mathsf{Ext}}) \leftarrow \mathcal{SE}^{\mathcal{A}(\rho)}(1^{\varepsilon^{-1}})$
Run $\mathcal{A}(\rho_{\mathsf{Ext}})$,
If $\mathcal{A}$ outputs $\mathsf{Fail} \ \vee s_{\mathsf{Ext}} \neq s^*$,
    Output $\mathsf{Fail}$
Else output $\mathcal{A}$'s final state.
```

[12] In [18], the lemma was called "extraction lemma". Here, we add "simulation" to emphasize that the extractor not only extracts but also simulates the adversary's state.

[13] There are two versions of their lemma: the statistically-binding case and the strong collapse-binding case. The abstraction given here is a generalization of the statistically-binding case.

Generalizing the Lemma. Note that their lemma will enable us to extract s^* from $\mathcal{A}$ *only if* $\mathcal{A}$ *reveals the value* s^* *at the end.* As shown in [18], this already suffices for the constant-round ZK proof by Goldreich and Kahan [35], where the verifier first commits to the challenge and opens it (i.e., "reveals it at the end") later. However, this does not seem to help obtain extractable commitments, because the committed message is not revealed *at the end the commit stage* (i.e., before decommitment happens); but the definition of extractable commitments does require extraction before decommitment happens.

To deal with this issue, we generalize the [18] lemma as follows. Let $\mathcal{A}$ be a quantum algorithm that on an initial state ρ, outputs a classical symbol Succ or Fail. Moreover, suppose that there are a unique classical string s^* and a "simulation-less extractor" $\mathsf{Ext}_{\mathsf{Sim\text{-}less}}$ that outputs s^* or otherwise Fail. Also, suppose that

$$\Pr\left[\mathsf{Ext}_{\mathsf{Sim\text{-}less}}^{\mathcal{A}(\rho)} = s^*\right] \geq \left(\Pr[\mathcal{A}(\rho) = \mathsf{Succ}]\right)^c - \mathsf{negl}(\lambda) \tag{1}$$

for some constant c. Our generalized lemma says that the ε-closeness between $\mathsf{Exp}_{\mathsf{real}}$ and $\mathsf{Exp}_{\mathsf{ext}}$ holds in this setting as well.

One can think of $\mathcal{A}$ as a joint execution of a malicious committer and honest receiver where it outputs Succ if and only if the receiver accepts. In this setting, one can understand the above lemma as a lifting lemma from "simulation-less extractor" to "ε-simulation extractor" in the setting where the extracted string is unique. In the main body, we present the lemma in a more specific form (Lem. 1), where it is integrated with Watrous' rewinding lemma [65] and Unruh's rewinding lemma [64], because that is more convenient for our purpose. We will overview the intuition behind the above generalized lemma toward the end of this subsection.

Weakly Extractable Commitment. Next, we explain how to construct post-quantum extractable commitments using our extract-and-simulate lemma. We go through the following two steps:

1. Construct a commitment scheme $\mathsf{wExtCom}$ that satisfies a weak version of post-quantum extractability with ε-simulation.
2. Upgrade $\mathsf{wExtCom}$ into a scheme ExtCom with full-fledged post-quantum extractability with ε-simulation (which we call *strong* extractability with ε-simulation to distinguish it from the weak one).

We first explain Step 1, the construction of $\mathsf{wExtCom}$. Actually, our construction of $\mathsf{wExtCom}$ is exactly the same as the classical extractable commitments from OWFs given in [61], which are in turn based on earlier works [25, 62, 63]. Let Com be a computationally-hiding and statistically-binding commitment scheme (say, Naor's commitment [60]). Then, the commitment scheme $\mathsf{wExtCom}$ works as follows.

Commit Stage:

1. To commit to a message m, the committer C generates $k = \omega(\log \lambda)$ pairs of 2-out-of-2 additive secret shares $\{(v_i^0, v_i^1)\}_{i=1}^k$, i.e., they are uniformly

chosen conditioned on that $v_i^0 \oplus v_i^1 = m$ for each $i \in [k]$. Then, C commits independently to each v_i^b ($b \in \{0,1\}$) in parallel by using Com. We denote these commitments by $\{(\mathsf{com}_i^0, \mathsf{com}_i^1)\}_{i=1}^k$.

2. R randomly chooses $\mathbf{c} = (c_1, ..., c_k) \leftarrow \{0,1\}^k$ and sends it to C.
3. C decommits $\{\mathsf{com}_i^{c_i}\}_{i=1}^k$ to $\{v_i^{c_i}\}_{i=1}^k$, and R checks that the openings are valid.

Decommit Stage:

1. C sends m and opens all the remaining commitments; R checks that all openings are valid *and* $v_i^0 \oplus v_i^1 = m$ for all $i \in [k]$.

Suppose that a malicious committer C^* generates commitments $\{(\mathsf{com}_i^0, \mathsf{com}_i^1)\}_{i=1}^k$ in Step 1, and let ρ be its internal state at this point. Then, we consider $\mathcal{A}(\rho)$ that works as follows:

- Choose $\mathbf{c} = (c_1, ..., c_k) \leftarrow \{0,1\}^k$ at random.
- Send $\mathbf{c}$ to C^* and simulate Step 3 of C^* in the commit stage to get $\{v_i^{c_i}\}_{i=1}^k$ and the corresponding decommitment information.
- If all the openings are valid, output Succ; otherwise output Fail.

To use our extract-and-simulate lemma, we need to construct a simulation-less extractor $\mathsf{Ext}_{\mathsf{Sim\text{-}less}}$ satisfying Inequality (1). A natural idea is to use Unruh's rewinding lemma [64]. His lemma directly implies that if $\mathcal{A}$ returns Succ with probability δ, then we can obtain valid $\{v_i^{c_i}\}_{i=1}^k$ and $\{v_i^{c_i'}\}_{i=1}^k$ for two uniformly random challenges, $\mathbf{c} = (c_1, \ldots, c_k)$ and $\mathbf{c}' = (c_1', \ldots, c_k')$, with probability at least δ^3. In that case, unless $\mathbf{c} = \mathbf{c}'$ (which happens with negligible probability), we can "extract" $m = v_i^0 \oplus v_i^1$ from position $i \in [k]$ that satisfies $c_i \neq c_i'$. However, such an "extractor" does not satisfy the assumption for our generalized extract-and-simulate lemma in general, because $v_i^0 \oplus v_i^1$ may be different for each $i \in [k]$.

Therefore, to satisfy this requirement, we have to introduce an additional assumption that $\{(\mathsf{com}_i^0, \mathsf{com}_i^1)\}_{i=1}^k$ is *consistent*, i.e., if we denote the corresponding committed messages as $\{(v_i^0, v_i^1)\}_{i=1}^k$, then there exists a unique m such that $v_i^0 \oplus v_i^1 = m$ for all $i \in [k]$.[14] With this assumption, we can apply our generalized extract-and-simulate lemma. It enables us to extract the committed message *and* simultaneously ε-simulate C^*'s state, conditioned on that the receiver accepts in the commit stage. The case where the receiver rejects can be easily handled using Watrous' rewinding lemma [65] as we will explain later. As a result, we get an ε-simulating extractor that works well conditioned on that the commitments generated in Step 1 are consistent. We will refer to such a weak notion of simulation-extractability as *weak extractability with ε-simulation* (see Def. 7 for the formal definition).

Moreover, since Unruh's rewinding lemma naturally gives a simulation-less extractor in the parallel setting (where C^* interacts with many copies of R in parallel), we can prove the parallel version of the weak extractability with ε-simulation similarly. More generally, we prove that $\mathsf{wExtCom}$ satisfies a further

[14] The corresponding message is well-defined (except for negligible probability) since we assume that Com is statistically binding.

generalized notion of extractability which we call the *special parallel weak extractability with ε-simulation* (see Def. 10 for the formal definition). Roughly speaking, it requires an ε-simulating extractor to work in n-parallel execution as long as the commitments in some subset of $[n]$ are consistent and the committed messages in those sessions determine a unique value. We remark that this parallel extractability will play an important role in the weak-to-strong compiler which we discuss next.

Weak-to-Strong Compiler. The reason why we cannot directly prove that $\mathsf{wExtCom}$ satisfies the strong extractability with ε-simulation is related to an issue that is often referred to as *over-extraction* in the classical literature (e.g., [37, 30, 52]). Over-extraction means that an extractor may extract some non-$\bot$ message from an invalid commitment, instead of detecting the invalidness of the commitment. In particular, there does not exist a unique "committed message" when the commitment is ill-formed in $\mathsf{wExtCom}$, and extraction of such a non-unique message may collapse the committer's state. To deal with this issue, we have to add some mechanism which could help the receiver (and thus the extractor) detect (in)validness of the commitment.

One possible approach is to revisit the techniques developed in the classical setting, performing necessary surgery to make the proof work against QPT adversaries. However, as demonstrated by the above cited works, existing techniques in the classical setting are already delicate. Even if it would work eventually, such a non-black-box treatment would further complicate the proof undesirably. Therefore, we present an alternative approach that deviates from existing ones in the classical setting. As we will show later, this new approach turns out to be quantum-friendly.

Roughly speaking, our construction ExtCom works as follows:

Commit Stage:

1. The committer C generates shares $\{\mathsf{v}_i\}_{i=1}^n$ of a *verifiable secret sharing* (VSS) scheme of the message to be committed to, and then commits to each v_i using $\mathsf{wExtCom}$ separately in parallel.
2. C and the receiver R execute a "one-side simulatable" coin-flipping protocol based on $\mathsf{wExtCom}$ to generate a random subset T of $[n]$ of a certain size.[15] Specifically, they do the following:
   (a) R commits to a random string r_1 by $\mathsf{wExtCom}$.
   (b) C sends a random string r_2 in the clear.
   (c) R opens r_1. Then, both parties derive the subset T from $r_1 \oplus r_2$.
3. C opens the commitments corresponding to the subset T, and R checks their validness and consistency.

[15] We remark that it is a non-trivial task to construct constant-round two-party coin-flipping from OWFs in the quantum setting, achieving the (even ε-)simulation-based security *against both parties.* Indeed, that will be one application of the strongly extractable commitment with ε-simulation, which we are now constructing. However, this is not a circular reasoning. Here, we need simulation-based security only against a malicious receiver. For such a one-side simulatable coin-flipping, the weakly extractable commitment $\mathsf{wExtCom}$ (with ε-simulation) suffices.

Decommit Stage:

1. C opens all the commitments. R checks those openings are valid. If they are valid, R runs the reconstruction algorithm of VSS to recover the committed message.

Using a similar argument as that for the soundness of the *MPC-in-the-head paradigm* [48, 36],we can show that if a malicious committer passes the verification in the commit stage, then:

1. Most of the commitments of $\mathsf{wExtCom}$ generated in Step 1 are valid *as a commitment*; **and**
2. The committed shares in those valid commitments determines a *unique* message that can be recovered by the reconstruction algorithm of VSS.

Then, we can apply the special parallel weak extractability with ε-simulation of $\mathsf{wExtCom}$ to show the strong extractability with ε-simulation of ExtCom. We remark that essentially the same proof can be used to show that the *parallel* execution of ExtCom is still *strongly* extractable (with ε-simulation). We refer to this as the parallel-strong extractability with ε-simulation. It will play a critical role in our construction of ExtCom-and-Prove (see Sec. 2.2).

Dealing with Rejection in Commit Stage. So far, we have only focused on the case where the receiver accepts in the commit stage. However, the definition of (both weak and strong) extractability requires that the final state should be simulated even in the case where the receiver rejects in the commit stage. In this case, of course, the extractor does not need to extract anything, and thus the simulation is straightforward. A non-trivial issue, however, is that the extractor does not know if the receiver rejects in advance. This issue can be solved by a technique introduced in [10]. The idea is to just guess if the receiver accepts, and runs the corresponding extractor assuming that the guess is correct. This gives an intermediate extractor that succeeds with probability almost $1/2$ and its output correctly simulates the desired distribution conditioned on that it does not abort. Such an extractor can be compiled into a full-fledged extractor that does not abort by Watrous' rewinding lemma [65].

Proof Idea for the Generalized Extract-and-Simulate Lemma. Finally, we briefly explain the idea for the proof of our generalized extract-and-simulate lemma. The basic idea is similar to the original extract-and-simulate lemma in [18]—Use Jordan's lemma to decompose the adversary's internal state into "good" and "bad" subspaces, and amplify the extraction probability in the good subspace while effectively ignoring the bad-subspace components. However, the crucial difference is that in [18], they define those subspaces with respect to the success probability of $\mathcal{A}$ whereas we define them with respect to the success probability of $\mathsf{Ext}_{\mathsf{Sim\text{-}less}}$. That is, for a noticeable δ, we apply Jordan's lemma to define a subspaces $S_{<\delta}$ and $S_{\geq\delta}$ such that

1. When $\mathsf{Ext}_{\mathsf{Sim\text{-}less}}$'s input is in $S_{<\delta}$ (resp. $S_{\geq\delta}$), it succeeds in extracting s^* with probability $< \delta$ (resp. $\geq \delta$).

2. Given a state in $S_{\geq\delta}$, we can extract s^* with overwhelming probability within $O(\delta^{-1})$ steps.
3. The above procedure does not cause any interference between $S_{<\delta}$ and $S_{\geq\delta}$.

We define $\mathcal{SE}$ to be an algorithm that runs the procedure in Item 2 and outputs s (which is supposed to be s^* in the case of success) and the post-execution state of $\mathcal{A}$. First, we consider simpler cases where the initial state of the experiments is a pure state $|\psi\rangle$ that is in either $S_{\geq\delta}$ or $S_{<\delta}$.

Case of $|\psi\rangle \in S_{\geq\delta}$: In this case, Item 2 implies that $\mathcal{SE}$ outputs s^* with overwhelming probability. In general, such an almost-deterministic quantum procedure can be done (almost) without affecting the state (e.g., see the *Almost-as-Good-as-New Lemma* in [1, Lemma 2.2]). Therefore, $\mathsf{Exp}_{\mathsf{real}}$ and $\mathsf{Exp}_{\mathsf{ext}}$ are negligibly indistinguishable in this case.

Case of $|\psi\rangle \in S_{<\delta}$: For any state $|\psi_{<\delta}\rangle \in S_{<\delta}$, Item 1 implies

$$\Pr\left[\mathsf{Ext}_{\mathsf{Sim\text{-}less}}^{\mathcal{A}(|\psi_{<\delta}\rangle)} = s^*\right] \leq \delta.$$

On the other hand, our assumption (i.e., Inequality (1)) implies

$$\Pr\left[\mathsf{Ext}_{\mathsf{Sim\text{-}less}}^{\mathcal{A}(|\psi_{<\delta}\rangle)} = s^*\right] \geq \left(\Pr[\mathcal{A}(|\psi_{<\delta}\rangle) = \mathsf{Succ}]\right)^c - \mathsf{negl}(\lambda)$$

for some constant c. By combining them, we have

$$\Pr[\mathcal{A}(|\psi_{<\delta}\rangle) = \mathsf{Succ}] \leq (\delta + \mathsf{negl}(\lambda))^{1/c}.$$

We note that the second output of $\mathcal{SE}$ in $\mathsf{Exp}_{\mathsf{ext}}$ is in $S_{<\delta}$ if the initial state is in $S_{<\delta}$ by Item 3. Therefore, if we run $\mathsf{Exp}_{\mathsf{real}}$ or $\mathsf{Exp}_{\mathsf{ext}}$ with an initial state in $S_{<\delta}$, it outputs Fail with probability $> 1 - (\delta + \mathsf{negl}(\lambda))^{1/c}$. Recall that when an experiment outputs Fail, no information about the internal state of $\mathcal{A}$ is revealed. Thus, the distinguishing advantage between those experiments can be bounded by $O(\delta^{1/c})$.

In general, the initial state is a superposition of $S_{<\delta}$ component and $S_{\geq\delta}$ component. Thanks to Item 3, we can reduce the general case to the above two cases. When doing that, there occurs an additional loss of the 4-th power of δ due to a technical reason. Still, we can bound the distinguishing advantage between the two experiments by $O(\delta^{1/(4c)})$. This can be made to be an arbitrarily small noticeable function because δ is an arbitrarily small noticeable function. This suffices for establishing the ε-closeness of those experiments.

2.2 Black-Box ε-Simulatable ExtCom-and-Prove

Black-box zero-knowledge commit-and-prove allows a committer to commit to some message m (the Commit Stage), and later prove in zero-knowledge that the committed m satisfies some predicate ϕ (the Prove Stage). What makes this primitive non-trivial is the requirement of black-box use of cryptographic building blocks; otherwise, it can be fulfilled easily by giving a standard commitment to m first, and then running any zero-knowledge system over the commitment in a non-black-box manner.

Our construction follows the classical "MPC-in-the-head" paradigm [47, 37] with the following modifications. To make the commitment stage extractable, we ask the committer to use the ε-simulatable parallel-strongly extractable commitment. We remark that the parallel-strong extractability is essential for obtaining a constant round construction since the committer has to parallelly commit to many secret shares of its message in the construction. Another caveat is that the protocol relies on coin-flipping to conduct a "cut-and-choose" type of argument. As explained in Sec. 2.1, we can implement a "one-sided simulatable" coin flipping from (weakly) extractable commitments. Based on this observation, we upgrade the classical security proof to the quantum setting.

Due to space constraints, we provide a more detailed overview of this construction in [16, Section 2.2].

2.3 Black-Box ε-Simulatable 2PC

It is well-known that there exist black-box constant-round constructions of general-purpose 2PC from semi-honest OTs and (simulation-secure) commitments in the universally-composable (UC) model [49, 42, 19]. In the stand-alone setting, it had been a folklore that a similar conversion works if we assume suitable parallel-simulation-secure commitments, but we are not aware of any work that *formally* proved it until the recent work of [40]. [40] addressed this issue by defining a functionality called $\mathcal{F}^t_{\textsc{so-com}}$, and showed that the above conversion works in the $\mathcal{F}^t_{\textsc{so-com}}$-hybrid model *in the stand-alone setting.* $\mathcal{F}^t_{\textsc{so-com}}$ is a two-party ideal functionality that allows a committer to commit to an a-priori fixed polynomial number $t(\lambda)$ of messages in parallel, and later decommit to a subset of these commitments named by the receiver (thus, "SO" stands for "selectively opening").

Thus, for obtaining a black-box constant-round construction of general-purpose ε-simulatable 2PC, all we need to do is to construct a constant-round black-box commitment scheme that implements $\mathcal{F}^t_{\textsc{so-com}}$ with ε-close simulation. It is straightforward to construct such a commitment scheme based on our ExtCom-and-Prove protocol, since it enables the committer to prove any predicate on committed values, which of course supports revealing a subset of them.

Moreover, if we are allowed to use quantum communication, [40] showed that we can construct black-box constant-round (maliciously-secure) OTs in the $\mathcal{F}^t_{\textsc{so-com}}$-hybrid model. Thus, we can drop the additional assumption of semi-honest OTs in this case.

We provide a more detailed technical overview in [16, Section 2.3].

3 Preliminaries

We postpone basic notations, definitions, and known lemmas to [16, Section 3].

3.1 Post-Quantum Extractable Commitment

We give a definition of post-quantum (strongly) extractable commitments with ε-simulation. We will omit the security parameter from the input to parties when it is clear from the context.

Definition 1 (Post-Quantum Commitment). *A* post-quantum commitment scheme Π *is a classical interactive protocol between interactive* PPT *machines C and R. Let $m \in \{0,1\}^{\ell(\lambda)}$ (where $\ell(\cdot)$ is some polynomial) is a message that C wants to commit to. The protocol consists of the following stages:*

- **Commit Stage:** *$C(m)$ and R interact with each other to generate a transcript (which is also called a commitment) denoted by com,*[16] *C's state ST_C, and R's output $b_{\mathrm{com}} \in \{0,1\}$ indicating acceptance ($i.e., b_{\mathrm{com}} = 1$) or rejection ($i.e., b_{\mathrm{com}} = 0$). We denote this execution by $(\mathsf{com}, \mathsf{ST}_C, b_{\mathrm{com}}) \leftarrow \langle C(m), R\rangle(1^\lambda)$. When C is honest, ST_C is classical, but when we consider a malicious quantum committer $C^*(\rho)$, we allow it to generate any quantum state ST_{C^*}. Similarly, a malicious quantum receiver $R^*(\rho)$ can output any quantum state, which we denote by OUT_{R^*} instead of b_{com}.*
- **Decommit Stage:** *C generates a decommitment decom from ST_C. We denote this procedure by $\mathsf{decom} \leftarrow C(\mathsf{ST}_C)$.*[17] *Then it sends a message m and decommitment decom to R, and R outputs a bit $b_{\mathrm{dec}} \in \{0,1\}$ indicating acceptance ($i.e., b_{\mathrm{dec}} = 1$) or rejection ($i.e., b_{\mathrm{dec}} = 0$). We assume that R's verification procedure is deterministic and denote it by $\mathsf{Verify}(\mathsf{com}, m, \mathsf{decom})$.*[18] *W.l.o.g., we assume that R always rejects (i.e., $\mathsf{Verify}(\mathsf{com}, \cdot, \cdot) = 0$) whenever $b_{\mathrm{com}} = 0$. (Note that w.l.o.g., com can include b_{com} because we can always modify the protocol to ask R to send b_{com} as the last round message.)*

The scheme satisfies the following correctness requirement:

1. **Correctness.** *For any $m \in \{0,1\}^{\ell(\lambda)}$, it holds that*

$$\Pr\left[b_{\mathrm{com}} = b_{\mathrm{dec}} = 1 : \begin{array}{c} (\mathsf{com}, \mathsf{ST}_C, b_{\mathrm{com}}) \leftarrow \langle C(m), R\rangle(1^\lambda) \\ \mathsf{decom} \leftarrow C(\mathsf{ST}_C) \\ b_{\mathrm{dec}} \leftarrow \mathsf{Verify}(\mathsf{com}, m, \mathsf{decom}) \end{array}\right] = 1.$$

Definition 2 (Computationally Hiding). *A post-quantum commitment Π is* computationally hiding *if for any $m_0, m_1 \in \{0,1\}^{\ell(\lambda)}$ and any non-uniform QPT receiver $R^*(\rho)$, the following holds:*

$$\{\mathsf{OUT}_{R^*} : (\mathsf{com}, \mathsf{ST}_C, \mathsf{OUT}_{R^*}) \leftarrow \langle C(m_0), R^*(\rho)\rangle(1^\lambda)\}_\lambda$$
$$\overset{c}{\approx}\{\mathsf{OUT}_{R^*} : (\mathsf{com}, \mathsf{ST}_C, \mathsf{OUT}_{R^*})\langle C(m_1), R^*(\rho)\rangle(1^\lambda)\}_\lambda.$$

[16] That is, we regard the whole transcript as a commitment.

[17] We could define ST_C to be decom itself w.l.o.g. However, we define them separately because this is more convenient when we define ExtCom-and-Prove, which is an extension of post-quantum extractable commitments.

[18] Note that Verify is well-defined since our syntax does not allow R to keep a state from the commit stage.

Definition 3 (Statistically Binding). *A post-quantum commitment* Π *is* statistically binding *if for any unbounded-time comitter* C^**, the following holds:*

$$\Pr\left[\begin{array}{l} \exists\ \{m_b, \mathsf{decom}_b\}_{b\in\{0,1\}}, m_0 \neq m_1 \\ \wedge\ \mathsf{Verify}(\mathsf{com}, m_b, \mathsf{decom}_b) = 1 \\ \quad \textit{for } b \in \{0,1\} \end{array} : (\mathsf{com}, \mathsf{ST}_{C^*}, b_{\mathrm{com}}) \leftarrow \langle C^*, R\rangle(1^\lambda)\right] = \mathsf{negl}(\lambda).$$

Definition 4 (Committed Values). *For a post-quantum commitment* Π*, we define the value function as follows:*

$$\mathsf{val}_\Pi(\mathsf{com}) := \begin{cases} m & \textit{if } \exists \textit{ unique } m \textit{ s.t. } \exists\ \mathsf{decom}, \mathsf{Verify}(\mathsf{com}, m, \mathsf{decom}) = 1 \\ \perp & \textit{otherwise} \end{cases}.$$

We say that com *is valid if* $\mathsf{val}_\Pi(\mathsf{com}) \neq \perp$ *and invalid if* $\mathsf{val}_\Pi(\mathsf{com}) = \perp$.

Then we give the definition of the strong extractability with ε-simulation. The definition is similar to that of post-quantum extractable commitments in [10, 9] except that we allow an (arbitrarily small) noticeable approximation error similarly to post-quantum ε-zero-knowledge [18]. We note that we call it the *strong* extractability since we also define a weaker version of extractability in Def. 7 in Sec. 5.1.

Definition 5 (Strong Extractability with ε-Simulation). *A commitment scheme* Π *is* strongly extractable with ε-simulation *if there exists a QPT algorithm* $\mathcal{SE}$ *(called the* ε*-simulation strong-extractor) such that for any noticeable* $\varepsilon(\lambda)$ *and any non-uniform QPT* $C^*(\rho)$*,*

$$\left\{\mathcal{SE}^{C^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}})\right\}_\lambda \overset{c}{\approx}_\varepsilon \left\{(\mathsf{val}_\Pi(\mathsf{com}), \mathsf{ST}_{C^*}) : (\mathsf{com}, \mathsf{ST}_{C^*}, b_{\mathrm{com}}) \leftarrow \langle C^*(\rho), R\rangle(1^\lambda)\right\}_\lambda.$$

We also define the parallel version.

Definition 6 (Parallel-Strong Extractability with ε-Simulation). *A commitment scheme* Π *is* parallelly strongly extractable with ε-simulation *if for any integer* $n = \mathsf{poly}(\lambda)$*, there exists a QPT algorithm* $\mathcal{SE}_{\mathsf{par}}$ *(called the* ε*-simulation parallel-strong-extractor) such that for any noticeable* $\varepsilon(\lambda)$ *and any non-uniform QPT* $C^*(\rho)$*,*

$$\begin{aligned} &\left\{\mathcal{SE}_{\mathsf{par}}^{C^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}})\right\}_\lambda \\ \overset{c}{\approx}_\varepsilon &\left\{(\Lambda_{\{b_{\mathrm{com},j}\}_{j=1}^n}(\{\mathsf{val}(\mathsf{com}_j)\}_{j=1}^n), \mathsf{ST}_{C^*}) : \begin{array}{l} (\{\mathsf{com}_j\}_{j=1}^n, \mathsf{ST}_{C^*}, \{b_{\mathrm{com},j}\}_{j=1}^n) \\ \quad \leftarrow \langle C^*(\rho), R^n\rangle(1^\lambda) \end{array}\right\}_\lambda \end{aligned}$$

where $(\{\mathsf{com}_j\}_{j=1}^n, \mathsf{ST}_{C^*}, \{b_{\mathrm{com},j}\}_{j=1}^n) \leftarrow \langle C^*(\rho), R^n\rangle(1^\lambda)$ *means that* $C^*(\rho)$ *interacts with* n *copies of the honest receiver* R *in parallel and the execution results in transcripts* $\{\mathsf{com}_j\}_{j=1}^n$*, the final state* ST_{C^*}*, and outputs* $\{b_{\mathrm{com},j}\}_{j=1}^n$ *of each copy of* R *and*

$$\Lambda_{\{b_{\mathrm{com},j}\}_{j=1}^n}(\{\mathsf{val}(\mathsf{com}_j)\}_{j=1}^n) := \begin{cases} \{\mathsf{val}_\Pi(\mathsf{com}_j)\}_{j=1}^n & \textit{if } \forall\ j \in [n]\ b_{\mathrm{com},j} = 1 \\ \perp & \textit{otherwise} \end{cases}.$$

Remark 1. We remark that the above definition only requires the extractor to extract the committed values when R accepts in all the parallel sessions. In particular, when R accepts in some sessions but not in others, the extractor does not need to extract the committed values at all. An alternative stronger (and probably more natural) definition would require the extractor to extract $\mathsf{val}_{\Pi}(\mathsf{com}_j)$ for all $j \in [n]$ such that R accepts in the j-th session. But we define it in the above way since it suffices for our purpose and we do not know if our construction satisfies the stronger one.

4 Extract-and-Simulate Lemma

We prove a lemma that can be seen as an ε-simulation variant of Unruh's rewinding lemma ([64, Lemma 7]) in typical applications. This lemma is the technical core of all the results in this paper.

4.1 Statement of Extract-and-Simulate Lemma

Our lemma is stated as follows.

Lemma 1 (Extract-and-Simulate Lemma). *Let C be a finite set. Let $\{\Pi_i\}_{i\in C}$ be orthogonal projectors on a Hilbert space $\mathcal{H}$ such that the measurement $\{\Pi_i, I-\Pi_i\}$ can be efficiently implemented. Let $|\psi_{\mathsf{init}}\rangle \in \mathcal{H}$ be a unit vector.*

Suppose that there are a subset $S \in C^2$ and a QPT algorithm $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ that satisfies the following:

1. *S consists of an overwhelming fraction of C^2, i.e., $\frac{|S|}{|C|^2} = 1 - \mathsf{negl}(\lambda)$.*
2. *For all $i \in C$, there exists a classical string s_i such that*
$$\Pr\left[\mathcal{A}_0\left(i, \frac{\Pi_i |\psi_{\mathsf{init}}\rangle}{\|\Pi_i |\psi_{\mathsf{init}}\rangle\|}\right) = s_i\right] = 1.$$
3. *There exists a classical string s^* such that for any $(i,j) \in S$,*
$$\Pr[\mathcal{A}_1(i, j, s_i, s_j) = s^*] = 1.$$

Let $\mathsf{Exp}(\lambda, \{\Pi_i\}_{i\in C}, |\psi_{\mathsf{init}}\rangle)$ be an experiment that works as follows:

- *Choose $i \leftarrow C$.*
- *Apply the measurement $\{\Pi_i, I-\Pi_i\}$ on $|\psi_{\mathsf{init}}\rangle$.*
  - *If the state is projected onto Π_i, the experiment outputs i, the classical string s^*, and the resulting state $\frac{\Pi_i|\psi_{\mathsf{init}}\rangle}{\|\Pi_i|\psi_{\mathsf{init}}\rangle\|}$.*[19]
  - *If the state is projected onto $I - \Pi_i$, the experiment outputs i, $\perp$, and the resulting state $\frac{(I-\Pi_i)|\psi_{\mathsf{init}}\rangle}{\|(I-\Pi_i)|\psi_{\mathsf{init}}\rangle\|}$.*

Then, there is a QPT algorithm $\mathcal{SE}$ such that for any noticeable ε,
$$\{\mathcal{SE}(1^{\lambda}, 1^{\varepsilon^{-1}}, \{\Pi_i\}_{i\in C}, \mathcal{A}, |\psi_{\mathsf{init}}\rangle)\}_{\lambda} \overset{s}{\approx}_{\varepsilon} \{\mathsf{Exp}(\lambda, \{\Pi_i\}_{i\in C}, |\psi_{\mathsf{init}}\rangle)\}_{\lambda}.$$

Due to space constraints, we postpone the proof to the full version [16, Section 4.2]. But note that the key ideas of this proof are already described in Sec. 2.1.

[19] We stress that we do not assume that the experiment is efficient. Especially, it may be computationally hard to find s^* from $\frac{\Pi_i|\psi_{\mathsf{init}}\rangle}{\|\Pi_i|\psi_{\mathsf{init}}\rangle\|}$.

5 Black-Box ε-Simulation-Extractable Commitments in Constant Rounds

In this section, we present our construction of post-quantum commitment that satisfies the (parallel) strong extractability with ε-simulation. Namely, we prove the following lemma.

Lemma 2. *Assume the existence of post-quantum secure OWFs. Then, there exists a constant-round construction of post-quantum commitment that satisfies computational hiding (Def. 2), statistical binding (Def. 3), and (parallel) strongly extractable commitment with ε-simulation. Moreover, this construction makes only black-box use of the assumed OWF.*

Toward proving that, we first construct a scheme that satisfies a weaker notion of ε-simulatable extractability in Sec. 5.1. In Sec. 5.2, we present a compiler that converts the weak scheme in Sec. 5.1 into one that satisfies the (parallel) strong extractability with ε-simulation.

5.1 Weakly Extractable Commitment

We construct a commitment scheme that satisfies weak notions of extractability defined in Def. 7 and 10 based on OWFs. The description of the scheme is given in Prot. 1, where Com is a statistically-binding and computationally-hiding commitment scheme (e.g., Naor's commnitment). We remark that the scheme is identical to the *classical* extractable commitment in [61], which in turn is based on earlier works [25, 62, 63].

Protocol 1: Extractable Commitment Scheme $\mathsf{wExtCom}$

The extractable commitment scheme, based on any commitment scheme Com, works in the following way.

Input:

- both the committer C and the receiver R get security parameter 1^λ as the common input.
- C gets a string $m \in \{0,1\}^{\ell(\lambda)}$ as his private input, where $\ell(\cdot)$ is a polynomial

Commitmment Phase:

1. The committer C commits using Com to $k = \lambda$ pairs of strings $\{(v_i^0, v_i^1)\}_{i=1}^k$ where $(v_i^0, v_i^1) = (\eta_i, m \oplus \eta_i)$ and η_i are random strings in $\{0,1\}^\ell$ for $1 \leq i \leq k$.[20]We denote those commitments by $\overline{\mathsf{com}} = \{\mathsf{com}_i^0, \mathsf{com}_i^1\}_{i=1}^k$.
2. Upon receiving a challenge $\mathbf{c} = (c_1, \ldots, c_k)$ from the receiver R, S opens the commitments to $\mathbf{v} := (v_1^{c_1}, \ldots, v_k^{c_k})$ with the corresponding decommitment $\overline{\mathsf{decom}} := (\mathsf{decom}_1^{c_1}, \ldots, \mathsf{decom}_k^{c_k})$.
3. R checks that the openings are valid.

Decommitment Phase:

- C sends σ and opens the commitments to all k pairs of strings. R checks that all the openings are valid, and also that $m = v_1^0 \oplus v_1^1 = \cdots = v_k^0 \oplus v_k^1$.

[20] Actually, the scheme will be secure as long as we use Com to commit $k = \omega(\log \lambda)$ pairs of strings.

Proof of Security. The correctness and the statistically-binding property of $\mathsf{wExtCom}$ follows straightforwardly from that of Com. The computationally-hiding property of $\mathsf{wExtCom}$ can be reduced to that of Com by standard arguments.

Lemma 3 (Computational Hiding). $\mathsf{wExtCom}$ *is computationally hiding.*

The proof is similar to the classical counterpart in [61]. We postpone it to the full version [16, Section 5.1].

We prove that $\mathsf{wExtCom}$ satisfies a weak version of the extractability which we call the *weak extractability with* ε*-simulation.* Intuitively, it requires the simulation-extractor to perform extraction *and* ε-simulation properly, *as long as the commitment is valid.* A formal definition is given below.

Definition 7 (Weak Extractability with ε-Simulation). *A commitment scheme* Π *is* weakly extractable with ε-simulation *if there exists a QPT algorithm* $\mathcal{SE}_{\mathsf{weak}}$ *(called the* ε*-simulation weak-extractor) such that for any noticeable* $\varepsilon(\lambda)$ *and any non-uniform QPT* $C^*(\rho)$*,*

$$\begin{aligned}&\left\{\mathit{\Gamma}_{\mathsf{com}}(m_{\mathsf{Ext}}, \widetilde{\mathsf{ST}}_{C^*}) : (\mathsf{com}, m_{\mathsf{Ext}}, \widetilde{\mathsf{ST}}_{C^*}) \leftarrow \mathcal{SE}^{C^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}})\right\}_\lambda \\ \overset{c}{\approx}_\varepsilon &\left\{\mathit{\Gamma}_{\mathsf{com}}(\mathsf{val}_\Pi(\mathsf{com}), \mathsf{ST}_{C^*}) : (\mathsf{com}, \mathsf{ST}_{C^*}, b_{\mathsf{com}}) \leftarrow \langle C^*(\rho), R\rangle(1^\lambda)\right\}_\lambda\end{aligned}$$

where $\mathit{\Gamma}_{\mathsf{com}}(m, \mathsf{ST}_{C^*}) := \begin{cases}(m, \mathsf{ST}_{C^*}) & \textit{if } \mathsf{val}_\Pi(\mathsf{com}) \neq \bot \\ \bot & \textit{otherwise}\end{cases}.$

Lemma 4 (Weak Extractability with ε-Simulation). $\mathsf{wExtCom}$ *is weakly extractable with* ε*-simulation (as per Def. 7).*

Before proving Lem. 4, we prepare several definitions.

Definition 8 (Validness of $\overline{\mathsf{com}}$). *For a sequence* $\overline{\mathsf{com}} = \{\mathsf{com}_i^0, \mathsf{com}_i^1\}_{i=1}^k$ *of commitments of the scheme* Com*, we say that* $\overline{\mathsf{com}}$ *is valid if there exists* $m \in \{0,1\}^\ell$ *such that* $\mathsf{val}_{\mathsf{Com}}(\mathsf{com}_i^b) \neq \bot$ *for all* $i \in [k]$ *and* $b \in \{0,1\}$ *and* $\mathsf{val}_{\mathsf{Com}}(\mathsf{com}_i^0) \oplus \mathsf{val}_{\mathsf{Com}}(\mathsf{com}_i^1) = m$ *for all* $i \in [k]$ *where* $\mathsf{val}_{\mathsf{Com}}(\mathsf{com}_i^b)$ *is the value function as defined in Def. 4. We denote by* $\mathsf{val}_{\mathsf{Com}}(\overline{\mathsf{com}})$ *to mean such* m *if* $\overline{\mathsf{com}}$ *is valid and otherwise* $\bot$*.*

Definition 9 (Accepting Opening of $\overline{\mathsf{com}}$). *For a sequence* $\overline{\mathsf{com}} = \{\mathsf{com}_i^0, \mathsf{com}_i^1\}_{i=1}^k$ *of commitments of the commitment scheme* Com *and* $\mathbf{c} = (c_1, ..., c_k) \in \{0,1\}^k$*, we say that* $(\mathbf{v} = (v_1, ..., v_k), \overline{\mathsf{decom}} = (\mathsf{decom}_1, ..., \mathsf{decom}_k))$ *is an accepting opening of* $\overline{\mathsf{com}}$ *w.r.t.* $\mathbf{c}$ *if* $\mathsf{Verify}_{\mathsf{Com}}(\mathsf{com}_i^{c_i}, v_i, \mathsf{decom}_i) = 1$ *for all* $i \in [k]$*.*

Then we prove Lem. 4.

Proof of Lem. 4. For simplicity, we assume that Com satisfies perfect binding. It is straightforward to extend the proof to the statistically binding case by excluding the bad case where any commitment of Com is not bounded to a unique message, which happens with a negligible probability.

Remark that the weak extractability with ε-simulation only requires the extractor to correctly extract and simulate if the commitment generated in the commit stage is valid in the sense of Def. 4. When the commitment is valid, $\overline{\mathsf{com}}$ generated in Step 1 is also valid in the sense of Def. 8 (because otherwise a committer cannot pass the verification in the decommitment stage). Therefore, it suffices to prove that the extractor works for any fixed valid $\overline{\mathsf{com}}$.

Let $C^*(\rho)$ be a non-uniform QPT malicious committer. For $\mathbf{c} \in \{0,1\}^k$, let $U_{\mathbf{c}}$ be the unitary corresponding to the action of C^* in Step 2. That is, for the state ρ' before Step 2, it applies $U_{\mathbf{c}}$ to get $U_{\mathbf{c}}\rho' U_{\mathbf{c}}^{\dagger}$ and measures designated registers $\mathbf{V}$ and $\mathbf{D}$ to get the message $\mathbf{v}$ and opening information $\overline{\mathsf{decom}}$ in Step 2. Let $\Pi_{\mathbf{c}}^{\mathsf{test}}$ be the projection that maps onto states that contain an accepting opening $\mathbf{v}$ and $\overline{\mathsf{decom}}$ of $\overline{\mathsf{com}}$ w.r.t. $\mathbf{c}$ (as defined in Def. 9) in $\mathbf{V} \otimes \mathbf{D}$. For $\mathbf{c} \in \{0,1\}^k$, we define $\Pi_{\mathbf{c}} := U_{\mathbf{c}}^{\dagger} \Pi_{\mathbf{c}}^{\mathsf{test}} U_{\mathbf{c}}$.

We apply Lem. 1 for $\{\Pi_{\mathbf{c}}\}_{\mathbf{c}\in\{0,1\}^k}$ with the following correspondence.

- $\mathcal{H}$ is the internal space of C^*.
- The initial state is ρ'.[21]
- $C = \{0,1\}^k$.
- $S = \{((c_1, ..., c_k), (c'_1, ..., c'_k)) : \exists i \in [k] \text{ s.t. } c_i \neq c'_i\}$
- $\mathcal{A}_0$ applies $U_{\mathbf{c}}$ on its input, measures $\mathbf{V}$ to get $\mathbf{v}$, applies $U_{\mathbf{c}}^{\dagger}$, and outputs $\mathbf{v}$.
- $\mathcal{A}_1$ is given as input $(\mathbf{c}, \mathbf{c}') \in S$, $\mathbf{v}_{\mathbf{c}} = (v_1^{c_1}, ..., v_k^{c_k})$, and $\mathbf{v}_{\mathbf{c}'} = (v_1^{c'_1}, ..., v_k^{c'_k})$. $\mathcal{A}_1$ outputs $v_i^{c_i} \oplus v_i^{c'_i}$ for the smallest $i \in [k]$ such that $c_i \neq c'_i$. Note that such i exists since we assume $(\mathbf{c}, \mathbf{c}') \in S$.

If $\overline{\mathsf{com}}$ is valid, we can see that the assumptions for Lem. 1 are satisfied as follows:

1. By the definition of S, it is easy to see that $\frac{|S|}{|C|^2} = 1 - 2^{-k} = 1 - \mathsf{negl}(\lambda)$.
2. For any $\mathbf{c}$, if $\mathcal{A}_0$ takes a state in the span of $\Pi_{\mathbf{c}}$ as input, it outputs $s_{\mathbf{c}} := (\mathsf{val}_{\mathsf{Com}}(\mathsf{com}_1^{c_1}), ..., \mathsf{val}_{\mathsf{Com}}(\mathsf{com}_k^{c_k}))$ with probability 1 by the definition of $\Pi_{\mathbf{c}}$ and the perfect binding property of Com.
3. For any $(\mathbf{c}, \mathbf{c}') \in S$, if $\mathcal{A}_1$ takes as input the $s_{\mathbf{c}}$ and $s_{\mathbf{c}'}$ defined as follows:
$$\begin{cases} s_{\mathbf{c}} = (\mathsf{val}_{\mathsf{Com}}(\mathsf{com}_1^{c_1}), \ldots, \mathsf{val}_{\mathsf{Com}}(\mathsf{com}_k^{c_k})) \\ s_{\mathbf{c}'} = (\mathsf{val}_{\mathsf{Com}}(\mathsf{com}_1^{c'_1}), \ldots, \mathsf{val}_{\mathsf{Com}}(\mathsf{com}_k^{c'_k})) \end{cases};$$
then, it outputs $s^* := \mathsf{val}_{\mathsf{Com}}(\overline{\mathsf{com}})$ as defined in Def. 8 since we assume that $\overline{\mathsf{com}}$ is valid.

Let $\widetilde{\mathcal{SE}}$ be the ε-simulation extractor of Lem. 1 in the above setting. Then Lem. 1 gives us the following:

$$\{\widetilde{\mathcal{SE}}(1^{\lambda}, 1^{\varepsilon^{-1}}, \{\Pi_{\mathbf{c}}\}_{\mathbf{c}\in\{0,1\}^k}, \mathcal{A}, \rho')\}_{\lambda} \overset{\mathrm{s}}{\approx}_{\varepsilon} \{\mathsf{Exp}(\lambda, \{\Pi_{\mathbf{c}}\}_{\mathbf{c}\in\{0,1\}^k}, \rho')\}_{\lambda}$$

where $\mathsf{Exp}(\lambda, \{\Pi_{\mathbf{c}}\}_{\mathbf{c}\in\{0,1\}^k}, \rho')$ is as defined in Lem. 1. That is, $\mathsf{Exp}(\lambda, \{\Pi_{\mathbf{c}}\}_{\mathbf{c}\in\{0,1\}^k}, \rho')$ works as follows:

[21] Though we assume that the initial state $|\psi_{\mathsf{init}}\rangle$ is a pure state in Lem. 1, the lemma holds for any mixed state since a mixed state can be seen as a probability distribution over pure states.

– Choose $\mathbf{c} \leftarrow \{0,1\}^k$.
– Apply the measurement $\{\mathit{\Pi}_{\mathbf{c}}, I - \mathit{\Pi}_{\mathbf{c}}\}$ on ρ'.
  - If the state is projected onto $\mathit{\Pi}_{\mathbf{c}}$, the experiment outputs $\mathbf{c}$, the classical string $\mathsf{val}_{\mathsf{Com}}(\overline{\mathsf{com}})$, and the resulting state.
  - If the state is projected onto $I - \mathit{\Pi}_{\mathbf{c}}$, the experiment outputs $\mathbf{c}$, $\perp$, and the resulting state.

One can see that the state in the third output of $\mathsf{Exp}(\lambda, \rho')$ is similar to the final state of C^* in the real execution except that C^* applies the unitary $U_{\mathbf{c}}$ instead of the measurement $\{\mathit{\Pi}_{\mathbf{c}}, I - \mathit{\Pi}_{\mathbf{c}}\}$ and measures $\mathbf{V}$ and $\mathbf{D}$. By noting that $\mathit{\Pi}_{\mathbf{c}}^{\mathsf{test}} U_{\mathbf{c}} = U_{\mathbf{c}} \mathit{\Pi}_{\mathbf{c}}$ and that measuring $\mathbf{V}$ and $\mathbf{D}$ is the same as first applying the measurement $\{\mathit{\Pi}_{\mathbf{c}}^{\mathsf{test}}, I - \mathit{\Pi}_{\mathbf{c}}^{\mathsf{test}}\}$ and then measuring $\mathbf{V}$ and $\mathbf{D}$, if we apply $U_{\mathbf{c}}$ on the third output of $\mathsf{Exp}(\lambda, \{\mathit{\Pi}_{\mathbf{c}}\}_{\mathbf{c} \in \{0,1\}^k}, \rho')$ and then measure $\mathbf{V}$ and $\mathbf{D}$, the state is exactly the same as the final state of C^*.

Therefore, the following extractor $\mathcal{SE}_{\mathsf{weak}}$ works for the weak ε-simulation extractability:

$\mathcal{SE}_{\mathsf{weak}}^{C^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}})$:

1. Run the commit stage of $\mathsf{wExtCom}$ between $C^*(\rho)$ and the honest receiver R until C^* sends $\overline{\mathsf{com}}$ in Step 1. Let ρ' be the internal state of C^* at this point.
2. Run $(\mathbf{c}, m_{\mathsf{Ext}}, \rho_{\mathsf{Ext}}) \leftarrow \widetilde{\mathcal{SE}}(1^\lambda, 1^{\varepsilon^{-1}}, \{\mathit{\Pi}_{\mathbf{c}}\}_{\mathbf{c} \in \{0,1\}^k}, \mathcal{A}, \rho')$ where $\mathcal{A} = (\mathcal{A}_0, \mathcal{A}_1)$ is as defined above. Remark that the definition of $\mathit{\Pi}_{\mathbf{c}}$ depends on $\overline{\mathsf{com}}$, and it uses $\overline{\mathsf{com}}$ generated in the previous step.
3. Apply $U_{\mathbf{c}}$ on ρ_{Ext} to generate $U_{\mathbf{c}} \rho_{\mathsf{Ext}} U_{\mathbf{c}}^\dagger$ and measures registers $\mathbf{V}$ and $\mathbf{D}$ to get $\mathbf{v}$ and $\overline{\mathsf{decom}}$. Let ρ_{final} be the state after the measurement.
4. Output $(m_{\mathsf{Ext}}, \rho_{\mathsf{final}})$.

□

On the Parallel Execution of wExtCom. We can prove that $\mathsf{wExtCom}$ satisfies a parallel version of the weak extractability with ε-simulation in a similar way. In the following, we prove that $\mathsf{wExtCom}$ satisfies even a generalized version of that, which we call *special parallel weak extractability with ε-simulation.* Looking ahead, this will be used in the proof of the (parallel) ε-simulation strong extractability of Prot. 2 in Sec. 5.2.

Intuitively, it requires the following: Suppose that a malicious committer C^* interacts with n copies of the honest receiver R in parallel, and let com_j be the commitment generated in the j-th execution. Suppose that com_j is valid for all $j \in V$ for some subset $V \subseteq [n]$. Let $F : \{0,1\}^\ell \cup \{\perp\} \rightarrow \{0,1\}^*$ be a function that is determined by $\{\mathsf{val}(\mathsf{com}_j)\}_{j \in V}$, i.e., $F(m_1, ..., m_n)$ takes a unique value m^* as long as $m_j = \mathsf{val}(\mathsf{com}_j)$ for all $j \in V$. Then, the extractor can extract m^* while simulating the post-execution state of C^*. A formal definition is given below.

Definition 10 (Special Parallel Weak Extractability with ε-Simulation). *We say that a commitment scheme* Π *satisfies the* special parallel weak extractability with ε-simulation *if the following is satisfied. For any integer* $n = \mathsf{poly}(\lambda)$ *and an efficiently computable function* $F : \{\{0,1\}^\ell \cup \{\bot\}\}^n \to \{0,1\}^*$, *there exists* $\mathcal{SE}_F$ *that satisfies the following: For commitments* $\{\mathsf{com}_j\}_{j=1}^n$*, we say that* $\{\mathsf{com}_j\}_{j=1}^n$ *is* F*-good if it satisfies the following:*

1. *there exists* $V \subseteq [n]$ *such that* com_j *is valid (i.e.,* $\mathsf{val}_\Pi(\mathsf{com}_j) \neq \bot$*) for all* $j \in V$*;* **and**
2. *there exists a unique* m^* *such that* $F(m'_1, ..., m'_n) = m^*$ *for all* $(m'_1, ..., m'_n)$ *such that* $m'_j = \mathsf{val}_\Pi(\mathsf{com}_j)$ *for all* $j \in V$*.*

Then it holds that

$$\left\{ \mathit{\Gamma}_{F,\{\mathsf{com}_j\}_{j=1}^n}(m_{\mathsf{Ext}}, \mathsf{ST}_{C^*}) : (\{\mathsf{com}_j\}_{j=1}^n, m_{\mathsf{Ext}}, \mathsf{ST}_{C^*}) \leftarrow \mathcal{SE}_F^{C^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}}) \right\}_\lambda$$

$$\overset{c}{\approx}_\varepsilon \left\{ \begin{array}{l} \mathit{\Gamma}_{F,\{\mathsf{com}_j\}_{j=1}^n}(F(\mathsf{val}_\Pi(\mathsf{com}_1), ..., \mathsf{val}_\Pi(\mathsf{com}_n)), \mathsf{ST}_{C^*}) \\ : (\{\mathsf{com}_j\}_{j=1}^n, \mathsf{ST}_{C^*}, \{b_{\mathrm{com},j}\}_{j=1}^n) \leftarrow \langle C^*(\rho), R^n \rangle(1^\lambda) \end{array} \right\}_\lambda,$$

where $(\{\mathsf{com}_j\}_{j=1}^n, \mathsf{ST}_{C^*}, \{b_{\mathrm{com},j}\}_{j=1}^n) \leftarrow \langle C^*(\rho), R^n \rangle(1^\lambda)$ *means that* $C^*(\rho)$ *interacts with* n *copies of the honest receiver* R *in parallel and the execution results in transcripts* $\{\mathsf{com}_j\}_{j=1}^n$*, the final state* ST_{C^*}*, and outputs* $\{b_{\mathrm{com},j}\}_{j=1}^n$ *of each copy of* R *and*

$$\mathit{\Gamma}_{F,\{\mathsf{com}_j\}_{j=1}^n}(m, \mathsf{ST}_{C^*}) := \begin{cases} (m, \mathsf{ST}_{C^*}) & \textit{if } \{\mathsf{com}_j\}_{j=1}^n \textit{ is } F\textit{-good} \\ \bot & \textit{otherwise} \end{cases}.$$

Lemma 5 (Special Parallel Weak Extractability with ε-Simulation). $\mathsf{wExtCom}$ *satisfies the special parallel weak extractability with* ε*-simulation (as per Def. 10).*

The proof of Lem. 5 is similar to that of Lem. 4. Due to space constraints, we postpone it to the full version [16, Section 5.1].

5.2 Strongly Extractable Commitment

In this section, we present the *strongly* extractable commitment with ε-simulation. The scheme is shown in Prot. 2. It relies on the following building blocks:

1. the ε-simulatable *weakly* extractable commitment $\mathsf{wExtCom}$ given in Prot. 1. We remark that the security of Prot. 2 relies on the particular $\mathsf{wExtCom}$ presented in Prot. 1 because we also need the special parallel weak extractability with ε-simulation (Def. 10); we do not know if Prot. 2 can be based on any $\mathsf{wExtCom}$ satisfying the weak extractability with ε-simulation as in Def. 7.
2. a $(n+1, t)$-perfectly verifiable secret sharing scheme $\mathsf{VSS} = (\mathsf{VSS}_{\mathsf{Share}}, \mathsf{VSS}_{\mathsf{Recon}})$. We require that t is a constant fraction of n such that $t \leq n/3$. There are known constructions (without any computational assumptions) satisfying these properties [6, 20].

Protocol 2: ε-Simulatable Strongly Extractable Commitment ExtCom

Let $n(\lambda)$ be a polynomial on λ. Let t be a constant fraction of n such that $t \leq n/3$.

Input: both the (committer) C and the receiver R get security parameter 1^λ as the common input; C gets a string $m \in \{0,1\}^{\ell(\lambda)}$ as his private input, where $\ell(\cdot)$ is a polynomial.

Commit Stage:

1. C emulates $n+1$ (virtual) players $\{P_i\}_{i\in[n+1]}$ to execute the $\mathsf{VSS}_{\mathsf{Share}}$ protocol "in his head", where the input to P_{n+1} (i.e., the Dealer) is m. Let $\{\mathsf{v}_i\}_{i\in[n+1]}$ be the views of the $n+1$ players describing the execution.
2. C and R involve in n executions of wExtCom in parallel, where in the i-th instance ($i \in [n]$), C commits to v_i.
3. R picks a random string r_1 and commits to it using wExtCom.
4. C picks a random string r_2 and sends it to R.
5. R sends to C the value r_1 together with the corresponding decommitment information w.r.t. the wExtCom in Step 3. Now, both parties learn a coin-tossing result $r = r_1 \oplus r_2$, which specifies a size-t random subset $T \subseteq [n]$.
6. C sends to R in *one round* the following messages: $\{\mathsf{v}_i\}_{i\in T}$ together with the corresponding decommitment information w.r.t. the wExtCom in Step 2.
7. R checks the following conditions:
   (a) All the decommitments in Step 6 are valid; **and**
   (b) for any $i, j \in T$, views $(\mathsf{v}_i, \mathsf{v}_j)$ are consistent w.r.t. the $\mathsf{VSS}_{\mathsf{Share}}$ execution as described in Step 1.

   If all the checks pass, R accepts; otherwise, R rejects.

Decommit Stage:

1. C sends $\{\mathsf{v}_i\}_{i\in[n]}$ together with all the corresponding information w.r.t. the wExtCom in Step 1 of the Commit Stage.
2. R constructs $\{\mathsf{v}'_i\}_{i\in[n]}$ as follows: in Step 1 of the Decommit Stage, if the i-th decommitment is valid, R sets $\mathsf{v}'_i := \mathsf{v}_i$; otherwise, R sets $\mathsf{v}'_i := \bot$.
3. R outputs $m' := \mathsf{VSS}_{\mathsf{Recon}}(\mathsf{v}'_1, \ldots, \mathsf{v}'_n)$.

Security. Correctness and statistical binding property of ExtCom follows straightforwardly from that of wExtCom. We show that ExtCom is computationally-hiding and (parallel) strong extractable with ε-simulation.

Lemma 6 (Computational Hiding). ExtCom *is computationally hiding.*

Computational hinding property can be shown based on the weak extractability of wExtCom used in Step 3, computational hiding property of wExtCom used in Step 2, and the secrecy property of VSS by a standard hybrid argument. The proof is postponed to the full version [16, Section 5.2].

In the following, we prove the (parallel-)strong extractability with ε-simulation. Though we finally prove the parallel version, we first give a proof for the stand-alone version since that is simpler and the proof is readily extended to that of the parallel version.

Lemma 7 (Strong Extractability with ε-Simulation). ExtCom *is strongly extractable with ε-simulation (as per Def. 5).*

Proof. Suppose that a non-uniform QPT committer C^* interacts with the honest receiver R in the commit stage of ExtCom. We consider two cases where R accepts or rejects, respectively. By using Watrous' rewinding lemma [65] in a similar way to the proof of Lem. 1, it suffices to construct a simulator that correctly extracts and simulates for each case separately. Moreover, when R rejects, the commitment is invalid and thus the extractor does not need to extract anything. Thus, there is a trivial perfect simulation extractor for this case: it can simply run the interaction between $C^*(\rho)$ and R by playing the role of R and outputs the final state of C^*. What is left is to construct an extractor that correctly extracts and simulates assuming that R accepts in the committing stage. That is, it suffices to prove the following claim.

Claim 1 (Extraction and Simulation for Accepting Case)**.** *There exists a QPT algorithm $\mathcal{SE}_{\mathsf{Acc}}$ such that for any noticeable $\varepsilon(\lambda)$ and any non-uniform QPT $C^*(\rho)$, it holds that*

$$\begin{aligned}&\left\{\Gamma_{b_{\mathrm{com}}}(m_{\mathsf{Ext}}, \mathsf{ST}_{C^*}) : (m_{\mathsf{Ext}}, \mathsf{ST}_{C^*}, b_{\mathrm{com}}) \leftarrow \mathcal{SE}_{\mathsf{Acc}}^{C^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}})\right\}_\lambda \\ \overset{c}{\approx}_\varepsilon &\left\{\Gamma_{b_{\mathrm{com}}}(\mathsf{val}_{\mathsf{ExtCom}}(\mathsf{com}), \mathsf{ST}_{C^*}) : (\mathsf{com}, \mathsf{ST}_{C^*}, b_{\mathrm{com}}) \leftarrow \langle C^*(\rho), R\rangle(1^\lambda)\right\}_\lambda,\end{aligned}$$

where $\Gamma_{b_{\mathrm{com}}}(m, \mathsf{ST}_{C^*}) := \begin{cases}(m, \mathsf{ST}_{C^*}) & \textit{if } b_{\mathrm{com}} = 1 \\ \bot & \textit{otherwise}\end{cases}.$

Remark 2. One may think that the above claim is similar to the weak extractability with ε-simulation (Def. 7). However, the crucial difference is that the extractor $\mathcal{SE}_{\mathsf{Acc}}$ should declare if the simulation has succeeded by outputting b_{com} in the clear. On the other hand, in Def. 7, $\mathcal{SE}_{\mathsf{weak}}$ is only required to indirectly declare that depending on if com is valid, which may not be known by $\mathcal{SE}_{\mathsf{weak}}$.

Proof of Claim 1. Let $\mathsf{wExtCom.com}_i$ be the i-th commitment of $\mathsf{wExtCom}$ in Step 2 in the commit stage. In the execution of $(\mathsf{com}, \mathsf{ST}_{C^*}, b_{\mathrm{com}}) \leftarrow \langle C^*(\rho), R\rangle(1^\lambda)$, let Good be the event that $\{\mathsf{wExtCom.com}_i\}_{i=1}^n$ is $\mathsf{VSS}_{\mathsf{Recon}}$-good in the sense of Def. 10, i.e.,

- there exists $V \subseteq [n]$ such that $\mathsf{wExtCom.com}_i$ is valid (i.e., $\mathsf{val}_{\mathsf{wExtCom}}(\mathsf{wExtCom.com}_i) \neq \bot$) for all $i \in V$, and
- there exists m^* such that $\mathsf{VSS}_{\mathsf{Recon}}(\mathsf{v}'_1, \ldots, \mathsf{v}'_n) = m^*$ for all $(\mathsf{v}'_1, \ldots, \mathsf{v}'_n)$ such that
$$\forall i \in V,\ \mathsf{v}'_i = \mathsf{val}_{\mathsf{wExtCom}}(\mathsf{wExtCom.com}_i).$$

Let Bad be the complementary event of Good. We prove the following claim.

Claim 2. *It holds that*

$$\Pr\left[\mathsf{Bad} \wedge b_{\mathrm{com}} = 1 : (\mathsf{com}, \mathsf{ST}_{C^*}, b_{\mathrm{com}}) \leftarrow \langle C^*(\rho), R\rangle(1^\lambda)\right] = \mathsf{negl}(\lambda). \quad (2)$$

Claim 2 can be proven based on a similar argument to those used in previous black-box commit-and-prove literature [47, 37, 39, 54]. We postpone the proof to [16, Section 5.2].

Given Claim 2, it is straightforward to finish the proof of Claim 1 by using Lem. 5. Claim 2 means that the Good occurs whenever $b_{\mathrm{com}} = 1$ except for negligible probability. Since $\mathcal{SE}_{\mathsf{Acc}}$ is only required to correctly extract and simulate when $b_{\mathrm{com}} = 1$, it suffices to give an extractor that correctly extracts and simulates when $\{\mathsf{wExtCom.com}_i\}_{i=1}^n$ satisfies the condition for Good. Since $\mathsf{wExtCom}$ satisfies the special parallel weak extractability with ε-simulation as shown in Lem. 5, $\mathcal{SE}_{\mathsf{VSS}_{\mathsf{Recon}}}$ given in Def. 10 (where we set $F := \mathsf{VSS}_{\mathsf{Recon}}$) directly gives $\mathcal{SE}_{\mathsf{Acc}}$. Specifically, $\mathcal{SE}_{\mathsf{Acc}}$ as described below suffices for Claim 1.

$\mathcal{SE}_{\mathsf{Acc}}^{C^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}})$:

1. Run $(\{\mathsf{wExtCom.com}_i\}_{i=1}^n, m_{\mathsf{Ext}}, \mathsf{ST}_{C_2^*}) \leftarrow \mathcal{SE}_{\mathsf{VSS}_{\mathsf{Recon}}}^{C_2^*(\rho)}(1^\lambda, 1^{\varepsilon^{-1}})$ where C_2^* denotes the action of C^* until Step 2 in the commit stage where it outputs $\{\mathsf{wExtCom.com}_i\}_{i=1}^n$.
2. Simulate the interaction between C^* and R from Step 3 where the state of C^* is initialized to be $\mathsf{ST}_{C_2^*}$. Let b_{com} be R's decision (i.e., $b_{\mathrm{com}} = 1$ if and only if R accepts) and ST_{C^*} be the post-execution state of S
3. Output $(m_{\mathsf{Ext}}, \mathsf{ST}_{C^*}, b_{\mathrm{com}})$.

This finishes the proof of Claim 1. □

This eventually concludes the proof of Lem. 7. □

The above proof can be extended to prove the parallel-strong extractability (i.e. Lem. 8). We postpone it to the full version [16, Section 5.2].

Lemma 8 (Parallel-Strong Extractability with ε-Simulation). ExtCom *is parallel-strongly extractable with ε-simulation.*

6 Black-Box ε-Simulatable ExtCom-and-Prove in Constant Rounds

Roughly speaking, ε-simulatable ExtCom-and-Prove is a strongly extractable commitment scheme with ε-simulation with the additional functionality that the committer can later prove any statement of the committed message. Besides the security requirements as strongly extractable commitments with ε-simulation, we additionally require soundness, which states that the committer cannot prove a false statement on the committed message, and ε-zero-knowledge property, which is defined similarly to in [18]. See [16, Definition 17] for the formal definition.

We show the following lemma.

Lemma 9. *Assume the existence of post-quantum secure OWFs. Then, there exists a constant-round ε-simulatable ExtCom-and-Prove scheme. Moreover, this construction makes only black-box use of the assumed OWF.*

Construction. The construction is shown in Prot. 3. It makes black-box use of the following building blocks:

1. The ε-simulatable, *parallel-strong* extractable commitment ExtCom constructed in Sec. 5.2, which in turn makes black-box use of any post-quantum secure OWFs.
2. A statistically-binding, computationally-hiding (against QPT adversaries) commitment Com. This is also known assuming only black-box access to post-quantum secure OWFs.
3. A $(n+1, t)$-perfectly secure verifiable secret sharing scheme $\mathsf{VSS} = (\mathsf{VSS}_{\mathsf{Share}}, \mathsf{VSS}_{\mathsf{Recon}})$ (see [16, Section 3.3]);
4. A (n, t)-perfectly secure MPC protocol Π_{MPC} (see [16, Section 3.4]).

For the VSS and MPC protocols, we require that t is a constant fraction of n such that $t \leq n/3$. There are information-theoretical constructions satisfying these properties [6, 20].

Protocol 3: ε-Simulatable ExtCom-and-Prove

Parameter Setting: Let $n(\lambda)$ be a polynomial on λ. Let t be a constant fraction of n such that $t \leq n/3$.

Input: Both P and the receiver V get 1^λ as the common input; P gets a string $m \in \{0,1\}^{\ell(\lambda)}$ as his private input, where $\ell(\cdot)$ is a polynomial.

Commit Stage:

1. P emulates $n+1$ (virtual) players $\{P_i\}_{i\in[n+1]}$ to execute the $\mathsf{VSS}_{\mathsf{Share}}$ protocol "in his head", where the input to P_{n+1} (i.e., the Dealer) is m. Let $\{\mathsf{v}_i\}_{i\in[n+1]}$ be the views of the $n+1$ players describing the execution.
2. P and V involve in n executions of ExtCom in parallel, where in the i-th instance $(i \in [n])$, P commits to v_i.

Decommit Stage:

1. P sends $\{\mathsf{v}_i\}_{i\in[n]}$ together with the corresponding decommitment information w.r.t. the ExtCom in Step 2 of the Commit Stage.
2. V checks that all the decommitments in Step 1 of the Decommit Stage are valid. If so, V outputs $\mathsf{VSS}_{\mathsf{Recon}}(\mathsf{v}_1, \ldots, \mathsf{v}_n)$ and then halts; otherwise, V outputs $\perp$ and then halts.

Prove Stage: both parties learn a polynomial-time computable predicate ϕ.

1. P emulates "in his head" n (virtual) players $\{P_i\}_{i\in[n]}$, where P_i's input is v_i (from Step 1 of the Commit Stage). These n parties execute Π_{MPC} for the following functionality: the functionality reconstructs $m' := \mathsf{VSS}_{\mathsf{Recon}}(\mathsf{v}_1, \ldots, \mathsf{v}_n)$ and sends the value $\phi(m')$ to all the parties as their output. For $i \in [n]$, let v'_i be the view of party P_i during Π_{MPC}.
2. P and V involve in n executions of Com in parallel, where in the i-th instance $(i \in [n])$, P commits to v'_i.
3. V picks a random string r_1 and commits to it using ExtCom.
4. P picks a random string r_2 and sends it to V.
5. V sends to P the value r_1 together with the corresponding decommitment information w.r.t. the ExtCom in Step 3. Now, both parties learn a coin-tossing result $r = r_1 \oplus r_2$, which specifies a size-t random subset $T \subseteq [n]$.

6. P sends to V in *one round* the following messages:
   (a) $\{\mathsf{v}_i\}_{i \in T}$ together with the corresponding decommitment information w.r.t. the ExtCom in Step 2 of the Commit Stage; **and**
   (b) $\{\mathsf{v}'_i\}_{i \in T}$ together with the corresponding decommitment information w.r.t. the Com in Step 2 of the Prove Stage.
7. V checks the following conditions:
   (a) All the decommitments in Steps 6a and 6b are valid; **and**
   (b) for any $i \in T$, v_i is the prefix of v'_i ; **and**
   (c) for any $i, j \in T$, views $(\mathsf{v}'_i, \mathsf{v}'_j)$ are consistent w.r.t. the $\mathsf{VSS}_{\mathsf{Share}}$ execution in Step 1 of the Commit Stage and the Π_{MPC} execution as described in Step 1 of the Prove Stage.

   If all the checks pass, V accepts; otherwise, V rejects.

Security. It is straightforward to see that Prot. 3 is constant-round and makes only black-box access to OWFs. Completeness follows from that of VSS, ExtCom, Com, and Π_{MPC}. In the following, we show ε-simulatable extractability (in Lem. 10), soundness (in Lem. 11), and ε-zero-knowledge (in Lem. 12). Due to space constraints, we postpone their proofs to [16, Section 6.5].

Lemma 10 (ε-Simulation Extractability). *Assume* ExtCom *is parallel-strongly extractable with ε-simulation. Then, Prot. 3 satisfies security as ε-simulation extractable commitment.*

Lemma 11 (Soundness). *Assume* ExtCom *and* Com *are statistically binding,* ExtCom *is computationally-hiding,* VSS *is $(n{+}1, t)$-perfectly verifiable-committing and Π_{MPC} is (n, t)-perfectly robust. Then, Prot. 3 satisfies the soundness requirement (see [16, Definition 17]).*

Lemma 12 (ε-Zero-Knowledge). *Assume* ExtCom *and* Com *are computationally-hiding,* ExtCom *is weakly extractable with ε-simulation,* VSS *is $(n + 1, t)$-secret (see [16, Definition 1]), and Π_{MPC} is (n, t)-semi-honest computationally private (see [16, Definition 4]). Then, Prot. 3 satisfies the ε-zero-knowledge property defined in [16, Definition 17].*

Applications. Applications of our ε-simulatable ExtCom-and-Prove protocol are postponed to the full version, where we will show to to obtain ε-simulatable coin-flipping [16, Section 6.2], zero-knowledge argument of knowledge with an ε-simulatable knowledge extractor [16, Section 6.3], and black-box ε-zero-knowledge for **QMA** [16, Section 6.4].

7 Black-Box ε-Simulatable PQ-2PC in Constant Rounds

The ε-simulatable ExtCom-and-Prove protocol constructed in Sec. 6 yields the following theorems. Due to space constraints, their proofs are postponed to the full version [16, Sections 7 and 8].

Theorem 3. *Assuming the existence of a constant-round semi-honest bit-OT secure against QPT adversaries, there exists a black-box, constant-round construction of ε-simulatable 2PC protocol secure against QPT adversaries.*

Theorem 4. *Assuming the existence of OWFs secure against QPT adversaries, there exists a black-box, constant-round construction of ε-simulatable 2PC protocol secure against QPT adversaries. This protocol makes use of quantum communication.*

8 Acknowledgments

Kai-Min Chung is supported in part by Ministry of Science and Technology, Taiwan, under Grant no. MOST 109-2223-E-001-001-MY3, the 2021 Academia Sinica Investigator Award (AS-IA-110-M02), and the Air Force Office of Scientific Research under award number FA2386-20-1-4066.

Xiao Liang is supported in part by Omkant Pandey's DARPA SIEVE Award HR00112020026 and NSF grants 1907908 and 2028920. 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, DARPA, or NSF.

References

1. Aaronson, S.: Limitations of quantum advice and one-way communication. Theory of Computing **1**(1), 1–28 (2005)
2. Agarwal, A., Bartusek, J., Goyal, V., Khurana, D., Malavolta, G.: Post-quantum multi-party computation. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 435–464. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77870-5_16
3. Ananth, P., Chung, K.M., Placa, R.L.L.: On the concurrent composition of quantum zero-knowledge. In: Annual International Cryptology Conference. pp. 346–374. Springer (2021)
4. Ananth, P., La Placa, R.L.: Secure quantum extraction protocols. In: Pass, R., Pietrzak, K. (eds.) TCC 2020, Part III. LNCS, vol. 12552, pp. 123–152. Springer, Heidelberg (Nov 2020). https://doi.org/10.1007/978-3-030-64381-2_5
5. Bartusek, J., Coladangelo, A., Khurana, D., Ma, F.: One-way functions imply secure computation in a quantum world. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part I. LNCS, vol. 12825, pp. 467–496. Springer, Heidelberg, Virtual Event (Aug 2021). https://doi.org/10.1007/978-3-030-84242-0_17
6. Ben-Or, M., Goldwasser, S., Wigderson, A.: Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In: 20th ACM STOC. pp. 1–10. ACM Press (May 1988). https://doi.org/10.1145/62212.62213
7. Bennett, C.H., Brassard, G., Crépeau, C., Skubiszewska, M.H.: Practical quantum oblivious transfer. In: Feigenbaum, J. (ed.) CRYPTO'91. LNCS, vol. 576, pp. 351–366. Springer, Heidelberg (Aug 1992). https://doi.org/10.1007/3-540-46766-1_29

8. Bitansky, N., Kalai, Y.T., Paneth, O.: Multi-collision resistance: a paradigm for keyless hash functions. In: Diakonikolas, I., Kempe, D., Henzinger, M. (eds.) 50th ACM STOC. pp. 671–684. ACM Press (Jun 2018). https://doi.org/10.1145/3188745.3188870
9. Bitansky, N., Lin, H., Shmueli, O.: Non-malleable commitments against quantum attacks. Cryptology ePrint Archive, Report 2021/920 (2021), https://ia.cr/2021/920
10. Bitansky, N., Shmueli, O.: Post-quantum zero knowledge in constant rounds. In: Makarychev, K., Makarychev, Y., Tulsiani, M., Kamath, G., Chuzhoy, J. (eds.) 52nd ACM STOC. pp. 269–279. ACM Press (Jun 2020). https://doi.org/10.1145/3357713.3384324
11. Bouman, N.J., Fehr, S.: Sampling in a quantum population, and applications. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 724–741. Springer, Heidelberg (Aug 2010). https://doi.org/10.1007/978-3-642-14623-7_39
12. Brakerski, Z., Yuen, H.: Quantum garbled circuits (2020)
13. Broadbent, A., Grilo, A.B.: QMA-hardness of consistency of local density matrices with applications to quantum zero-knowledge. In: 61st FOCS. pp. 196–205. IEEE Computer Society Press (Nov 2020). https://doi.org/10.1109/FOCS46700.2020.00027
14. Broadbent, A., Ji, Z., Song, F., Watrous, J.: Zero-knowledge proof systems for QMA. SIAM J. Comput. **49**(2), 245–283 (2020)
15. Chatterjee, R., Liang, X., Pandey, O.: Improved black-box constructions of composable secure computation. In: Czumaj, A., Dawar, A., Merelli, E. (eds.) ICALP 2020. LIPIcs, vol. 168, pp. 28:1–28:20. Schloss Dagstuhl (Jul 2020). https://doi.org/10.4230/LIPIcs.ICALP.2020.28
16. Chia, N.H., Chung, K.M., Liang, X., Yamakawa, T.: Post-quantum simulatable extraction with minimal assumptions: Black-box and constant-round. Cryptology ePrint Archive, Paper 2021/1516 (2021), https://eprint.iacr.org/2021/1516, https://eprint.iacr.org/2021/1516
17. Chia, N.H., Chung, K.M., Liu, Q., Yamakawa, T.: On the impossibility of post-quantum black-box zero-knowledge in constant rounds. In: 62nd FOCS (2021)
18. Chia, N.H., Chung, K.M., Yamakawa, T.: A black-box approach to post-quantum zero-knowledge in constant rounds. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part I. LNCS, vol. 12825, pp. 315–345. Springer, Heidelberg, Virtual Event (Aug 2021). https://doi.org/10.1007/978-3-030-84242-0_12
19. Choi, S.G., Dachman-Soled, D., Malkin, T., Wee, H.: Simple, black-box constructions of adaptively secure protocols. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 387–402. Springer, Heidelberg (Mar 2009). https://doi.org/10.1007/978-3-642-00457-5_23
20. Cramer, R., Damgård, I., Dziembowski, S., Hirt, M., Rabin, T.: Efficient multiparty computations secure against an adaptive adversary. In: Stern, J. (ed.) EUROCRYPT'99. LNCS, vol. 1592, pp. 311–326. Springer, Heidelberg (May 1999). https://doi.org/10.1007/3-540-48910-X_22
21. Crépeau, C., Kilian, J.: Achieving oblivious transfer using weakened security assumptions (extended abstract). In: 29th FOCS. pp. 42–52. IEEE Computer Society Press (Oct 1988). https://doi.org/10.1109/SFCS.1988.21920
22. Crépeau, C., Kilian, J.: Weakening security assumptions and oblivious transfer (abstract). In: Goldwasser, S. (ed.) CRYPTO'88. LNCS, vol. 403, pp. 2–7. Springer, Heidelberg (Aug 1990). https://doi.org/10.1007/0-387-34799-2_1

23. Damgård, I., Fehr, S., Lunemann, C., Salvail, L., Schaffner, C.: Improving the security of quantum protocols via commit-and-open. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 408–427. Springer, Heidelberg (Aug 2009). https://doi.org/10.1007/978-3-642-03356-8_24
24. Damgård, I., Ishai, Y.: Constant-round multiparty computation using a black-box pseudorandom generator. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 378–394. Springer, Heidelberg (Aug 2005). https://doi.org/10.1007/11535218_23
25. Dolev, D., Dwork, C., Naor, M.: Nonmalleable cryptography. SIAM J. Comput. **30**(2), 391–437 (2000)
26. Dulek, Y., Grilo, A.B., Jeffery, S., Majenz, C., Schaffner, C.: Secure multi-party quantum computation with a dishonest majority. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 729–758. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45727-3_25
27. Dwork, C., Naor, M., Sahai, A.: Concurrent zero-knowledge. J. ACM **51**(6), 851–898 (2004)
28. Fleischhacker, N., Goyal, V., Jain, A.: On the existence of three round zero-knowledge proofs. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part III. LNCS, vol. 10822, pp. 3–33. Springer, Heidelberg (Apr / May 2018). https://doi.org/10.1007/978-3-319-78372-7_1
29. Garg, R., Khurana, D., Lu, G., Waters, B.: Black-box non-interactive non-malleable commitments. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part III. LNCS, vol. 12698, pp. 159–185. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77883-5_6
30. Garg, S., Goyal, V., Jain, A., Sahai, A.: Concurrently secure computation in constant rounds. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 99–116. Springer, Heidelberg (Apr 2012). https://doi.org/10.1007/978-3-642-29011-4_8
31. Garg, S., Gupta, D., Miao, P., Pandey, O.: Secure multiparty RAM computation in constant rounds. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part I. LNCS, vol. 9985, pp. 491–520. Springer, Heidelberg (Oct / Nov 2016). https://doi.org/10.1007/978-3-662-53641-4_19
32. Garg, S., Kiyoshima, S., Pandey, O.: A new approach to black-box concurrent secure computation. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part II. LNCS, vol. 10821, pp. 566–599. Springer, Heidelberg (Apr / May 2018). https://doi.org/10.1007/978-3-319-78375-8_19
33. Garg, S., Liang, X., Pandey, O., Visconti, I.: Black-box constructions of bounded-concurrent secure computation. In: Galdi, C., Kolesnikov, V. (eds.) SCN 20. LNCS, vol. 12238, pp. 87–107. Springer, Heidelberg (Sep 2020). https://doi.org/10.1007/978-3-030-57990-6_5
34. Gennaro, R., Lysyanskaya, A., Malkin, T., Micali, S., Rabin, T.: Algorithmic tamper-proof (ATP) security: Theoretical foundations for security against hardware tampering. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 258–277. Springer, Heidelberg (Feb 2004). https://doi.org/10.1007/978-3-540-24638-1_15
35. Goldreich, O., Kahan, A.: How to construct constant-round zero-knowledge proof systems for NP. Journal of Cryptology **9**(3), 167–190 (Jun 1996)
36. Goyal, V.: Constant round non-malleable protocols using one way functions. In: Fortnow, L., Vadhan, S.P. (eds.) 43rd ACM STOC. pp. 695–704. ACM Press (Jun 2011). https://doi.org/10.1145/1993636.1993729
37. Goyal, V., Lee, C.K., Ostrovsky, R., Visconti, I.: Constructing non-malleable commitments: A black-box approach. In: 53rd FOCS. pp. 51–60. IEEE Computer Society Press (Oct 2012). https://doi.org/10.1109/FOCS.2012.47

38. Goyal, V., Lin, H., Pandey, O., Pass, R., Sahai, A.: Round-efficient concurrently composable secure computation via a robust extraction lemma. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part I. LNCS, vol. 9014, pp. 260–289. Springer, Heidelberg (Mar 2015). https://doi.org/10.1007/978-3-662-46494-6_12
39. Goyal, V., Ostrovsky, R., Scafuro, A., Visconti, I.: Black-box non-black-box zero knowledge. In: Shmoys, D.B. (ed.) 46th ACM STOC. pp. 515–524. ACM Press (May / Jun 2014). https://doi.org/10.1145/2591796.2591879
40. Grilo, A.B., Lin, H., Song, F., Vaikuntanathan, V.: Oblivious transfer is in MiniQCrypt. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part II. LNCS, vol. 12697, pp. 531–561. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77886-6_18
41. Haitner, I.: Semi-honest to malicious oblivious transfer - the black-box way. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 412–426. Springer, Heidelberg (Mar 2008). https://doi.org/10.1007/978-3-540-78524-8_23
42. Haitner, I., Ishai, Y., Kushilevitz, E., Lindell, Y., Petrank, E.: Black-box constructions of protocols for secure computation. SIAM J. Comput. **40**(2), 225–266 (2011). https://doi.org/10.1137/100790537, `https://doi.org/10.1137/100790537`
43. Hallgren, S., Smith, A., Song, F.: Classical cryptographic protocols in a quantum world. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 411–428. Springer, Heidelberg (Aug 2011). https://doi.org/10.1007/978-3-642-22792-9_23
44. Hazay, C., Venkitasubramaniam, M.: On the power of secure two-party computation. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part II. LNCS, vol. 9815, pp. 397–429. Springer, Heidelberg (Aug 2016). https://doi.org/10.1007/978-3-662-53008-5_14
45. Hazay, C., Venkitasubramaniam, M.: Round-optimal fully black-box zero-knowledge arguments from one-way permutations. In: Beimel, A., Dziembowski, S. (eds.) TCC 2018, Part I. LNCS, vol. 11239, pp. 263–285. Springer, Heidelberg (Nov 2018). https://doi.org/10.1007/978-3-030-03807-6_10
46. Ishai, Y., Kushilevitz, E., Lindell, Y., Petrank, E.: Black-box constructions for secure computation. In: Kleinberg, J.M. (ed.) 38th ACM STOC. pp. 99–108. ACM Press (May 2006). https://doi.org/10.1145/1132516.1132531
47. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Zero-knowledge from secure multiparty computation. In: Johnson, D.S., Feige, U. (eds.) 39th ACM STOC. pp. 21–30. ACM Press (Jun 2007). https://doi.org/10.1145/1250790.1250794
48. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Cryptography with constant computational overhead. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC. pp. 433–442. ACM Press (May 2008). https://doi.org/10.1145/1374376.1374438
49. Ishai, Y., Prabhakaran, M., Sahai, A.: Founding cryptography on oblivious transfer - efficiently. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 572–591. Springer, Heidelberg (Aug 2008). https://doi.org/10.1007/978-3-540-85174-5_32
50. Khurana, D., Ostrovsky, R., Srinivasan, A.: Round optimal black-box "commit-and-prove". In: Beimel, A., Dziembowski, S. (eds.) TCC 2018, Part I. LNCS, vol. 11239, pp. 286–313. Springer, Heidelberg (Nov 2018). https://doi.org/10.1007/978-3-030-03807-6_11
51. Kilian, J.: Founding cryptography on oblivious transfer. In: 20th ACM STOC. pp. 20–31. ACM Press (May 1988). https://doi.org/10.1145/62212.62215
52. Kiyoshima, S.: Round-efficient black-box construction of composable multi-party computation. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part II. LNCS, vol. 8617, pp. 351–368. Springer, Heidelberg (Aug 2014). https://doi.org/10.1007/978-3-662-44381-1_20

53. Kiyoshima, S.: Round-optimal black-box commit-and-prove with succinct communication. In: Micciancio, D., Ristenpart, T. (eds.) CRYPTO 2020, Part II. LNCS, vol. 12171, pp. 533–561. Springer, Heidelberg (Aug 2020). https://doi.org/10.1007/978-3-030-56880-1_19
54. Liang, X., Pandey, O.: Towards a unified approach to black-box constructions of zero-knowledge proofs. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part IV. LNCS, vol. 12828, pp. 34–64. Springer, Heidelberg, Virtual Event (Aug 2021). https://doi.org/10.1007/978-3-030-84259-8_2
55. Lin, H., Pass, R.: Black-box constructions of composable protocols without set-up. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 461–478. Springer, Heidelberg (Aug 2012). https://doi.org/10.1007/978-3-642-32009-5_27
56. Lindell, Y.: A note on constant-round zero-knowledge proofs of knowledge. Journal of Cryptology **26**(4), 638–654 (Oct 2013). https://doi.org/10.1007/s00145-012-9132-7
57. Lombardi, A., Ma, F., Spooner, N.: Post-quantum zero knowledge, revisited (or: How to do quantum rewinding undetectably). Cryptology ePrint Archive, Report 2021/1543 (2021), `https://ia.cr/2021/1543`
58. Lunemann, C., Nielsen, J.B.: Fully simulatable quantum-secure coin-flipping and applications. In: Nitaj, A., Pointcheval, D. (eds.) AFRICACRYPT 11. LNCS, vol. 6737, pp. 21–40. Springer, Heidelberg (Jul 2011)
59. Micciancio, D., Ong, S.J., Sahai, A., Vadhan, S.P.: Concurrent zero knowledge without complexity assumptions. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 1–20. Springer, Heidelberg (Mar 2006). https://doi.org/10.1007/11681878_1
60. Naor, M.: Bit commitment using pseudorandomness. Journal of Cryptology **4**(2), 151–158 (Jan 1991). https://doi.org/10.1007/BF00196774
61. Pass, R., Wee, H.: Black-box constructions of two-party protocols from one-way functions. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 403–418. Springer, Heidelberg (Mar 2009). https://doi.org/10.1007/978-3-642-00457-5_24
62. Prabhakaran, M., Rosen, A., Sahai, A.: Concurrent zero knowledge with logarithmic round-complexity. In: 43rd FOCS. pp. 366–375. IEEE Computer Society Press (Nov 2002). https://doi.org/10.1109/SFCS.2002.1181961
63. Rosen, A.: A note on constant-round zero-knowledge proofs for NP. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 191–202. Springer, Heidelberg (Feb 2004). https://doi.org/10.1007/978-3-540-24638-1_11
64. Unruh, D.: Quantum proofs of knowledge. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 135–152. Springer, Heidelberg (Apr 2012). https://doi.org/10.1007/978-3-642-29011-4_10
65. Watrous, J.: Zero-knowledge against quantum attacks. SIAM Journal on Computing **39**(1), 25–58 (2009)
66. Wyner, A.D.: The wire-tap channel. Bell system technical journal **54**(8), 1355–1387 (1975)