A New Approach to Black-Box Concurrent Secure Computation[*]

Sanjam Garg[1**], Susumu Kiyoshima[2], Omkant Pandey[3]

[1] University of California, Berkeley, USA
sanjamg@berkeley.edu
[2] NTT Secure Platform Laboratories, Tokyo, Japan
kiyoshima.susumu@lab.ntt.co.jp
[3] Stony Brook University, Stony Brook, USA
omkant@cs.stonybrook.edu

Abstract. We consider the task of constructing concurrently composable protocols for general secure computation by making only *black-box* use of underlying cryptographic primitives. Existing approaches for this task first construct a black-box version of CCA-secure commitments which provide a strong form of concurrent security to the committed value(s). This strong form of security is then crucially used to construct higher level protocols such as concurrently secure OT/coin-tossing (and eventually all functionalities).

This work explores a fresh approach. We first aim to construct a concurrently-secure OT protocol whose concurrent security is proven directly using concurrent simulation techniques; in particular, it does not rely on the usual "non-polynomial oracles" of CCA-secure commitments. The notion of concurrent security we target is *super-polynomial simulation* (SPS). We show that such an OT protocol can be constructed from *polynomial* hardness assumptions in a *black-box* manner, and within a *constant* number of rounds. In fact, we only require the existence of (constant round) semi-honest OT and standard collision-resistant hash functions.

Next, we show that such an OT protocol is sufficient to obtain SPS-secure (concurrent) multiparty computation (MPC) for general functionalities. This transformation does not require any additional assumptions; it also maintains the black-box nature as well as the constant round feature of the original OT protocol. Prior to our work, the only known black-box construction of constant-round concurrently composable MPC required stronger assumptions; namely, verifiable perfectly binding homomorphic commitment schemes and PKE with oblivious public-key generation.

[*] This article is based on an article that appears in the proceedings of EUROCRYPT 2018.

[**] Research supported in part from AFOSR YIP Award, DARPA/ARL SAFEWARE Award W911NF15C0210, AFOSR Award FA9550-15-1-0274, and research grants by the Okawa Foundation, Visa Inc., and Center for Long-Term Cybersecurity (CLTC, UC Berkeley). The views expressed are those of the author and do not reflect the official policy or position of the funding agencies.

1 Introduction

Secure multiparty computation (MPC) protocols enable mutually distrustful parties to compute a joint functionality on their private inputs without compromising the correctness of the outputs and the privacy of their inputs. They have been studied in both two-party and multi-party cases. General constructions of such protocols for computing any functionality even when a majority of players are adversarial have been long known [51,17]. In this work, we are interested in MPC protocols that only make a black-box use of cryptographic primitives and maintain security in a concurrent environment with many simultaneous executions.

Black-box constructions. General purpose MPC protocols are often *non-black-box* in nature. They use the code of the underlying primitives at some stage of the computation, e.g., an NP reduction for general zero-knowledge proofs. Such non-black-use of the primitives is generally undesirable since not only it is computationally expensive, it also renders the protocol useless in situations where such code is not available (e.g., primitives based on hardware-tokens). One therefore seeks *black-box* constructions of such protocols which use the underlying primitives only in black-box way (i.e., only through their input/output interfaces).

Recently, a number of works have studied black-box constructions of general MPC protocols. Ishai et al. [27] presented the first black-box construction of general purpose MPC based on enhanced trapdoor permutations or homomorphic public-key encryption schemes. Combined with the subsequent work of Haitner [23] on black-box OT, this gives a black-box construction of general MPC based assuming only semi-honest OT [24]. Subsequently, Wee [50] reduced the round complexity of these constructions to $O(\log^* n)$, and Goyal [18] to only constant rounds. In the two-party setting, black-box construction were obtained by Pass and Wee [46] in constant-rounds and Ostrovsky et al. [41] in optimal 5-rounds.

Concurrent security. The standard notion of security for MPC, also called *stand-alone security* considers only a single execution of this protocol. While this is sufficient for many applications, other situations (such as protocol executions over the Internet) require stronger notions of security. Such a more demanding setting, where there may be many protocols executions at the same time, is called the *concurrent* setting. Unfortunately, it is known that stand-alone security does not necessarily imply security in the concurrent setting [13].

Secure computation in the concurrent setting is quite challenging to define. Canetti [4] proposed the notion of *universally composable* (UC) security where protocols maintain their strong simulation based security guarantees even in the presence of other arbitrary protocols. Achieving such strong notion of UC-security turned out to be impossible in the plain model [4,5]. Moreover, Lindell [35,36] proved that even in the special case where only instantiations of the *same* protocol are allowed, standard notion of polynomial-time simulation is impossible to achieve. (This is "self composition" and corresponds to the setting we are interested in.)

These strong negative results motivated the study of alternative notions of security; of these, most relevant to us are super-polynomial simulation (SPS) [43], angel-based security [48,6], and security with shielded oracles [3].

SPS Security. SPS security is similar to UC security except that the simulator is allowed to run in super-polynomial time. It guarantees that whatever an adversary can do in the real world can also be done in the ideal world *in super-polynomial time*. While SPS-security is a weaker guarantee, it is still meaningful security for many functionalities, and allows concurrent self-composition in the plain model. (In what follows, by SPS security we mean SPS-security under concurrent self-composition.) Prabhakaran and Sahai [48] provided the initial positive result for SPS security. Although, these early results [48,2,37,34] relied on non-standard/sub-exponential assumptions, Canetti, Lin and Pass achieved this notion under standard polynomial-time assumptions [6] in polynomially many rounds. Soon after, Garg et al. [15] presented a *constant round* construction. The works of [48,37,6] actually get angel-based security, discussed below.

Angel-based Security. Angel-based UC security is the same as UC security except that the environment/adversary and the simulator have access to an additional entity—an *angel*—that allows some judicious use of super-polynomial resources. Angel-based UC security, though weaker than UC security, is meaningful for many settings and implies SPS security. Furthermore, like UC security, it also guarantees composability. As noted above, the works in [48,37,6] achieve angel-based security, though only [6] relies on standard polynomial hardness. Subsequently, Goyal et al. [21] presented a $\widetilde{O}(\log n)$ round construction under the same assumptions.

Black-box constructions of angel-based secure computation were first presented by Lin and Pass [32] assuming the existence of semi-honest OT, in $O(\max(n^{\epsilon}, R_{\mathsf{OT}}))$ rounds, where $\epsilon > 0$ is an arbitrary constant and R_{OT} is the round complexity of the underlying OT protocol. (Hence, if the underlying OT protocol has only constant round, the round complexity is $O(n^{\epsilon})$.) Subsequently, Kiyoshima [29] provided a $\widetilde{O}(\log^2 n)$-round construction under the same assumption.

Security with Shielded Oracles. Security with shielded oracles, proposed very recently by Broadnax et al. [3], is similar to angel-based security where the environment and the simulator have access to an additional entity—a *shielded oracle*—that can perform some super-polynomial computation. However, unlike angel-based security, the results of super-polynomial time computation are "shielded away" from the simulator, in the sense that the shielded oracle directly interacts with the ideal functionality; the simulator cannot observe their communication. This notion lies strictly between SPS and angel-based security, and guarantees composability. A constant-round MPC protocol satisfying this notion were also presented in [3]; one of their protocol is black-box and relies on standard polynomial hardness. More specifically, it requires (verifiable perfectly binding) homomorphic commitment schemes and PKE with oblivious public-key generation.

State-of-the-art. All of the constructions of concurrently-secure MPC protocols we have discussed so far, rely on first constructing non-malleable commitment schemes with strong concurrent or UC-security properties; in particular (robust) "CCA-secure commitments" or "coin tossing" or UC-secure commitments. These schemes are then used to build higher level protocols such as OT and general secure computation. However, the concurrent security of these higher level protocols is proven indirectly, by relying on the strong concurrent security of the CCA-secure commitments. While this approach leads to (better) angel-based security, it is quite expensive in terms of rounds, requiring $\tilde{O}(\log^2 n)$ in [29] for *black-box* constructions. The work of Broadnax et al. [3] significantly improves this situation by relaxing the angel-based security requirement to SPS with shielded-oracles, and obtains a constant round construction. However, it still needs stronger assumptions (see above) and represents the only approach so far for obtaining constant round black-box constructions. In contrast, much of the results that make non-black-box use of the primitives, can rely on the minimal assumption of semi-honest OT. The approach of Broadnax et al. [3] is still based on first constructing a sufficiently strong commitment scheme with UC properties and using it to obtain OT and general functionalities. It is highly desirable to find new approaches to construct such protocols which have the potential to rely on minimal assumptions in constant rounds.

1.1 Our Contribution

In this work, we seek new approaches for constructing concurrently-secure black-box MPC protocols which can lead to qualitative improvements over existing constructions, such as minimal underlying assumptions, a constant number of rounds, and so on. Towards this goal, we deviate from the existing approaches which focus on incorporating both *concurrent security* and *non-malleability* into a single primitive such as (CCA-secure) commitment schemes or coin-tossing. Instead, we take a different approach and focus on incorporating concurrent security into the oblivious transfer functionality. We present a black-box OT protocol satisfying the SPS notion of concurrent-security. We achieve this by using concurrent simulation techniques and non-malleable commitments in a somewhat modular way where (roughly speaking) the former is primarily used for trapdoor extraction/simulation and the latter for independence of committed values. The protocol has constant rounds and relies only on the existence of (constant round) semi-honest OT and standard collision-resistant hash functions (CRHFs).

Having obtained concurrent security for OT, we proceed to construct SPS-secure MPC protocols for all functionalities. Our method does not require any additional assumptions, and maintains the black-box and constant round properties of the original OT protocol. Consequently, we obtain SPS-secure constant-round black-box MPC under much weaker assumptions than [3]. On the flip side, our work achieves a weaker security notion than [3].

Theorem 1 (Informal). *Assume the existence of constant-round semi-honest oblivious transfer protocols and collision-resistant hash functions. Then, there exists a constant-round black-box construction of concurrently secure MPC protocol that achieve SPS security.*

The formal statement is given as Theorem 3 in Section 5.

1.2 Other Related Work

Other than the works mentioned above, there are several works that study SPS security/angel-based security. For SPS-security, Pass et al. [45] present a constant-round non-black-box construction of MPC from constant-round semi-honest OT. Dachman-Soled et al. and Venkitasubramaniam [11,49] present a non-black-box construction that satisfies adaptive security. And very recently, Badrinarayanan et al. [1] present a non-black-box 3-round construction assuming sub-exponential hardness assumptions. For angel-based security, Kiyoshima et al. [30] present a constant-round *black-box* construction albeit under a sub-exponential hardness assumption, and Hazay and Venkitasubramaniam [26] present a non-black-box construction that achieves adaptive security.

We have not discussed several works that focus on other notions of concurrent security such as input-indistinguishable computation, bounded concurrent composition, and multiple ideal-query model [44,38,19].

Black-box constructions have been extensively explored for several other primitives such as non-malleable/CCA-secure encryption, non-malleable commitments, zero-knowledge proofs and so on (e.g., [10,47,9,20,22,42]). For concurrent OT, Garay and MacKenzie [14] presented a protocols for independent inputs under the DDH assumption, and Garg et al. [16] showed the impossibility of this task for general input distributions.

2 Overview of Our Techniques

We obtain our MPC protocol in two steps. First, we construct a constant-round black-box construction of a SPS-secure concurrent OT protocol. Second, we compose this OT protocol with an existing constant-round OT-hybrid UC-secure MPC protocol. We elaborate on each step below.

We remark that we consider concurrent security in the interchangeable-roles setting. So, in the case of OT, the adversary can participate a session as the sender while concurrently participating another session as the receiver.

2.1 Constant-round Black-box Concurrent OT

Our starting point is the (super-constant-round) black-box concurrent OT protocol of Lin and Pass [32], which is secure under angel-based security and makes only black-box use of semi-honest OT protocols. Our approach is to modify their protocol so that it has only constant number of rounds (while degrading security from angel-based security to SPS security).

Let us first recall the OT protocol of [32]. At a high level, it uses a semi-honest OT protocol in the black-box way in a similar manner to the stand-alone black-box OT of Haitner et al. [24] does. Specifically, the OT protocol of [32] proceeds roughly as follows.

1. First, the sender S and the receiver R execute many instances of a semi-honest OT protocol in parallel, where in each instance S and R use the inputs and the randomness that are generated by a coin-tossing protocol. (S and R execute two instances of coin tossing for each instance of OT; the sender obtains random coin in the first coin tossing and the receiver obtains random coin in the second coin tossing.)
2. Next, S and R do a simple trick called *OT combiner*, which allows them to execute an OT with their real inputs securely when most of the OT instances in the previous step are correctly executed. To check that most of the OT instances in the previous step were indeed correctly executed, S and R do the well-known *cut-and-choose* trick, where S (resp., R) chooses a constant fraction of the OT instances randomly and R (resp., S) reveals the input and randomness that it used in those instances so that S (resp., R) can verity whether R executed those instances correctly.

(Actually, the underlying OT protocol is required to be secure against malicious senders, but we ignore this requirement in this overview.)

The OT protocol of [32] has more than constant number of rounds because it uses *CCA-secure commitment schemes* [6,7] in the coin-tossing part of the protocol and existing constructions of CCA-secure commitment schemes have more than constant number of rounds under standard assumptions.[4] Key observations by the authors of [32] are that CCA-secure commitment schemes can be used to obtain a "concurrently secure" coin tossing protocol,[5] and that their OT protocol is concurrently secure when its coin-tossing part is concurrently secure.

To obtain a constant-round protocol, we need to remove the CCA-secure commitments from the protocol of [32]. A naive approach is to simply replace the CCA-secure commitments with *(concurrent) non-malleable commitments*, which provide weaker security than CCA-secure ones but are known to have a constant-round black-box instantiation under the existence of one-way functions [20]. However, this approach does not work because, as mentioned by Lin and Pass [32], non-malleable commitment schemes only lead to "parallel secure"

[4] Roughly speaking, CCA-secure commitment schemes guarantee that the hiding property holds even when the adversary has access to the *committed-value oracle*, which computes the committed value of a given commitment by brute force.

[5] Concretely, the resultant coin-tossing protocol satisfies *simulation soundness*, which guarantees that any concurrent man-in-the-middle adversary cannot bias the outcome of a coin-tossing when it concurrently participates in simulated coin tossings.

coin tossing protocols[6] and the parallel security of the coin tossing protocol is insufficient for proving the concurrent security of the OT protocol of [32].

At a high level, we remove the CCA-secure commitments from the protocol of [32] as follows. Our starting idea is to prove the concurrent security of the OT protocol of [32] without relying on the concurrent security of the coin tossing part (and therefore without using CCA-secure commitments there). To prove the concurrent security in this way, we modify the protocol of [32] so that it uses non-malleable commitments in a similar manner to the constant-round (non-black-box) SPS-secure concurrent MPC protocol of Garg et al. [15] does. Informally speaking, the protocol of Garg et al. [15] uses non-malleable commitments when each party commits to a witness for the fact that the "trapdoor statement" is false, where the trapdoor statement is a statement about the transcript and it is guaranteed that any adversary cannot "cheat" in the protocol when the trapdoor statement is false. With this use of non-malleable commitments, the concurrent security of the protocol of Garg et al. [15] is proven in two steps:

1. First, it is shown that in the real experiment (where an adversary interacts with honest parties in multiple sessions of the protocol concurrently), the non-malleable commitment from the adversary in each session is indeed a commitment of a valid witness for the fact that the trapdoor statement is false. (This is guaranteed by a zero-knowledge proof in the protocol).
2. Second, it is shown that if the non-malleable commitment in a session is indeed a commitment of a valid witness (which implies that the trapdoor statement is false in that session, which in turn implies that the adversary cannot "cheat" in that session), it is possible to switch the honest parties in that session with the simulator in an indistinguishable way, and furthermore this switch does not affect the non-malleable commitments in the other sessions (i.e., their committed values remain to be valid witnesses). (The latter is guaranteed by non-malleability of the non-malleable commitments[7].)
3. Now, the concurrent security follows from the above two since the honest parties can be switched to the simulator in all the sessions by repeatedly using what is shown in the second part.

Following this approach by Garg et al. [15], we first identify the trapdoor statement of the OT protocol of Lin and Pass [32] and then add non-malleable commitments to their protocol in such a way that the trapdoor statement is false whenever the committed values of the non-malleable commitments satisfy a specific condition. With this modification, we can prove the concurrent security of the OT protocol of [32] without relying on the concurrent security of coin tossing by following the approach of [15] outlined above.

[6] Very roughly speaking, this is because non-malleability allows the man-in-the adversary to obtain replies from the committed-value oracle only in parallel.

[7] Actually, *non-malleability w.r.t. other protocols* [31] is also required, where non-malleability w.r.t. a protocol Π guarantees non-malleability against man-in-the-middle adversaries that participates in the non-malleable commitment in the right interaction and Π in the left interaction.

Remark 1. It is not straightforward to use the approach of Garg et al. [15] in the OT protocol of Lin and Pass [32] since its trapdoor statement does not have a simple witness for the fact that the statement is false. Because of this difficulty, we do not use non-malleable commitments to commit to a witness; rather, we use them in such a way that there exists a condition on the committed values of the non-malleable commitments such that the trapdoor statement is false as long as this condition holds. For details, see Section 4 (in particular, Definitions 5 and 6).

2.2 Composition of OT with OT-hybrid MPC

We next compose our OT protocol with a OT-hybrid UC-secure MPC protocol (i.e., replace each invocation of the ideal OT functionality in the latter with an execution of the former), thereby obtaining a MPC protocol in the plain model. A problem is that the security of the resultant MPC protocol cannot be derived trivially from those of the components since SPS security does not guarantee composability. Hence, we prove the security by analyzing the MPC protocol directly. In essence, what we do is to observe that the security proof for our OT protocol (which consists of a hybrid argument from the real world to the ideal world) still works even after the OT protocol is composed with a OT-hybrid MPC protocol, and in particular we observe that the condition on the committed values of the non-malleable commitments (which is mentioned in Section 2.1) remains to hold in each session even after switching the OT-hybrid MPC protocol in any session to simulation. Fortunately, this can be observed easily thanks to the non-malleability of the non-malleable commitments, so we can prove the concurrent security of our MPC protocol under SPS security easily.

3 Preliminaries

We denote the security parameter by n. We assume familiarity with basic cryptographic protocols (e.g., commitment schemes and oblivious transfer protocols). Some basic notions, terminologies, and definitions (about secret sharing schemes, commitment schemes, and extractable commitment schemes) are given in Appendix A.

3.1 Non-malleable Commitment Schemes.

We recall the definition of non-malleable commitment schemes from [31]. Let $\langle C, R\rangle$ be a tag-based commitment scheme (i.e., a commitment scheme that takes a n-bit string—a *tag*—as an additional input). For any man-in-the-middle adversary $\mathcal{M}$, consider the following experiment. On input security parameter 1^n and auxiliary input $z \in \{0,1\}^*$, $\mathcal{M}$ participates in one left and one right interactions simultaneously. In the left interaction, $\mathcal{M}$ interacts with the committer of $\langle C, R\rangle$ and receives a commitment to value v using identity $\mathsf{id} \in \{0,1\}^n$ of its choice. In the right interaction, $\mathcal{M}$ interacts with the receiver of $\langle C, R\rangle$ and gives a commitment using identity $\widetilde{\mathsf{id}}$ of its choice. Let $\widetilde{v}$ be the value that $\mathcal{M}$ commits to on the right. If the right commitment is invalid or undefined, $\widetilde{v}$ is defined to be $\perp$. If $\mathsf{id} = \widetilde{\mathsf{id}}$, value $\widetilde{v}$ is also defined to be $\perp$. Let $\mathsf{mim}(\langle C, R\rangle, \mathcal{M}, v, z)$ be a random variable representing $\widetilde{v}$ and the view of $\mathcal{M}$ in the above experiment.

Definition 1. *A commitment scheme $\langle C, R\rangle$ is* ***non-malleable*** *if for any* PPT *adversary $\mathcal{M}$, the following are computationally indistinguishable.*

- $\{\mathsf{mim}(\langle C, R\rangle, \mathcal{M}, v, z)\}_{n\in\mathbb{N}, v\in\{0,1\}^n, v'\in\{0,1\}^n, z\in\{0,1\}^*}$
- $\{\mathsf{mim}(\langle C, R\rangle, \mathcal{M}, v', z)\}_{n\in\mathbb{N}, v\in\{0,1\}^n, v'\in\{0,1\}^n, z\in\{0,1\}^*}$

The above definition can be generalized naturally so that the adversary gives multiple commitments *in parallel* in the right interaction. The non-malleability in this generalized setting is called *parallel non-malleability.* (It is known that this “one-many” definition implies the “many-many” one, where the adversary receives multiple commitments in the left session [33].)

Robust non-malleability. We next recall the definition of k-robust non-malleability (a.k.a. non-malleability w.r.t. k-round protocols) [31]. Consider a man-in-the-middle adversary $\mathcal{M}$ that participates in one left interaction—communicating with a machine B—and one right interaction—communicating with a receiver a commitment scheme $\langle C, R\rangle$. As in the standard definition of non-malleability, $\mathcal{M}$ can choose the identity in the right interaction. We denote by $\mathsf{mim}^{B,\mathcal{M}}_{\langle C,R\rangle}(y, z)$ the random variable consisting of the view of $\mathcal{M}(z)$ in a man-in-the-middle execution when communicating with $B(y)$ on the left and an honest receiver on the right, combined with the value $\mathcal{M}(z)$ commits to on the right. Intuitively, $\langle C, R\rangle$ is non-malleable w.r.t. B if $\mathsf{mim}^{B,\mathcal{M}}_{\langle C,R\rangle}(y_1, z)$ and $\mathsf{mim}^{B,\mathcal{M}}_{\langle C,R\rangle}(y_2, z)$ are indistinguishable whenever interactions with $B(y_1)$ and $B(y_2)$ are indistinguishable.

Definition 2. *Let $\langle C, R\rangle$ be a commitment scheme and B be a* PPT *ITM. We say that a commitment scheme $\langle C, R\rangle$ is* ***non-malleable w.r.t.*** *B if the following holds: For every two sequences $\{y^1_n\}_{n\in\mathbb{N}}$ and $\{y^2_n\}_{n\in\mathbb{N}}$ such that $y^1_n, y^2_n \in \{0,1\}^n$, if it holds that for any* PPT *ITM $\mathcal{A}$,*

$$\left\{\langle B(y^1_n), \mathcal{A}(z)\rangle(1^n)\right\}_{n\in\mathbb{N}, z\in\{0,1\}^*} \approx \left\{\langle B(y^2_n), \mathcal{A}(z)\rangle(1^n)\right\}_{n\in\mathbb{N}, z\in\{0,1\}^*} \quad ,$$

it also holds that for any PPT *man-in-the-middle adversary $\mathcal{M}$,*

$$\left\{\mathsf{mim}^{B,\mathcal{M}}_{\langle C,R\rangle}(y_1, z)\right\}_{n\in\mathbb{N}, z\in\{0,1\}^*} \approx \left\{\mathsf{mim}^{B,\mathcal{M}}_{\langle C,R\rangle}(y_2, z)\right\}_{n\in\mathbb{N}, z\in\{0,1\}^*} \quad .$$

$\langle C, R\rangle$ is *k-robust* if $\langle C, R\rangle$ is non-malleable w.r.t. any machine that interacts with the adversary in k rounds. We define parallel k-robustness naturally.

Black-box instantiation. There exists a constant-round black-box construction of a parallel (actually, concurrent) non-malleable commitment scheme based on one-way functions [20]. In Appendix B, we show that any parallel non-malleable commitment can be transformed into a parallel k-robust non-malleable one in the black-box way by using collision-resistant hash functions (more precisely, by using statistically hiding commitment schemes, which can be constructed from collision-resistant hash functions). If k is constant, the round complexity increases only by a constant factor in this transformation.

3.2 UC Security and Its SPS Variant

We next recall the definition of UC security [4] and its SPS variant [48,2,15]. A part of the text below is taken from [15].

UC Security. We assume that the readers are familiar with the UC framework. A brief overview is given in Appendix C. For full details, see [4].

Recall that in the UC framework, the model for protocol execution consists of the environment $\mathcal{Z}$, the adversary $\mathcal{A}$, and the parties running protocol π. In this paper, we consider static adversaries and assume the existence of authenticated communication channels. Let $\mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}(n,z)$ denote a random variable for the output of $\mathcal{Z}$ on security parameter $n \in \mathbb{N}$ and input $z \in \{0,1\}^*$ with a uniformly chosen random tape. Let $\mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}$ denote the ensemble $\{\mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}(n,z)\}_{n\in\mathbb{N},z\in\{0,1\}^*}$.

The security of a protocol π is defined using the *ideal protocol*. In the ideal protocol, all the parties simply hand their inputs to the *ideal functionality* $\mathcal{F}$, which carries out the desired task securely and gives outputs to the parties; the parties then forward these outputs to $\mathcal{Z}$. The adversary $\mathcal{S}im$ in the execution of the ideal protocol is often called the *simulator*. Let $\pi(\mathcal{F})$ denote the ideal protocol for functionality $\mathcal{F}$.

We say that a protocol π *emulates* protocol ϕ if for any adversary $\mathcal{A}$ there exists an adversary $\mathcal{S}im$ such that no environment $\mathcal{Z}$, on any input, can tell whether it is interacting with $\mathcal{A}$ and parties running π or it is interacting with $\mathcal{S}im$ and parties running ϕ. We say that π *securely realizes* an ideal functionality $\mathcal{F}$ if it emulates the ideal protocol $\mathit{\Pi}(\mathcal{F})$.

UC Security with Super-polynomial Simulation. UC-SPS security is a relaxed notion of UC security where the simulator is given access to super-polynomial computational resources.

Definition 3. *Let π and ϕ be protocols. We say that π* UC-SPS-emulates *ϕ if for any adversary $\mathcal{A}$ there exists a super-polynomial-time adversary $\mathcal{S}im$ such that for any environment $\mathcal{Z}$ that obeys the rules of interaction for UC security, we have $\mathrm{EXEC}_{\phi,\mathcal{S}im,\mathcal{Z}} \approx \mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}$.*

Definition 4. *Let $\mathcal{F}$ be an ideal functionality and let π be a protocol. We say that π* UC-SPS-realizes *$\mathcal{F}$ if π UC-SPS-emulates the ideal process $\mathit{\Pi}(\mathcal{F})$.*

The ideal OT functionality $\mathcal{F}_{\mathrm{OT}}$ interacts with a sender S and a receiver R.

- Upon receiving a message $(\mathsf{sid}, \mathsf{sender}, v_0, v_1)$ from S, where each $v_i \in \{0,1\}^n$, store (v_0, v_1).
- Upon receiving a message $(\mathsf{sid}, \mathsf{receiver}, u)$ from R, where $u \in \{0,1\}$, check if a $(\mathsf{sid}, \mathsf{sender}, \ldots)$ message was previously sent. If yes, send (sid, v_u) to R and (sid) to the adversary $\mathcal{S}im$ and halt. If not, send nothing to R.

Fig. 1. The oblivious transfer functionality $\mathcal{F}_{\mathrm{OT}}$.

The multi-session extension of an ideal functionality. When showing concurrent security of a protocol π under SPS security, we need to construct a simulator in a setting where parties execute π concurrently. To consider the simulator in this setting, we use a *multi-session extension* of an ideal functionality [8]. Roughly speaking, the multi-session extension $\hat{\mathcal{F}}$ of an ideal functionality $\mathcal{F}$ is a functionally that internally runs multiple copies of $\mathcal{F}$.

4 Our SPS Concurrent OT Protocol

In this section, we prove the following theorem.

Theorem 2. *Assume the existence of constant-round semi-honest oblivious transfer protocols and collision-resistant hash functions. Let $\mathcal{F}_{\mathrm{OT}}$ be the ideal oblivious transfer functionality (Fig. 1) and $\hat{\mathcal{F}}_{\mathrm{OT}}$ be its multi-session extension. Then, there exists a constant-round protocol that UC-SPS realizes $\hat{\mathcal{F}}_{\mathrm{OT}}$, and it uses the underlying primitives in the black-box way.*

4.1 Protocol Description

In our protocol, we use the following building blocks.

- A two-round statistically binding commitment Com and a four-round statistically binding extractable commitment ExtCom, both of which can be constructed from one-way functions in the black-box way [39,25,46].
- A $O(1)$-round OT protocol $\mathsf{mS\text{-}OT}$ that is secure against malicious senders and semi-honest receivers.[8] As shown in [24], such a OT protocol can be obtained from any semi-honest one in the black-box way.
- A $O(1)$-round parallel non-malleable commitment NMCom that is parallel k-robust for sufficiently large constant k. (Concretely, we require that k is larger than the round complexity of the above three building blocks.) As remarked in Section 3.1, we show in Appendix B that such a non-malleable commitment scheme can be constructed from collision-resistant hash functions in the black-box way.

[8] We only requires $\mathsf{mS\text{-}OT}$ to be secure under a game-based definition (which is preserved under parallel composition). For details, see the the proofs of Lemma 5 and Claim 5.

Our OT protocol $\mathit{\Pi}_{\mathrm{OT}}$ is described below. As explained in Section 2.1, (1) our protocol is based on the OT protocol of Lin and Pass [32], which roughly consists of coin-tossing, semi-honest OT, OT combiner, and cut-and-choose, and (2) our protocol additionally uses non-malleable commitments, which will be used in the security proof to argue that the adversary cannot make the "trapdoor statement" true even in the concurrent setting. Below, we give intuitive explanations in italic.

Inputs: The input to the sender S is $v_0, v_1 \in \{0,1\}^n$. The input to the receiver R is $u \in \{0,1\}$.

Stage 1: (Preprocess for cut-and-choose)

1. S commits to a random subset $\mathit{\Gamma}_S \subset [11n]$ of size n by using Com.
2. R commits to a random subset $\mathit{\Gamma}_R \subset [11n]$ of size n by using Com.

Comment: *As in the OT protocol of Lin and Pass [32], the subsets that will be used in the cut-and-choose stages are committed in advance to prevent selective opening attacks.*

Stage 2:

1. **(Coin tossing for S)** S commits to random strings $\boldsymbol{a}^S = (a_1^S, \dots, a_{11n}^S)$ by using Com; let $d_1^S, \dots, d_{11n}^S$ be the decommitments. R then sends random strings $\boldsymbol{b}^S = (b_1^S, \dots, b_{11n}^S)$ to S. S then defines $\boldsymbol{r}^S = (r_1^S, \dots, r_{11n}^S)$ by $r_i^S \stackrel{\text{def}}{=} a_i^S \oplus b_i^S$ for each $i \in [11n]$ and parses r_i^S as $s_{i,0} \parallel s_{i,1} \parallel \tau_i^S$ for each $i \in [11n]$.
2. **(Coin tossing for R)** R commits to random strings $\boldsymbol{a}^R = (a_1^R, \dots, a_{11n}^R)$ by using Com; let $d_1^R, \dots, d_{11n}^R$ be the decommitments. S then sends random strings $\boldsymbol{b}^R = (b_1^R, \dots, b_{11n}^R)$ to R. R then defines $\boldsymbol{r}^R = (r_1^R, \dots, r_{11n}^R)$ by $r_i^R \stackrel{\text{def}}{=} a_i^R \oplus b_i^R$ for each $i \in [11n]$ and parses r_i^R as $c_i \parallel \tau_i^R$ for each $i \in [11n]$.

Stage 3: (mS-OTs with random inputs)

S and R execute $11n$ instances of $\mathsf{mS\text{-}OT}$ in parallel. In the i-th instance, S uses $(s_{i,0}, s_{i,1})$ as the input and τ_i^S as the randomness, and R uses c_i as the input and τ_i^R as the randomness, where $\{s_{i,0}, s_{i,1}, \tau_i^S\}_i$ and $\{c_i, \tau_i^R\}_i$ are the random coins that were obtained in Stage 2. The output to R is denoted by $\widetilde{s}_1, \dots, \widetilde{s}_{11n}$, which are supposed to be equal to $s_{1,c_1}, \dots, s_{11n,c_{11n}}$.

Stage 4: (NMCom and ExtCom for checking honesty of R)

1. R commits to $(a_1^R, d_1^R), \dots (a_{11n}^R, d_{11n}^R)$ using NMCom. Let $e_1^R, \dots, e_{11n}^R$ be the decommitments.
2. R commits to $(a_1^R, d_1^R, e_1^R), \dots (a_{11n}^R, d_{11n}^R, e_{11n}^R)$ using ExtCom.

Comment: *Roughly, the commitments in this stage, along with the cut-and-choose in the next stage, will be used in the security proof to argue that even cheating R must behave honestly in most instances of* $\mathsf{mS\text{-}OT}$ *in Stage 3. A key point is that given the values that are committed to in* NMCom *or* ExtCom *in this stage, one can obtain the random coins that R obtained in Stage 2 and thus can check whether R behaved honestly in Stage 3.*

Stage 5: (Cut-and-choose against R)

1. S reveals $\mathit{\Gamma}_S$ by decommitting the Com commitment in Stage 1-1.
2. For every $i \in \mathit{\Gamma}_S$, R reveals (a_i^R, d_i^R, e_i^R) by decommitting the i-th ExtCom commitment in Stage 4.

3. For every $i \in \mathit{\Gamma}_S$, S checks the following.
   - $((a_i^R, d_i^R), e_i^R)$ is a valid decommitment of the i-th NMCom commitment in Stage 4.
   - (a_i^R, d_i^R) is a valid decommitment of the i-th Com commitment in Stage 2-2.
   - R executed the i-th mS-OT in Stage 3 honestly using $c_i \,\|\, \tau_i^R$, which is obtained from $r_i^R = a_i^R \oplus b_i^R$ as specified by the protocol.

Comment: *In other words, for each index that it randomly selected in Stage 1, S checks whether R behaved honestly in Stages 3 and 4 on that index.*

Stage 6: (OT combiner) Let $\mathit{\Delta} := [11n] \setminus \mathit{\Gamma}_S$.

1. R sends $\alpha_i := u \oplus c_i$ to S for every $i \in \mathit{\Delta}$.
2. S computes a $(6n+1)$-out-of-$10n$ secret sharing of v_0, denoted by $\boldsymbol{\rho}_0 = (\rho_{0,i})_{i \in \mathit{\Delta}}$, and computes a $(6n+1)$-out-of-$10n$ secret sharing of v_1, denoted by $\boldsymbol{\rho}_1 = (\rho_{1,i})_{i \in \mathit{\Delta}}$. Then, S sends $\beta_{b,i} := \rho_{b,i} \oplus s_{i,b\oplus\alpha_i}$ to R for every $i \in \mathit{\Delta}$, $b \in \{0,1\}$.
3. R computes $\widetilde{\rho}_i := \beta_{u,i} \oplus \widetilde{s}_i$ for every $i \in \mathit{\Delta}$. Let $\widetilde{\boldsymbol{\rho}} := (\widetilde{\rho}_i)_{i \in \mathit{\Delta}}$.

Comment: *In this stage, S and R execute OT with their true inputs by using the outputs of* mS-OT *in Stage 3. Roughly speaking, this stage is secure as long as most instances of* mS-OT *in Stage 3 are correctly executed.*

Stage 7: (NMCom and ExtCom for checking honesty of S)

1. S commits to $(a_1^S, d_1^S), \ldots (a_{11n}^S, d_{11n}^S)$ using NMCom. Let $e_1^S, \ldots, e_{11n}^S$ be the decommitments.
2. S commits to $(a_1^S, d_1^S, e_1^S), \ldots (a_{11n}^S, d_{11n}^S, e_{11n}^S)$ using ExtCom.

Stage 8: (Cut-and-choose against S)

1. R reveals $\mathit{\Gamma}_R$ by decommitting the Com commitment in Stage 1-2.
2. For every $i \in \mathit{\Gamma}_R$, S reveals (a_i^S, d_i^S, e_i^S) by decommitting the i-th ExtCom commitment in Stage 7.
3. For every $i \in \mathit{\Gamma}_R$, R checks the following.
   - $((a_i^S, d_i^S), e_i^S)$ is a valid decommitment of the i-th NMCom commitment in Stage 7.
   - (a_i^S, d_i^S) is a valid decommitment of the i-th Com commitment in Stage 2-1.
   - S executed the i-th mS-OT in Stage 3 honestly using $s_{i,0} \,\|\, s_{i,1} \,\|\, \tau_i^S$, which is obtained from $r_i^S = a_i^S \oplus b_i^S$ as specified by the protocol.

Output: R outputs $\mathsf{Value}(\widetilde{\boldsymbol{\rho}}, \mathit{\Gamma}_R \cap \mathit{\Delta})$, where $\mathsf{Value}(\cdot,\cdot)$ is the function that is defined in Fig. 2.

Comment: *As in the OT protocol of Lin and Pass [32], a carefully designed reconstruction procedure* $\mathsf{Value}(\cdot,\cdot)$ *is used here so that the simulator can extract correct implicit inputs from cheating S by obtaining sharing that is sufficiently "close" to $\widetilde{\boldsymbol{\rho}}$.*

4.2 Simulator $\mathcal{S}im_{\mathrm{OT}}$

To prove the security of $\mathit{\Pi}_{\mathrm{OT}}$, we consider the following simulator $\mathcal{S}im_{\mathrm{OT}}$. Recall that our goal is to prove that $\mathit{\Pi}_{\mathrm{OT}}$ US-SPS realizes the multi-session extension

Reconstruction procedure $\mathsf{Value}(\cdot,\cdot)$*:* For a sharing $\boldsymbol{s} = (s_i)_{i \in \Delta}$ and a set $\Theta \subset \Delta$, the output of $\mathsf{Value}(\boldsymbol{s}, \Theta)$ is computed as follows. If $\boldsymbol{s}$ is 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i \in \Delta}$ that satisfies $s_i = w_i$ for every $i \in \Theta$, then $\mathsf{Value}(\boldsymbol{s}, \Theta)$ is the value decoded from $\boldsymbol{w}$; otherwise, $\mathsf{Value}(\boldsymbol{s}, \Theta) = \perp$.

Fig. 2. The function $\mathsf{Value}(\cdot,\cdot)$.

of $\mathcal{F}_{\mathrm{OT}}$. We therefore consider a simulator that works against adversaries that participate in multiple sessions of Π_{OT} both as senders and as receivers.

Let $\mathcal{Z}$ be any environment, $\mathcal{A}$ be any adversary that participates in multiple sessions of Π_{OT}. Our simulator $\mathcal{S}im_{\mathrm{OT}}$ internally invokes $\mathcal{A}$ and simulates each of the sessions for $\mathcal{A}$ as follows.

When $\boldsymbol{R}$ is corrupted: In a session where the receiver R is corrupted, $\mathcal{S}im_{\mathrm{OT}}$ simulates the sender S for $\mathcal{A}$ by extracting the implicit input $u^* \in \{0,1\}$ from $\mathcal{A}$. During the simulation, $\mathcal{S}im_{\mathrm{OT}}$ extracts the committed subset and random coins in Stages 1 and 2 by brute force; the former extraction is needed to execute most instances $\mathsf{mS\text{-}OT}$ in Stage 3 with true randomness (which is crucial to use their security in the analysis), and the latter extraction is needed to infer what information $\mathcal{A}$ obtained in the $\mathsf{mS\text{-}OT}$ instances in Stage 3 (which is crucial to extract the implicit input $u^* \in \{0,1\}$ from $\mathcal{A}$).

Concretely, $\mathcal{S}im_{\mathrm{OT}}$ simulates S for $\mathcal{A}$ as follows. From Stage 1 to Stage 5, $\mathcal{S}im_{\mathrm{OT}}$ interacts with $\mathcal{A}$ in the same way as an honest S except for the following.

- From the Com commitments from $\mathcal{A}$ in Stages 1 and 2, the committed subset Γ_R and the committed strings $\boldsymbol{a}^R = (a_1^R, \ldots, a_{11n}^R)$ are extracted by brute force.
  $\mathcal{S}im_{\mathrm{OT}}$ then defines $\boldsymbol{r}^R = (r_1^R, \ldots, r_{11n}^R)$ by $r_i^R \stackrel{\text{def}}{=} a_i^R \oplus b_i^R$ for each $i \in [11n]$ and parses r_i^R as $c_i \,\|\, \tau_i^R$ for each $i \in [11n]$. (Notice that $\boldsymbol{r}^R$ is the outcome of the coin-tossing that $\mathcal{A}$ must have obtained.)
- In Stage 3, the i-th mS-OT is executed with a random input and true randomness rather than with $(s_{i,0}, s_{i,1})$ and τ_i^S for every $i \notin \Gamma_R$.

In Stage 6, $\mathcal{S}im_{\mathrm{OT}}$ interacts with $\mathcal{A}$ as follows.

1. Receive $\{\alpha_i\}_{i \in \Delta}$ from $\mathcal{A}$ in Stage 6-1.
2. Determine the implicit input u^* of $\mathcal{A}$ as follows. Let I_0, I_1 be the sets such that for $b \in \{0,1\}$ and $i \in \Delta$, we have $i \in I_b$ if and only if:
   - $i \in \Gamma_R$, or
   - $\mathcal{A}$ did not execute the i-th mS-OT in Stage 3 honestly using $c_i \,\|\, \tau_i^R$ as the input and randomness, or
   - $c_i = b \oplus \alpha_i$, and $\mathcal{A}$ executed the i-th mS-OT in Stage 3 honestly using $c_i \,\|\, \tau_i^R$ as the input and randomness.

   Abort the simulation if both of $|I_0| \geq 6n+1$ and $|I_1| \geq 6n+1$ hold. Otherwise, define u^* by $u^* \stackrel{\text{def}}{=} 0$ if $|I_0| \geq 6n+1$ and $u^* \stackrel{\text{def}}{=} 1$ otherwise. (Roughly, $|I_b|$

is the number of strings that $\mathcal{A}$ can obtain out of $\{s_{i,b\oplus\alpha_i}\}_{i\in\Delta}$ by requiring S to reveal them in Stage 8, by cheating in mS-OT, or by executing mS-OT honestly with input $b\oplus\alpha_i$. We remind the readers that $\{s_{i,b\oplus\alpha_i}\}_{i\in\Delta}$ are the strings that are used to mask $\boldsymbol{\rho}_b = (\rho_{b,i})_{i\in\Delta}$ in Stage 6.)

3. Send u^* to the ideal functionality and obtains v^*.
4. Subsequently, interact with $\mathcal{A}$ in the same way as an honest S assuming that the inputs to S are $v_{u^*} = v^*$ and random v_{1-u^*}.

From Stage 7 to Stage 8, $\mathcal{S}im_{\mathrm{OT}}$ interacts with $\mathcal{A}$ in the same way as an honest S except that in Stage 7, an all-zero string is committed in the i-th NMCom rather than (a_i^S, d_i^S) for every $i \notin \Gamma_R$, and an all-zero string is committed in the i-th ExtCom rather than (a_i^S, d_i^S, e_i^S) for every $i \notin \Gamma_R$.

When S is corrupted: In a session where the sender S is corrupted, $\mathcal{S}im_{\mathrm{OT}}$ simulates the receiver R for $\mathcal{A}$ by extracting the implicit input v_0^*, v_1^* from $\mathcal{A}$. During the simulation, $\mathcal{S}im_{\mathrm{OT}}$ extracts the committed subset and random coins in Stages 1 and 2 by brute force; the former extraction is needed to execute most instances $\mathsf{mS\text{-}OT}$ in Stage 3 with true randomness (which is crucial to use their security in the analysis), and the latter extraction is needed to learn what input $\mathcal{A}$ used in the $\mathsf{mS\text{-}OT}$ instances in Stage 3 (which is crucial to extract the implicit input v_0^*, v_1^* from $\mathcal{A}$).

Concretely, $\mathcal{S}im_{\mathrm{OT}}$ simulates R for $\mathcal{A}$ as follows. $\mathcal{S}im_{\mathrm{OT}}$ interacts with $\mathcal{A}$ in the same way as an honest R in all the stages except for the following.

- From the Com commitment from $\mathcal{A}$ in Stage 1, the committed subset Γ_S is extracted by brute force.
- In Stage 3, the i-th mS-OT is executed with a random input and true randomness rather than with c_i and τ_i^R for every $i \notin \Gamma_S$.
- In Stage 4, an all-zero string is committed in the i-th NMCom rather than (a_i^S, d_i^S) for every $i \notin \Gamma_S$, and an all-zero string is committed in the i-th ExtCom rather than (a_i^S, d_i^S, e_i^S) for every $i \notin \Gamma_S$.
- In Stage 6, α_i is a random bit rather than $\alpha_i = u \oplus c_i$ for every $i \in \Delta$, and $\widetilde{\rho}_i$ is not computed for any $i \in \Delta$.

Then, $\mathcal{S}im_{\mathrm{OT}}$ determines the implicit inputs v_0^*, v_1^* of $\mathcal{A}$ as follows.

1. From the Com commitments from $\mathcal{A}$ in Stage 2, extract the committed strings $\boldsymbol{a}^S = (a_1^S, \ldots, a_{11n}^S)$ by brute force.
2. Define $\boldsymbol{r}^S = (r_1^S, \ldots, r_{11n}^S)$ by $r_i^S \overset{\mathrm{def}}{=} a_i^S \oplus b_i^S$ for each $i \in [11n]$ and parse r_i^S as $s_{i,0} \parallel s_{i,1} \parallel \tau_i^S$ for each $i \in [11n]$. (Notice that $\boldsymbol{r}^S$ is the outcome of the coin-tossing that $\mathcal{A}$ must have obtained.)
3. Define $\boldsymbol{\rho}_b^{\mathsf{ext}} = (\rho_{b,i}^{\mathsf{ext}})_{i\in\Delta}$ for each $b \in \{0,1\}$ as follows: $\rho_{b,i}^{\mathsf{ext}} \overset{\mathrm{def}}{=} \beta_{b,i} \oplus s_{i,b\oplus\alpha_i}$ if $\mathcal{A}$ executed the i-th mS-OT in stage 3 honestly using $s_{i,0} \parallel s_{i,1} \parallel \tau_i^S$, and $\rho_{b,i}^{\mathsf{ext}} \overset{\mathrm{def}}{=} \bot$ otherwise.
4. For each $b \in \{0,1\}$, define $v_b^* \overset{\mathrm{def}}{=} \mathsf{Value}(\boldsymbol{\rho}_b^{\mathsf{ext}}, \Gamma_R \cap \Delta)$.

Then, $\mathcal{S}im_{\mathrm{OT}}$ sends v_0^*, v_1^* to the ideal functionality if both of the following hold for each $b \in \{0,1\}$.

1. $|\{i \in \Delta \text{ s.t. } \rho_{b,i}^{\mathsf{ext}} = \bot\}| < 0.1n$.
2. $\boldsymbol{\rho}_b^{\mathsf{ext}}$ is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i \in \Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{ext}}$ for every $i \in \Gamma_R$ or 0.14-far from any such valid codeword.

Otherwise (i.e., if there exists $b \in \{0,1\}$ such that one of the above does not holds), $\mathcal{S}im_{\mathrm{OT}}$ aborts the simulation.

4.3 Proof of Indistinguishability

We show the indistinguishability by using a hybrid argument. Before defining hybrid experiments, we define *special messages*, which we use in the definitions of the hybrid experiments. (Essentially, they are the messages on which the simulator applies brute-force extractions.)

- first special message is the Com commitment in Stage 1-1.
- second special message is the Com commitment in Stage 1-2.
- third special message is the Com commitments in Stage 2-1.
- fourth special message is the Com commitments in Stage 2-2.

Hybrid experiments. Now, we define hybrid experiments. Let m denote an upper bound on the number of the sessions that $\mathcal{A}$ starts. Note that the number of special messages among m sessions can be bounded by $4m$. We order those $4m$ special messages by the order of their appearances; we use SM_k to denote the k-th special messages, and $s(k)$ to denote the session that SM_k belongs to.

We define hybrids H_0 and $H_{k:1}, \ldots, H_{k:7}$ ($k \in [4m]$) as follows. (For convenience, in what follows we occasionally denote H_0 as $H_{0:7}$.)

Remark 2 (Rough idea of the hybrids). In the sequence of the hybrid experiments, we gradually modify the read-world experiment to the ideal-world one. All the experiments (except for H_0) involve super-polynomial-time brute-force extraction, but we make sure that $H_{k:i}$ ($i \in [7]$) involves brute-force extraction only until SM_k, and it deviates from the previous hybrid only after SM_k. These properties help us prove the indistinguishability of each neighboring hybrids because we can think the results of brute-force extraction as non-uniform advice and use the non-uniform security of the underlying primitives to show the indistinguishability.[9] ◊

Hybrid H_0. H_0 is the same as the real experiment.
Hybrid $H_{k:1}$. $H_{k:1}$ is the same as $H_{k-1:7}$ except that in session $s(k)$, if S is corrupted and SM_k is first special message,

[9] We remark that, unlike Garg et al. [15] (who give a non-black-box constant-round SPS protocol), we cannot replace brute-force extraction with rewinding extraction for obtaining polynomial-time hybrids. This is because when considering black-box constructions, we cannot easily guarantee that brute-force extraction and rewinding one obtain the same value.

- the committed subset $\mathit{\Gamma}_S$ is extracted by brute force in Stage 1-1,
- the value committed to in the i-th NMCom commitment in Stage 4 is switched to an all-zero string for every $i \notin \mathit{\Gamma}_S$, and
- the value committed to in the i-th ExtCom commitment in Stage 4 is switched to an all-zero string for every $i \notin \mathit{\Gamma}_S$.

Hybrid $H_{k:2}$. $H_{k:2}$ is the same as $H_{k:1}$ except that in session $s(k)$, if S is corrupted and SM_k is first special message, the i-th mS-OT in Stage 3 is executed with a random input and true randomness for every $i \notin \mathit{\Gamma}_S$.

Hybrid $H_{k:3}$. $H_{k:3}$ is the same as $H_{k:2}$ except that in session $s(k)$, if S is corrupted and SM_k is third special message, the following modifications are made.

1. The committed strings $\boldsymbol{a}^S = (a_1^S, \dots, a_{11n}^S)$ are extracted by brute force in Stage 2-1. Define $\boldsymbol{r}^S = (r_1^S, \dots, r_{11n}^S)$ by $r_i^S \overset{\text{def}}{=} a_i^S \oplus b_i^S$ for each $i \in [11n]$, and parse r_i^S as $s_{i,0} \,\|\, s_{i,1} \,\|\, \tau_i^S$ for each $i \in [11n]$. Define $\boldsymbol{\rho}_b^{\mathsf{ext}} = (\rho_{b,i}^{\mathsf{ext}})_{i \in \mathit{\Delta}}$ for each $b \in \{0,1\}$ as follows: $\rho_{b,i}^{\mathsf{ext}} \overset{\text{def}}{=} \beta_{b,i} \oplus s_{i,b\oplus\alpha_i}$ if $\mathcal{A}$ executed the i-th mS-OT in stage 3 honestly using $s_{i,0} \,\|\, s_{i,1} \,\|\, \tau_i^S$, and $\rho_{b,i}^{\mathsf{ext}} = \bot$ otherwise.
2. R outputs $\mathsf{Value}(\boldsymbol{\rho}_u^{\mathsf{ext}}, \mathit{\Gamma}_R \cap \mathit{\Delta})$ rather than $\mathsf{Value}(\widetilde{\rho}, \mathit{\Gamma}_R \cap \mathit{\Delta})$ (recall that u is the real input to R) if both of the following hold for each $b \in \{0,1\}$.
   (a) $|\{i \in \mathit{\Delta} \text{ s.t. } \rho_{b,i}^{\mathsf{ext}} = \bot\}| < 0.1n$.
   (b) $\boldsymbol{\rho}_b^{\mathsf{ext}}$ is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i \in \mathit{\Delta}}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{ext}}$ for every $i \in \mathit{\Gamma}_R$ or 0.15-far from any such valid codeword.

   Otherwise (i.e., if there exists $b \in \{0,1\}$ such that one of the above does not holds), the execution of the hybrid is aborted.

Hybrid $H_{k:4}$. $H_{k:4}$ is the same as $H_{k:3}$ except that in session $s(k)$, if S is corrupted and SM_k is third special message, α_i is a random bit rather than $\alpha_i = u \oplus c_i$ for every $i \in \mathit{\Delta}$ in Stage 6-1 and $\widetilde{\rho}_i$ is no longer computed for any $i \in \mathit{\Delta}$ in Stage 6-3.

Hybrid $H_{k:5}$. $H_{k:5}$ is the same as $H_{k:4}$ except that in session $s(k)$, if R is corrupted and SM_k is second special message,

- the committed subset $\mathit{\Gamma}_R$ is extracted by brute force in Stage 1-2,
- the value committed in the i-th NMCom commitment in Stage 7 is switched to an all-zero string for every $i \notin \mathit{\Gamma}_R$, and
- the value committed in the i-th ExtCom commitment in Stage 7 is switched to an all-zero string for every $i \notin \mathit{\Gamma}_R$.

Hybrid $H_{k:6}$. $H_{k:6}$ is the same as $H_{k:5}$ except that in session $s(k)$, if R is corrupted and SM_k is second special message, the i-th mS-OT in Stage 3 is executed with a random input and true randomness for every $i \notin \mathit{\Gamma}_R$.

Hybrid $H_{k:7}$. $H_{k:7}$ is the same as $H_{k:6}$ except that in session $s(k)$, if R is corrupted and SM_k is fourth special message, the following modifications are made.

1. The committed strings $\boldsymbol{a}^R = (a_1^R, \dots, a_{11n}^R)$ are extracted by brute force in Stage 2-2. Define $\boldsymbol{r}^R = (r_1^R, \dots, r_{11n}^R)$ by $r_i^R \overset{\text{def}}{=} a_i^R \oplus b_i^R$ for each $i \in [11n]$, and parse r_i^R as $c_i \,\|\, \tau_i^R$ for each $i \in [11n]$. Define u^* as follows. Let I_0 and I_1 be the set such that for $b \in \{0,1\}$ and $i \in \mathit{\Delta}$, we have $i \in I_b$ if and only if:

   - $i \in \Gamma_R$, or
   - $\mathcal{A}$ did not execute the i-th mS-OT in Stage 3 honestly using $c_i \,\|\, \tau_i^R$ as the input and randomness, or
   - $c_i = b \oplus \alpha_i$, and $\mathcal{A}$ executed the i-th mS-OT in Stage 3 honestly using $c_i \,\|\, \tau_i^R$ as the input and randomness.

   Abort the execution of the hybrid if both of $|I_0| \geq 6n+1$ and $|I_1| \geq 6n+1$ hold. Otherwise, define u^* by $u^* \overset{\text{def}}{=} 0$ if $|I_0| \geq 6n+1$ and $u^* \overset{\text{def}}{=} 1$ otherwise.
2. In Stage 6, $\boldsymbol{\rho}_{1-u^*}$ is a secret sharing of a random bit rather than that of v_{1-u^*}.

We remark that in $H_{4m:7}$, all the messages from the honest parties and their output are computed as in $\mathcal{S}im_{\mathrm{OT}}$.

Indistinguishability of each neighboring hybrids. Below, we show that each neighboring hybrids are indistinguishable, and additionally show, for technical reasons, that an invariant condition holds in each session of every hybrid.

First, we define the invariant condition.

Definition 5 (Invariant Condition (when R is corrupted)). *For any session in which R is corrupted, we say that the invariant condition holds in that session if the following holds when the cut-and-choose in Stage 5 is accepted.*

1. *Let* $(\hat{a}_1^R, \hat{d}_1^R), \ldots (\hat{a}_{11n}^R, \hat{d}_{11n}^R)$ *be the values that are committed in* NMCom *in Stage 4. Let* $I_{\mathrm{bad}} \subset [11n]$ *be the set such that* $i \in I_{\mathrm{bad}}$ *if and only if*
   (a) $(\hat{a}_i^R, \hat{d}_i^R)$ *is not a valid decommitment of the i-th* Com *commitment in Stage 2-2, or*
   (b) R does not execute the i-th mS-OT in Stage 3 honestly using $\hat{c}_i \,\|\, \hat{\tau}_i^R$ *as the input and randomness, where* $\hat{c}_i \,\|\, \hat{\tau}_i^R$ *is obtained from* $\hat{r}_i^R = \hat{a}_i^R \oplus b_i^R$.

   Then, it holds that $|I_{\mathrm{bad}}| < n$.

Remark 3. Roughly speaking, this condition guarantees that most of the mS-OTs in Stage 3 are honestly executed using the outcome of the coin tossing, which in turn guarantees that the cheating receiver's input can be extracted by extracting the outcome of the coin tossing. ♢

Remark 4. When Stage 5 is accepted, we also have $I_{\mathrm{bad}} \cap \Gamma_S = \emptyset$ from the definition of I_{bad}. ♢

Definition 6 (Invariant Condition (when S is corrupted)). *For any session in which S is corrupted, we say that the invariant condition holds in that session if the following hold when the cut-and-choose in Stage 8 is accepted.*

1. *Let* $(\hat{a}_1^S, \hat{d}_1^S), \ldots (\hat{a}_{11n}^S, \hat{d}_{11n}^S)$ *be the values that are committed in* NMCom *in Stage 7. Let* $I_{\mathrm{bad}} \subset [11n]$ *be the set such that* $i \in I_{\mathrm{bad}}$ *if and only if*
   (a) $(\hat{a}_i^S, \hat{d}_i^S)$ *is not a valid decommitment of the i-th* Com *commitment in Stage 2-1, or*

(b) S does not execute the i-th mS-OT in Stage 3 honestly using $\hat{s}_{i,0} \parallel \hat{s}_{i,1} \parallel \hat{\tau}_i^S$ as the input and randomness, where $\hat{s}_{i,0} \parallel \hat{s}_{i,1} \parallel \hat{\tau}_i^S$ is obtained from $\hat{r}_i^S = \hat{a}_i^S \oplus b_i^S$.

Then, it holds that $|I_{\mathrm{bad}}| < 0.1n$.

2. *For each $b \in \{0,1\}$, define $\boldsymbol{\rho}_b^{\mathsf{nm}} = (\rho_{b,i}^{\mathsf{nm}})_{i \in \Delta}$ as follows: $\rho_{b,i}^{\mathsf{nm}} \stackrel{\mathrm{def}}{=} \beta_{b,i} \oplus \hat{s}_{i,b\oplus\alpha_i}$ if $i \notin I_{\mathrm{bad}}$ and $\rho_{b,i}^{\mathsf{nm}} \stackrel{\mathrm{def}}{=} \perp$ otherwise. Then, for each $b \in \{0,1\}$, ρ_b^{nm} is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i \in \Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{nm}}$ for every $i \in \Gamma_R$ or 0.15-far from any such valid codeword.*

Remark 5. Roughly speaking, this condition guarantees that the cheating sender's input can be extracted from the outcome of the coin tossing. In particular, it guarantees that the sharing that is computed from the outcome of mS-OTs (i.e., the sharing that is computed by the honest receiver) and the sharing that is computed from the outcome of the coin tossing (i.e., the sharing that is computed by the simulator) are very "close" (see Claim 3 below). ♢

Remark 6. When Stage 8 is accepted, we also have $I_{\mathrm{bad}} \cap \Gamma_R = \emptyset$ from the definition of I_{bad}. ♢

Next, we show that the invariant condition holds in every session in H_0 (i.e., the real experiment).

Definition 7. *We say that $\mathcal{A}$* ***cheats*** *in a session if the invariant condition does not hold in that session.*

Lemma 1. *In H_0, $\mathcal{A}$ does not cheat in every session except with negligible probability.*

Proof. Assume for contradiction that in H_0, $\mathcal{A}$ cheats in a session with non-negligible probability. Since the number of the sessions is bounded by a polynomial, there exists a function $i^*(\cdot)$ and a polynomial $p(\cdot)$ such that for infinitely many n, $\mathcal{A}$ cheats in the $i^*(n)$-th session with probability at least $1/p(n)$; furthermore, since $\mathcal{A}$ cheats only when either R or S is corrupted, in the $i^*(n)$-th session either R is corrupted for infinitely many such n or S is corrupted for infinitely many such n. In both cases, we derive contradiction by using $\mathcal{A}$ to break the hiding property of Com.

Case 1. R is corrupted in the $i^*(n)$-th session. We show that when $\mathcal{A}$ cheats, we can break the hiding property of the Com commitment in Stage 1-1 (i.e., the commitment by which Γ_S is committed to). From the definition of the invariant condition (Definition 5), when $\mathcal{A}$ cheats, we have $|I_{\mathrm{bad}}| \geq n$ even though the cut-and-choose in Stage 5 is accepting (and hence $I_{\mathrm{bad}} \cap \Gamma_S = \emptyset$ as remarked in Remark 4), where $I_{\mathrm{bad}} \subseteq [11n]$ is the set defined from the committed values of the NMCom commitments in Stage 4. If we can compute I_{bad} efficiently, we can use it to distinguish Γ_S from a random subset of size n (this is because a random subset Γ of size n satisfies $I_{\mathrm{bad}} \cap \Gamma = \emptyset$ only with negligible probability when $|I_{\mathrm{bad}}| \geq n$), so we can use it to break the hiding property of the commitment to

$\mathit{\Gamma}_S$. However, I_{bad} is not efficiently computable since the committed values of the NMCom commitments are not efficiently computable. We thus first show that we can "approximate" I_{bad} by extracting the committed values of the ExtCom commitments in Stage 4. Details are given below.

First, we observe that if we extract the committed values of the ExtCom commitments in Stage 4 of the $i^*(n)$-th session, the extracted values, $(\hat{a}_1^R, \hat{d}_1^R, \hat{e}_1^R), \ldots, (\hat{a}_{11n}^R, \hat{d}_{11n}^R, \hat{e}_{11n}^R)$, satisfy the following condition.

- Let $\hat{I}_{\mathrm{bad}} \subset [11n]$ be a set such that $i \in \hat{I}_{\mathrm{bad}}$ if and only if
  1. $((\hat{a}_i^R, \hat{d}_i^R), \hat{e}_i^R)$ is not a valid decommitment of the i-th NMCom commitment in Stage 4, or
  2. $(\hat{a}_i^R, \hat{d}_i^R)$ is not a valid decommitment of the i-th Com commitment in Stage 2-2, or
  3. R does not execute the i-th mS-OT in Stage 3 honestly using $\hat{c}_i \,\|\, \hat{\tau}_i^R$ as the input and randomness, where $\hat{c}_i \,\|\, \hat{\tau}_i^R$ is obtained from $\hat{r}_i^R = \hat{a}_i^R \oplus b_i^R$.

  Then, $|\hat{I}_{\mathrm{bad}}| \geq n$ and $\hat{I}_{\mathrm{bad}} \cap \mathit{\Gamma}_S = \emptyset$ with probability at least $1/2p(n)$.

The extracted values satisfy this condition because when $\mathcal{A}$ cheats, we have $|\hat{I}_{\mathrm{bad}}| \geq n$ and $\hat{I}_{\mathrm{bad}} \cap \mathit{\Gamma}_S = \emptyset$ except with negligible probability. (We have $|\hat{I}_{\mathrm{bad}}| \geq n$ since we have $I_{\mathrm{bad}} \subset \hat{I}_{\mathrm{bad}}$ from the definitions of $I_{\mathrm{bad}}, \hat{I}_{\mathrm{bad}}$ and the binding property of NMCom. We have $\hat{I}_{\mathrm{bad}} \cap \mathit{\Gamma}_S = \emptyset$ since when the cut-and-choose in Stage 5 is accepting, for every $i \in \mathit{\Gamma}_S$ the i-th ExtCom commitment is a valid decommitment of the i-th NMCom commitment, and $I_{\mathrm{bad}} \cap \mathit{\Gamma}_S = \emptyset$.)

Based on this observation, we derive contradiction by considering the following adversary $\mathcal{A}_{\mathsf{Com}}$ against the hiding property of Com.

> $\mathcal{A}_{\mathsf{Com}}$ receives a Com commitment c^* in which either $\mathit{\Gamma}_S^0$ or $\mathit{\Gamma}_S^1$ is committed, where $\mathit{\Gamma}_S^0, \mathit{\Gamma}_S^1 \subset [11n]$ are random subsets of size n.
> Then, $\mathcal{A}_{\mathsf{Com}}$ internally executes the experiment H_0 honestly except that in the $i^*(n)$-th session, $\mathcal{A}_{\mathsf{Com}}$ uses c^* as the commitment in Stage 1-1 (i.e., as the Com commitment in which S commits to a subset $\mathit{\Gamma}_S$). When the experiment H_0 reaches Stage 4 of the $i^*(n)$-th session, $\mathcal{A}_{\mathsf{Com}}$ extracts the committed values of the ExtCom commitments in this stage by using its extractability.[10] Let $\hat{I}_{\mathrm{bad}} \subset [11n]$ be the set that is defined as above from the extracted values. Then, $\mathcal{A}_{\mathsf{Com}}$ outputs 1 if and only if $|\hat{I}_{\mathrm{bad}}| \geq n$ and $\hat{I}_{\mathrm{bad}} \cap \mathit{\Gamma}_S^1 = \emptyset$.

If $\mathcal{A}_{\mathsf{Com}}$ receives a commitment to $\mathit{\Gamma}_S^1$, $\mathcal{A}_{\mathsf{Com}}$ outputs 1 with probability at least $1/2p(n)$ (this follows from the above observation). In contrast, if $\mathcal{A}_{\mathsf{Com}}$ receives a commitment to $\mathit{\Gamma}_S^0$, $\mathcal{A}_{\mathsf{Com}}$ outputs 1 with exponentially small probability (this is because when no information about $\mathit{\Gamma}_S^1$ is fed into H_0, the probability that $|\hat{I}_{\mathrm{bad}}| \geq n$ but $\hat{I}_{\mathrm{bad}} \cap \mathit{\Gamma}_S^1 = \emptyset$ is exponentially small). Hence, $\mathcal{A}_{\mathsf{Com}}$ breaks the hiding property of Com.

[10] This extraction involves rewinding the execution of the whole experiment, i.e., the executions of the environment, the adversary, and all the other parties.

Case 2. S is corrupted in the $i^*(n)$-th session. The proof for this case is similar to (but a little more complex than) the one for Case 1. Specifically, we show that if the invariant condition does not hold, we can break the hiding property of Com in Stage 1-2 by approximating I_{bad} using the extractability of ExtCom. We give a formal proof for this case in Appendix D. (A somewhat similar proof is given as the proof of Claim 4 later.) □

Finally, we show the indistinguishability between each neighboring hybrids.

Lemma 2. *Assume that in $H_{k-1:7}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H_{k-1:7}$ and $H_{k:1}$ are indistinguishable, and*
- *in $H_{k:1}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Proof. We prove the lemma by using a hybrid argument. Specifically, we consider the following intermediate hybrid $H'_{k-1:7}$.
Hybrid $H'_{k-1:7}$. $H'_{k-1:7}$ is the same as $H_{k-1:7}$ except that in session $s(k)$, if S is corrupted and SM_k is $\mathsf{first\ special\ message}$,

- the committed subset Γ_S is extracted by brute force in Stage 1-1, and
- the value committed to in the i-th ExtCom commitment in Stage 4 is switched to an all-zero string for every $i \notin \Gamma_S$.

Claim 1. *Assume that in $H_{k-1:7}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H_{k-1:7}$ and $H'_{k-1:7}$ are indistinguishable, and*
- *in $H'_{k-1:7}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Proof. We first show the indistinguishability between $H_{k-1:7}$ and $H'_{k-1:7}$. Assume for contradiction that $H_{k-1:7}$ and $H'_{k-1:7}$ are distinguishable. From an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $H_{k-1:7}$ and $H'_{k-1:7}$ are still distinguishable. As remarked in Remark 2, no brute-force extraction is performed after SM_k in $H_{k-1:7}$ and $H'_{k-1:7}$; hence, by considering the transcript (including the inputs and randomness of all the parties) and the extracted values up until SM_k as non-uniform advice, we can break the hiding property of ExtCom as follows.

> The adversary $\mathcal{A}_{\mathsf{ExtCom}}$ internally executes $H_{k-1:7}$ from SM_k using the non-uniform advice. In Stage 4 of session $s(k)$, $\mathcal{A}_{\mathsf{ExtCom}}$ sends $(a_i^R, d_i^R, e_i^R)_{i \notin \Gamma_S}$ and $(0,0,0)_{i \notin \Gamma_S}$ to the external committer, receives back ExtCom commitments (in which either $(a_i^R, d_i^R, e_i^R)_{i \notin \Gamma_S}$ or $(0,0,0)_{i \notin \Gamma_S}$ are committed to), and feeds them into $H_{k-1:7}$. After the execution of $H_{k-1:7}$ finishes, $\mathcal{A}_{\mathsf{ExtCom}}$ outputs whatever $\mathcal{Z}$ outputs in the experiment.

> When $\mathcal{A}_{\mathsf{ExtCom}}$ receives commitments to $(a_i^R, d_i^R, e_i^R)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H_{k-1:7}$, whereas when $\mathcal{A}_{\mathsf{ExtCom}}$ receives a commitments to $(0,0,0)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H'_{k-1:7}$. Hence, from the assumption that $H_{k-1:7}$ and $H'_{k-1:7}$ are distinguishable (even after being fixed up until SM_k), $\mathcal{A}_{\mathsf{ExtCom}}$ distinguishes ExtCom commitments.

We next show that in $H'_{k-1:7}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. Assume for contradiction that in $H'_{k-1:7}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H_{k-1:7}$ but with non-negligible probability in $H'_{k-1:7}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the robust non-malleability of NMCom as follows. (Note that the ExtCom commitments in sessions $s(k), \ldots, s(4m)$ starts only after SM_k.)

> The man-in-the-middle adversary $\mathcal{A}_{\mathsf{NMCom}}$ internally executes $H_{k-1:7}$ from SM_k using the non-uniform advice. In Stage 4 of session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ sends $(a_i^R, d_i^R, e_i^R)_{i \notin \Gamma_S}$ and $(0,0,0)_{i \notin \Gamma_S}$ to the external committer, receives back ExtCom commitments (in which either $(a_i^R, d_i^R, e_i^R)_{i \notin \Gamma_S}$ or $(0,0,0)_{i \notin \Gamma_S}$ are committed to), and feeds them into $H_{k-1:7}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver (specifically, the NMCom commitments in Stage 4 if R is corrupted and in Stage 7 if S is corrupted). After the execution of $H_{k-1:7}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.
>
> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$. (Notice that given the committed values of the NMCom commitments, $\mathcal{D}_{\mathsf{NMCom}}$ can check whether $\mathcal{A}$ cheated or not in polynomial time.)
>
> When $\mathcal{A}_{\mathsf{NMCom}}$ receives commitments to $(a_i^R, d_i^R, e_i^R)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H_{k-1:7}$, whereas when $\mathcal{A}_{\mathsf{NMCom}}$ receives a commitments to $(0,0,0)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H'_{k-1:7}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H_{k-1:7}$ but with non-negligible probability in $H'_{k-1:7}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the robust non-malleability of NMCom.

This completes the proof of Claim 1. □

Claim 2. *Assume that in $H'_{k-1:7}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H'_{k-1:7}$ and $H_{k:1}$ are indistinguishable, and*
- *in $H_{k:1}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

This claim can be proven very similarly to Claim 1 (the only difference is that we use the hiding property of NMCom rather than that of ExtCom in the first part, and use the non-malleability of NMCom rather than its robust non-malleability in the second part). We therefore give a proof in Appendix D.

This completes the proof of Lemma 2. □

Lemma 3. *Assume that in $H_{k:1}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:1}$ and $H_{k:2}$ are indistinguishable, and*
- *in $H_{k:2}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Recall that hybrids $H_{k:1}, H_{k:2}$ differ only in the input and the randomness that are used in some of the mS-OTs in Stage 3, where those that are derived from the outcomes of the coin tossing is used in $H_{k:1}$ and random inputs and true randomness are used in $H_{k:2}$. Intuitively, we prove this lemma by using the security of the coin tossing (which is guaranteed by the hiding property of Com) because it guarantees that the outcome of the coin tossing is pseudorandom. The proof is quite similar to the proof of Claim 1 (we use the hiding of Com rather than that of ExtCom), and given in Appendix D.

Lemma 4. *Assume that in $H_{k:2}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:2}$ and $H_{k:3}$ are indistinguishable, and*
- *in $H_{k:3}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Proof. Recall that $H_{k:2}$ and $H_{k:3}$ differ only in that in session $s(k)$ of $H_{k:3}$, if S is corrupted and SM_k is $\mathsf{third\ special\ message}$, either R outputs $\mathsf{Value}(\boldsymbol{\rho}_u^{\mathsf{ext}}, \mathit{\Gamma}_R \cap \mathit{\Delta})$ rather than $\mathsf{Value}(\widetilde{\boldsymbol{\rho}}, \mathit{\Gamma}_R \cap \mathit{\Delta})$ or the hybrid is aborted.

For proving the lemma, it suffices to show that in session $s(k)$ of $H_{k:3}$,

1. the hybrid is not aborted except with negligible probability, and
2. if the hybrid is not aborted, we have $\mathsf{Value}(\boldsymbol{\rho}_u^{\mathsf{ext}}, \mathit{\Gamma}_R \cap \mathit{\Delta}) = \mathsf{Value}(\widetilde{\boldsymbol{\rho}}, \mathit{\Gamma}_R \cap \mathit{\Delta})$.

To see that showing these two is indeed sufficient for proving the lemma, observe the following. First, these two imply that in session $s(k)$ of $H_{k:3}$, the probability that the hybrid is aborted or we have $\mathsf{Value}(\boldsymbol{\rho}_u^{\mathsf{ext}}, \mathit{\Gamma}_R \cap \mathit{\Delta}) \neq \mathsf{Value}(\widetilde{\boldsymbol{\rho}}, \mathit{\Gamma}_R \cap \mathit{\Delta})$ is negligible, so $H_{k:2}$ and $H_{k:3}$ are statistically close. Second, since $H_{k:2}$ and $H_{k:3}$ proceed identically until the end of session $s(k)$, and

1. if the experiment is not aborted in session $s(k)$, $H_{k:2}$ and $H_{k:3}$ continue to proceed identically after the end of session $s(k)$, and
2. if the hybrid is aborted in session $s(k)$, $\mathcal{A}$ clearly does not cheat in any session after the end of session $s(k)$,

the probability that $\mathcal{A}$ cheat in sessions $s(k), \ldots, s(4m)$ is not increased in $H_{k:3}$.

Now, we first show that in session $s(k)$ of $H_{k:3}$, the hybrid is not aborted except with negligible probability. Since $H_{k:2}$ and $H_{k:3}$ proceed identically until the end of session $s(k)$, we have that in $H_{k:3}$, $\mathcal{A}$ does not cheat in session $s(k)$ except with negligible probability; so, it suffices to show that when session $s(k)$ is accepting and $\mathcal{A}$ does not cheat in session $s(k)$, the hybrid is not aborted in session $s(k)$. Recall that if $\mathcal{A}$ does not cheat in an accepting session (in which S is corrupted), we have the following.

1. Let $(\hat{a}_1^S, \hat{d}_1^S), \ldots (\hat{a}_{11n}^S, \hat{d}_{11n}^S)$ be the values committed in NMCom in Stage 7. Let $I_{\rm bad} \subset [11n]$ be the set that is defined as follows: $i \in I_{\rm bad}$ if and only if
   (a) $(\hat{a}_i^S, \hat{d}_i^S)$ is not a valid decommitment of the i-th Com commitment in Stage 2-1, or
   (b) S does not execute the i-th mS-OT in Stage 3 honestly using $\hat{s}_{i,0} \parallel \hat{s}_{i,1} \parallel \hat{\tau}_i^S$ as the input and randomness, where $\hat{s}_{i,0} \parallel \hat{s}_{i,1} \parallel \hat{\tau}_i^S$ is obtained from $\hat{r}_i^S = \hat{a}_i^S \oplus b_i^S$.

   Then, it holds that $|I_{\rm bad}| < 0.1n$.
2. For each $b \in \{0,1\}$, define $\boldsymbol{\rho}_b^{\mathsf{nm}} = (\rho_{b,i}^{\mathsf{nm}})_{i \in \Delta}$ as follows: $\rho_{b,i}^{\mathsf{nm}} \overset{\rm def}{=} \beta_{b,i} \oplus \hat{s}_{i,b\oplus\alpha_i}$ if $i \notin I_{\rm bad}$ and $\rho_{b,i}^{\mathsf{nm}} \overset{\rm def}{=} \bot$ otherwise. Then, for each $b \in \{0,1\}$, ρ_b^{nm} is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i\in\Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{nm}}$ for every $i \in \Gamma_R$ or 0.15-far from any such valid codeword.

We show that the above two imply that the hybrid is not aborted at the end of the session, i.e., that both of the following hold for each $b \in \{0,1\}$.

1. $|\{i \in \Delta \text{ s.t. } \rho_{b,i}^{\mathsf{ext}} = \bot\}| < 0.1n$.
2. $\boldsymbol{\rho}_b^{\mathsf{ext}}$ is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i\in\Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{ext}}$ for every $i \in \Gamma_R$ or 0.14-far from any such valid codeword.

Fix any $b \in \{0,1\}$. First, we notice that we can obtain $|\{i \in \Delta \text{ s.t. } \rho_{b,i}^{\mathsf{ext}} = \bot\}| < 0.1n$ from $|I_{\rm bad}| < 0.1n$ since we have $\{i \in \Delta \text{ s.t. } \rho_{b,i}^{\mathsf{ext}} = \bot\} \subseteq I_{\rm bad}$ from the definitions of $\boldsymbol{\rho}_b^{\mathsf{ext}}$ and $I_{\rm bad}$. Next, we observe that $\boldsymbol{\rho}_b^{\mathsf{ext}}$ is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i\in\Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{ext}}$ for every $i \in \Gamma_R$ or 0.14-far from any such valid codeword. From the assumption that $\mathcal{A}$ does not cheat, it suffices to consider the following two cases.

Case 1. $\boldsymbol{\rho}_b^{\mathsf{nm}}$ is 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i\in\Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{nm}}$ for every $i \in \Gamma_R \cap \Delta$: In this case, $\boldsymbol{\rho}_b^{\mathsf{ext}}$ is 0.9-close to $\boldsymbol{w}$, and $w_i = \rho_{b,i}^{\mathsf{ext}}$ holds for every $i \in \Gamma_R$. This is because for every i such that $\rho_{b,i}^{\mathsf{nm}} = w_i$, we have $\rho_{b,i}^{\mathsf{nm}} \neq \bot$ and thus we have $\rho_{b,i}^{\mathsf{nm}} = \rho_{b,i}^{\mathsf{ext}}$ from the definition of $\boldsymbol{\rho}_b^{\mathsf{nm}}$ and $\boldsymbol{\rho}_b^{\mathsf{ext}}$.

Case 2. $\boldsymbol{\rho}_b^{\mathsf{nm}}$ is 0.15-far from any valid codeword $\boldsymbol{w} = (w_i)_{i\in\Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{nm}}$ for every $i \in \Gamma_R \cap \Delta$: In this case, $\boldsymbol{\rho}_b^{\mathsf{ext}}$ is 0.14-far from any valid codeword $\boldsymbol{w'}$ that satisfies $w'_i = \rho_{b,i}^{\mathsf{ext}}$ for every $i \in \Gamma_R \cap \Delta$. This can be seen by observing the following: (1) for every $i \in \Gamma_R \cap \Delta$, we have $i \notin I_{\rm bad}$ (this is because the session is accepting) and hence $\rho_{b,i}^{\mathsf{ext}} = \rho_{b,i}^{\mathsf{nm}}$; (2) therefore, for any valid codeword $\boldsymbol{w'}$ that satisfies $w'_i = \rho_{b,i}^{\mathsf{ext}}$ for every $i \in \Gamma_R \cap \Delta$, we have that $\boldsymbol{w'}$ also satisfies $w'_i = \rho^{\mathsf{nm}}_{b,i}$ for every $i \in \Gamma_R \cap \Delta$; (3) then, from the assumption of this case, $\boldsymbol{\rho}^{\mathsf{nm}}_b$ is 0.15-far from $\boldsymbol{w'}$; (4) now, since $\boldsymbol{\rho}^{\mathsf{nm}}_b$ and $\boldsymbol{\rho}^{\mathsf{ext}}_b$ are 0.99-close (this follows from $|I_{\mathrm{bad}}| < 0.1n$), $\boldsymbol{\rho}^{\mathsf{ext}}_b$ is 0.14-far from $\boldsymbol{w'}$.

We therefore conclude that when session $s(k)$ is accepting and $\mathcal{A}$ does not cheat in session $s(k)$, the hybrid is not aborted in session $s(k)$.

Next, we show that in session $s(k)$ of $H_{k:3}$, if the hybrid is not aborted, we have $\mathsf{Value}(\boldsymbol{\rho}^{\mathsf{ext}}_u, \Gamma_R \cap \Delta) = \mathsf{Value}(\widetilde{\boldsymbol{\rho}}, \Gamma_R \cap \Delta)$. To show this, it suffices to show the following two claims.

Claim 3. *For any* $\boldsymbol{x} = (x_i)_{i\in\Delta}, \boldsymbol{y} = (y_i)_{i\in\Delta}$ *and a set* Θ*, we have* $\mathsf{Value}(\boldsymbol{x}, \Theta) = \mathsf{Value}(\boldsymbol{y}, \Theta)$ *if the following conditions hold.*

1. $\boldsymbol{x}$ *and* $\boldsymbol{y}$ *are* 0.99*-close, and* $x_i = y_i$ *holds for every* $i \in \Theta$*.*
2. *If* $x_i \neq \perp$*, then* $x_i = y_i$*.*
3. x *is either* 0.9*-close to a valid codeword* $\boldsymbol{w} = (w_i)_{i\in\Delta}$ *that satisfies* $w_i = x_i$ *for every* $i \in \Theta$ *or* 0.14*-far from any such valid codeword.*

Claim 4. *In session* $s(k)$ *of* $H_{k:3}$*, if the sender* S *is corrupted, the session is accepting, and the hybrid is not aborted, the following hold.*

1. $\boldsymbol{\rho}^{\mathsf{ext}}_u$ *and* $\widetilde{\boldsymbol{\rho}}$ *are* 0.99*-close, and* $\rho^{\mathsf{ext}}_{u,i} = \tilde{\rho}_i$ *holds for every* $i \in \Gamma_R \cap \Delta$*.*
2. *If* $\rho^{\mathsf{ext}}_{u,i} \neq \perp$*, then* $\rho^{\mathsf{ext}}_{u,i} = \tilde{\rho}_i$*.*
3. $\boldsymbol{\rho}^{\mathsf{ext}}_u$ *is either* 0.9*-close to a valid codeword* $\boldsymbol{w} = (w_i)_{i\in\Delta}$ *that satisfies* $w_i = \rho^{\mathsf{ext}}_{u,i}$ *for every* $i \in \Gamma_R \cap \Delta$ *or* 0.14*-far from any such valid codeword.*

We prove each of the claims below.

Proof (of Claim 3). We consider the following two cases.

Case 1. $\boldsymbol{x}$ **is** 0.9**-close to a valid codeword** $\boldsymbol{w} = (w_i)_{i\in\Delta}$ **that satisfies** $w_i = x_i$ **for every** $i \in \Theta$**:** First, we observe that $\boldsymbol{y}$ is 0.9-close to $\boldsymbol{w}$. Since $\boldsymbol{w}$ is a valid codeword, we have $w_i \neq \perp$ for every $i \in \Delta$; thus, for every i such that $x_i = w_i$, we have $x_i \neq \perp$. Recall that from the assumed conditions, for every i such that $x_i \neq \perp$, we have $x_i = y_i$. Therefore, for every i such that $x_i = w_i$, we have $y_i = w_i$, which implies that $\boldsymbol{y}$ is 0.9-close to $\boldsymbol{w}$.

Next, we observe that $\boldsymbol{w}$ satisfies $w_i = y_i$ for every $i \in \Theta$. From the assumed conditions, we have $x_i = y_i$ for every $i \in \Theta$. Also, from the condition of this case, $\boldsymbol{w}$ satisfies $w_i = x_i$ for every $i \in \Theta$. From these two, we have that $\boldsymbol{w}$ satisfies $w_i = y_i$ for every $i \in \Theta$.

Now, from the definition of $\mathsf{Value}(\cdot,\cdot)$, we have $\mathsf{Value}(\boldsymbol{x}, \Theta) = \mathsf{Value}(\boldsymbol{y}, \Theta) = \mathsf{Decode}(\boldsymbol{w})$.

Case 2. $\boldsymbol{x}$ **is** 0.14**-far from any valid codeword** $\boldsymbol{w} = (w_i)_{i\in\Delta}$ **that satisfies** $w_i = x_i$ **for every** $i \in \Theta$**:** For any valid codeword $\boldsymbol{w'} = (w'_i)_{i\in\Delta}$ that satisfies $w'_i = y_i$ for every $i \in \Theta$, we observe that $\boldsymbol{y}$ is 0.1-far from $\boldsymbol{w'}$. Since we assume that $x_i = y_i$ holds for every $i \in \Theta$, we have $w'_i = x_i$ for every $i \in \Theta$. Therefore, from the assumption of this case, $\boldsymbol{x}$ is 0.14-far from $\boldsymbol{w'}$. Now, since we assume that $\boldsymbol{x}$ and $\boldsymbol{y}$ are 0.99-close, $\boldsymbol{y}$ is 0.1-far from $\boldsymbol{w'}$.

Now, from the definition of $\mathsf{Value}(\cdot,\cdot)$, we conclude that $\mathsf{Value}(\boldsymbol{x}, \Theta) = \mathsf{Value}(\boldsymbol{y}, \Theta) = \perp$.

Notice that from the assumed conditions, either Case 1 or Case 2 is true. This concludes the proof of Claim 3. □

Proof (of Claim 4). Recall that if the hybrid is not aborted in an accepting session in which S is corrupted, we have the following for each $b \in \{0,1\}$ in that session.

1. $|\{i \in \Delta \text{ s.t. } \rho^{\mathsf{ext}}_{b,i} = \bot\}| < 0.1n$.
2. $\boldsymbol{\rho}^{\mathsf{ext}}_{b}$ is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i\in\Delta}$ that satisfies $w_i = \rho^{\mathsf{ext}}_{b,i}$ for every $i \in \Gamma_R$ or 0.14-far from any such valid codeword.

Thus, it suffices to show that the above two imply the first conditions in the claim statement.

First, we show that $\boldsymbol{\rho}^{\mathsf{ext}}_{u}$ and $\widetilde{\boldsymbol{\rho}}$ are 0.99-close and that $\rho^{\mathsf{ext}}_{u,i} = \tilde{\rho}_i$ holds for every $i \in \Gamma_R \cap \Delta$. From the definition of $\boldsymbol{\rho}^{\mathsf{ext}}_{u}$, we have $\rho^{\mathsf{ext}}_{u,i} = \tilde{\rho}_i$ for every i such that $\rho^{\mathsf{ext}}_{b,i} \neq \bot$ (this is because for every such i, $\mathcal{A}$ executed the i-th mS-OT in Stage 3 honestly using the coin obtained in Stage 2-1, which implies that the value $\widetilde{s}_i$ that was obtained from the i-th mS-OT is equal to the value s_{i,c_i} that was obtained by extracting the coin in Stage 2-1 by brute-force). Then, since $|\{i \in \Delta \text{ s.t. } \rho^{\mathsf{ext}}_{b,i} = \bot\}| < 0.1n$ and $\{i \in \Delta \text{ s.t. } \rho^{\mathsf{ext}}_{b,i} = \bot\} \cap \Gamma_R = \emptyset$ (the latter holds since the session would be rejected otherwise), we have that $\boldsymbol{\rho}^{\mathsf{ext}}_{u}$ and $\widetilde{\boldsymbol{\rho}}$ are 0.99-close and that $\rho^{\mathsf{ext}}_{u,i} = \tilde{\rho}_i$ holds for every $i \in \Gamma_R \cap \Delta$.

Next, we show that if $\rho^{\mathsf{ext}}_{u,i} \neq \bot$ then $\rho^{\mathsf{ext}}_{u,i} = \tilde{\rho}_i$. From the definition of $\boldsymbol{\rho}^{\mathsf{ext}}_{u}$, if $\rho^{\mathsf{ext}}_{u,i} \neq \bot$, $\mathcal{A}$ executed the i-th mS-OT in Stage 3 honestly using the coin obtained in Stage 2-1, so we have $\rho^{\mathsf{ext}}_{u,i} = \tilde{\rho}_i$ from the argument same as above.

This concludes the proof of Claim 4. □

This concludes the proof of Lemma 4. □

Lemma 5. *Assume that in $H_{k:3}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:3}$ and $H_{k:4}$ are indistinguishable, and*
- *in $H_{k:4}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Recall that $H_{k:3}$ and $H_{k:4}$ differ only in that in session $s(k)$ of $H_{k:4}$, if S is corrupted and SM_k is $\mathsf{third\ special\ message}$, α_i is a random bit rather than $\alpha_i = u \oplus c_i$ for every $i \in \Delta$ in Stage 6-1. Intuitively, we can prove this lemma by using the security of $\mathsf{mS\text{-}OT}$: For every $i \notin \Gamma_S$, the choice bit c_i of the i-th mS-OT in Stage 3 is hidden from $\mathcal{A}$ and hence $\alpha_i = u \oplus c_i$ in $H_{k:3}$ is indistinguishable from a random bit. Formally, we prove this Lemma in the same way as we do for Claim 1 (we use the security of $\mathsf{mS\text{-}OT}$ rather than the hiding of ExtCom); the proof is given in Appendix D.

Lemma 6. *Assume that in $H_{k:4}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:4}$ and $H_{k:5}$ are indistinguishable, and*

- *in $H_{k:5}$, $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability.*

Since hybrids $H_{k:4}, H_{k:5}$ differ only in the values committed to in NMCom and ExtCom for the indices outside of Γ_R, this lemma can be proven identically with Lemma 2. For completeness, we give a formal proof in Appendix D.

Lemma 7. *Assume that in $H_{k:5}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:5}$ and $H_{k:6}$ are indistinguishable, and*
- *in $H_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability.*

Since hybrids $H_{k:5}, H_{k:6}$ differ only in the inputs and the randomness that are used in some of the mS-OTs in Stage 3, this lemma can be proven identically with Lemma 3 (which in turn can be proven quite similarly to Lemma 2). For completeness, we give a formal proof in Appendix D.

Lemma 8. *Assume that in $H_{k:6}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:6}$ and $H_{k:7}$ are indistinguishable, and*
- *in $H_{k:7}$, $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability.*

Proof. We prove the lemma by considering the following intermediate hybrids $H'_{k:6}$, $H''_{k:6}$, and $H'''_{k:6}$.

Hybrid $H'_{k:6}$. $H'_{k:6}$ is the same as $H_{k:6}$ except that in session $s(k)$, if R is corrupted and SM_k is $\mathsf{fourth\ special\ message}$, the following modifications are made.

1. As in $H_{k:7}$, the committed strings $\boldsymbol{a}^R = (a_1^R, \dots, a_{11n}^R)$ are extracted by brute force in Stage 2-2, $\boldsymbol{r}^R = (r_1^R, \dots, r_{11n}^R)$ is defined by $r_i^R \overset{\text{def}}{=} a_i^R \oplus b_i^R$ for each $i \in [11n]$, and r_i^R is parsed as $c_i \parallel \tau_i^R$ for each $i \in [11n]$. Also, I_0, I_1, and u^* are defined as in $H_{k:7}$.
2. In Stage 6, $\beta_{b,i}$ is a random bit rather than $\beta_{b,i} = \rho_{b,i} \oplus s_{i,b\oplus\alpha_i}$ for every $b \in \{0,1\}$ and $i \in \Delta \setminus I_b$. (Recall that, roughly, $I_b \subset \Delta$ is the set of indices on which $\mathcal{A}$ could have obtained $s_{i,b\oplus\alpha_i}$.)

Hybrid $H''_{k:6}$. $H''_{k:6}$ is the same as $H'_{k:6}$ except that in session $s(k)$, if R is corrupted and SM_k is $\mathsf{fourth\ special\ message}$, the following modification is made.

1. The execution of the hybrid is aborted if both of $|I_0| \geq 6n+1$ and $|I_1| \geq 6n+1$ holds.
2. In Stage 6, $\boldsymbol{\rho}_{1-u^*} = \{\rho_{1-u^*,i}\}_{i\in\Delta}$ is a secret sharing of a random bit rather than that of v_{1-u^*}.

Hybrid $H'''_{k:6}$. $H'''_{k:6}$ is the same as $H''_{k:6}$ except that in session $s(k)$, if R is corrupted and SM_k is $\mathsf{fourth\ special\ message}$, the following modification is made.

1. In Stage 6, $\beta_{b,i}$ is $\beta_{b,i} = \rho_{b,i} \oplus s_{i,b\oplus\alpha_i}$ rather than a random bit for every $b \in \{0,1\}$ and $i \in \Delta \setminus I_b$.

Notice that $H'''_{k:6}$ is identical with $H_{k:7}$.

Claim 5. *Assume that in $H_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:6}$ and $H'_{k:6}$ are indistinguishable, and*
- *in $H'_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Recall that $H_{k:6}$ and $H'_{k:6}$ differ only in that in session $s(k)$ of $H'_{k:6}$, if R is corrupted and SM_k is $\mathsf{fourth\ special\ message}$, $\beta_{b,i}$ is a random bit rather than $\beta_{b,i} = \rho_{b,i} \oplus s_{i,b\oplus\alpha_i}$ for every $b \in \{0,1\}$ and $i \in \Delta \setminus I_b$. Intuitively, we can prove this claim by using the security of mS-OT: For every $i \in \Delta \setminus I_b$, $\mathcal{A}$ executed the i-th mS-OT honestly with choice bit $(1-b) \oplus \alpha_i$, and the sender's input and randomness of this mS-OT are not revealed in Stage 8; therefore, the value of $s_{i,b\oplus\alpha_i}$ is hidden from $\mathcal{A}$ and thus $\beta_{b,i} = \rho_{b,i} \oplus s_{i,b\oplus\alpha_i}$ is indistinguishable from a random bit. Formally, we prove this claim in the same way as we do for Claim 1 (we use the security of mS-OT rather than the hiding of ExtCom); a formal proof is given in Appendix D.

Claim 6. *Assume that in $H'_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H'_{k:6}$ and $H''_{k:6}$ are indistinguishable, and*
- *in $H''_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Proof. Recall that hybrid $H''_{k:6}$ differ from $H'_{k:6}$ in that in Stage 6 of session $s(k)$, either the hybrid is aborted or $\boldsymbol{\rho}_{1-u^*} = \{\rho_{1-u^*,i}\}_{i\in\Delta}$ is a secret sharing of a random bit rather than that of v_{1-u^*}.

For proving the lemma, it suffices to show that in session $s(k)$ of $H''_{k:6}$, the hybrid is not aborted (i.e., we have $|I_0| \le 6n$ or $|I_1| \le 6n$) except with negligible probability. To see that showing this is indeed sufficient for proving the lemma, observe the following. First, if the hybrid is not aborted, we have $|I_{1-u^*}| \le 6n$, so $\beta_{1-u^*,i}$ is a random bit on at least $4n$ indices and thus $\rho_{1-u^*,i}$ is hidden on at least $4n$ indices, which implies that $H'_{k:6}$ and $H''_{k:6}$ are statistically indistinguishable. Second, since $H'_{k:6}$ and $H''_{k:6}$ proceed identically until the beginning of Stage 6-2 of session $s(k)$, and

1. if the experiment is not aborted in session $s(k)$, $H'_{k:6}$ and $H''_{k:6}$ continue to proceed identically after Stage 6-2 of session $s(k)$, and
2. if the hybrid is aborted in session $s(k)$, $\mathcal{A}$ clearly does not cheat in any session after Stage 6-2 of session $s(k)$,

the probability that $\mathcal{A}$ cheat in sessions $s(k), \ldots, s(4m)$ is not increased in $H''_{k:6}$.

Hence, we show that in session $s(k)$ of $H''_{k:7}$, the hybrid is not aborted in except with negligible probability, or equivalently, that we have $|I_0| \leq 6n$ or $|I_1| \leq 6n$ except with negligible probability. Since $H''_{k:7}$ proceeds identically with $H'_{k:7}$ until Stage 6-2 of session $s(k)$, we have that $\mathcal{A}$ does not cheat in session $s(k)$ of $H''_{k:7}$ except with negligible probability, so it suffices to show that in session $s(k)$ of $H''_{k:7}$, we have either $|I_0| \leq 6n$ or $|I_1| \leq 6n$ whenever $\mathcal{A}$ does not cheat. Assume that $\mathcal{A}$ does not cheat in session $s(k)$ of $H''_{k:7}$. Then, since $|\Gamma_R| = n$ and that the number of indices on which $\mathcal{A}$ does not execute mS-OT using the outcome of coin-tossing is at most n, we have $|I_0 \cap I_1| \leq 2n$. Now, since $I_0, I_1 \subset \Delta$ and thus $|I_0 \cup I_1| \leq |\Delta| = 10n$, we have $|I_0| + |I_1| \leq 12n$, and hence, we have either $|I_0| \leq 6n$ or $|I_1| \leq 6n$. □

Claim 7. *Assume that in $H''_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H''_{k:6}$ and $H'''_{k:6}$ are indistinguishable, and*
- *in $H'''_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Proof. This claim can be proven identically with Claim 5. □

This completes the proof of Lemma 8. □

From Lemmas 2–8, we conclude that the output of H_0 and that of $H_{4m:7}$ are indistinguishable, i.e., the output of the real world and that of the ideal world are indistinguishable. This concludes the proof of Theorem 2.

5 Our SPS Concurrent MPC Protocol

In this section, we prove the following theorem.

Theorem 3. *Assume the existence of constant-round semi-honest oblivious transfer protocols and collision-resistant hash functions. Let $\mathcal{F}$ be any well-formed functionality and $\hat{\mathcal{F}}$ be its multi-session extension. Then, there exists a constant-round protocol that UC-SPS realizes $\hat{\mathcal{F}}$, and it uses the underlying primitives in the black-box way.*

We focus on the two-party case below (the MPC case is analogues).

Protocol Description. Roughly speaking, we obtain our SPS 2PC protocol by composing our SPS OT protocol in Section 4 with a UC-secure OT-hybrid 2PC protocol. Concretely, let Π_{OT} be our SPS OT protocol in Section 4, and $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ be a UC-secure OT-hybrid 2PC protocol with the following property: The two parties use the OT functionality $\mathcal{F}_{\mathrm{OT}}$ only at the beginning of the protocol, and they send only randomly chosen inputs to $\mathcal{F}_{OT}$. Then, we obtain our SPS 2PC protocol Π_{2PC} by replacing each invocation of $\mathcal{F}_{\mathrm{OT}}$ in $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ with an execution of Π_{OT} (i.e., the two parties execute Π_{OT} instead of calling to $\mathcal{F}_{OT}$), where all the executions of Π_{OT} are carried out in a synchronous manner, i.e., in a manner that the first message of all the executions are sent before the second message of any execution is sent etc.

As the UC-secure OT-hybrid 2PC protocol, we use the constant-round 2PC (actually, MPC) protocol of Ishai et al. [28], which makes only black-box use of pseudorandom generators (which in turn can be obtained in the black-box way from any semi-honest OT protocol). (The protocol of Ishai et al. [28] itself does not satisfy the above property, but it can be modified to satisfy it; see Appendix E.) Since the OT-hybrid protocol of Ishai et al. [28] is a black-box construction and has only constant number of rounds, our protocol Π_{2PC} is also a black-box construction and has only constant number of rounds.

Simulator $\mathcal{S}im$. As in Section 4.2, we consider a simulator that works against adversaries that participate in multiple sessions of Π_{2PC}. Let $\mathcal{Z}$ be any environment, $\mathcal{A}$ be any adversary that participates in multiple sessions of Π_{2PC}. Our simulator $\mathcal{S}im_{\mathrm{OT}}$ internally invokes the adversary $\mathcal{A}$, and simulates each of the sessions by using the simulator of Π_{OT} (Section 4.2) and that of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ as follows.

1. In each execution of Π_{OT} at the beginning of Π_{2PC}, $\mathcal{S}im$ simulates the honest party's messages for $\mathcal{A}$ in the same way as $\mathcal{S}im_{\mathrm{OT}}$.
   Recall that $\mathcal{S}im_{\mathrm{OT}}$ makes a query to $\mathcal{F}_{OT}$ during the simulation. When $\mathcal{S}im_{\mathrm{OT}}$ makes a query to $\mathcal{F}_{OT}$, $\mathcal{S}im$ sends those queries to the simulator of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ in order to simulate the answer from $\mathcal{F}_{OT}$. (Recall that the simulator of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ simulates $\mathcal{F}_{OT}$ for the adversary.)
2. In the execution of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ during Π_{2PC}, $\mathcal{S}im$ simulates the honest party's messages for $\mathcal{A}$ by using the simulator of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$, who obtained the queries to $\mathcal{F}_{OT}$ as above.

We remark that here we use the simulator of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ in the setting where multiple sessions of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ are concurrently executed and some super-polynomial-time computation is performed. However, the use of it in this setting does not cause any problem because it runs in the black-box straight-line manner.

Proof of Indistinguishability. We show that the output of the environment in the real world and that in the ideal world are indistinguishable. The proof proceeds very similarly to the proof for our SPS OT protocol (Section 4). To simplify the exposition, below we assume that $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ makes only a single call to $\mathcal{F}_{OT}$. (The proof can be modified straightforwardly when $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ makes multiple calls to $\mathcal{F}_{OT}$.)

Recall that Π_{2PC} is obtained by composing our OT protocol Π_{OT} with a OT-hybrid 2PC protocol $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$. Roughly, we consider a sequence of hybrid experiments in which:

- Each execution of Π_{OT} is gradually changed to simulation as in the sequence of hybrid experiments that we considered in the proof of Π_{OT} (Section 4.3).

- Once the execution of Π_{OT} in a session of Π_{2PC} is changed to simulation completely, the execution of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ in that session is changed to simulation.

More concretely, we consider hybrids H_0 and $H_{k:1}, \dots, H_{k:9}$ ($k \in [4m]$), where H_0 and $H_{k:1}, \dots, H_{k:7}$ are defined as in Section 4.3, and $H_{k:8}$ and $H_{k:9}$ are defined as follows.

Hybrid $H_{k:8}$. $H_{k:8}$ is the same as $H_{k:7}$ except that in session $s(k)$, if S is corrupted and SM_k is $\mathsf{third\ special\ message}$, all the messages of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ from R are generated by the simulator of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$. More concretely, the messages of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ from R are generated as follows. Recall that from the definition of Hybrid $H_{k:3}$, the implicit input $v_b^* \stackrel{\mathrm{def}}{=} \mathsf{Value}(\boldsymbol{\rho}_b^{\mathsf{ext}}, \Gamma_R \cap \Delta)$ ($b \in \{0,1\}$) to Π_{OT} is extracted from the adversary in session $s(k)$ (as $\boldsymbol{\rho}_b^{\mathsf{ext}}$ are computed for both $b \in \{0,1\}$). Now, the messages of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ from R are simulated by feeding those extracted implicit input and the subsequent messages to the simulator of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$.

Hybrid $H_{k:9}$. $H_{k:9}$ is the same as $H_{k:8}$ except that in session $s(k)$, if R is corrupted and SM_k is $\mathsf{fourth\ special\ message}$, all the messages of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ from S are generated by the simulator of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$.

Lemma 9. *Assume that in $H_{k:7}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:7}$ and $H_{k:8}$ are indistinguishable, and*
- *in $H_{k:8}$, $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability.*

Lemma 10. *Assume that in $H_{k:8}$ ($k \in [4m]$), $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability. Then,*

- *$H_{k:8}$ and $H_{k:9}$ are indistinguishable, and*
- *in $H_{k:9}$, $\mathcal{A}$ does not cheat in sessions $s(k), \dots, s(4m)$ except with negligible probability.*

Lemma 10 can be proven identically with Lemma 9, and Lemma 9 can be proven quite similarly to Claim 1 (Section 4.3); the only difference is that we use the security of $\Pi_{\mathrm{2PC}}^{\mathcal{F}_{\mathrm{OT}}}$ rather than the hiding of ExtCom. We give a proof of Lemma 9 in Appendix D.

By combining Lemmas 9 and 10 with Lemmas 2–8 in Section 4.3, we conclude that the output of H_0 and that of $H_{4m:9}$ are indistinguishable, i.e., the output of the real world and that of the ideal world are indistinguishable. This concludes the proof of Theorem 3.

References

1. Badrinarayanan, S., Goyal, V., Jain, A., Khurana, D., Sahai, A.: Round optimal concurrent MPC via strong simulation. Cryptology ePrint Archive, Report 2017/597 (2017), http://eprint.iacr.org/2017/597

2. Barak, B., Sahai, A.: How to play almost any mental game over the net - Concurrent composition via super-polynomial simulation. In: 46th FOCS. pp. 543–552. IEEE Computer Society Press (Oct 2005)
3. Broadnax, B., Döttling, N., Hartung, G., Müller-Quade, J., Nagel, M.: Concurrently composable security with shielded super-polynomial simulators. In: Coron, J., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part I. LNCS, vol. 10210, pp. 351–381. Springer, Heidelberg (May 2017)
4. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: 42nd FOCS. pp. 136–145. IEEE Computer Society Press (Oct 2001)
5. Canetti, R., Kushilevitz, E., Lindell, Y.: On the limitations of universally composable two-party computation without set-up assumptions. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 68–86. Springer, Heidelberg (May 2003)
6. Canetti, R., Lin, H., Pass, R.: Adaptive hardness and composable security in the plain model from standard assumptions. In: 51st FOCS. pp. 541–550. IEEE Computer Society Press (Oct 2010)
7. Canetti, R., Lin, H., Pass, R.: Adaptive hardness and composable security in the plain model from standard assumptions. SIAM Journal on Computing 45(5), 1793–1834 (2016)
8. Canetti, R., Lindell, Y., Ostrovsky, R., Sahai, A.: Universally composable two-party and multi-party secure computation. In: 34th ACM STOC. pp. 494–503. ACM Press (May 2002)
9. Choi, S.G., Dachman-Soled, D., Malkin, T., Wee, H.: A black-box construction of non-malleable encryption from semantically secure encryption. Journal of Cryptology (Mar 2017)
10. Cramer, R., Hanaoka, G., Hofheinz, D., Imai, H., Kiltz, E., Pass, R., shelat, a., Vaikuntanathan, V.: Bounded CCA2-secure encryption. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 502–518. Springer, Heidelberg (Dec 2007)
11. Dachman-Soled, D., Malkin, T., Raykova, M., Venkitasubramaniam, M.: Adaptive and concurrent secure computation from new adaptive, non-malleable commitments. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 316–336. Springer, Heidelberg (Dec 2013)
12. Damgård, I., Pedersen, T.P., Pfitzmann, B.: Statistical secrecy and multibit commitments. IEEE Transactions on Information Theory 44(3), 1143–1151 (1998)
13. Feige, U., Shamir, A.: Witness indistinguishable and witness hiding protocols. In: 22nd ACM STOC. pp. 416–426. ACM Press (May 1990)
14. Garay, J.A., MacKenzie, P.D.: Concurrent oblivious transfer. In: 41st FOCS. pp. 314–324. IEEE Computer Society Press (Nov 2000)
15. 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)
16. Garg, S., Kumarasubramanian, A., Ostrovsky, R., Visconti, I.: Impossibility results for static input secure computation. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 424–442. Springer, Heidelberg (Aug 2012)
17. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game or A completeness theorem for protocols with honest majority. In: Aho, A. (ed.) 19th ACM STOC. pp. 218–229. ACM Press (May 1987)
18. 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)

19. Goyal, V., Jain, A.: On concurrently secure computation in the multiple ideal query model. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 684–701. Springer, Heidelberg (May 2013)
20. 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)
21. 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)
22. 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)
23. 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)
24. 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)
25. Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM Journal on Computing 28(4), 1364–1396 (1999)
26. Hazay, C., Venkitasubramaniam, M.: Composable adaptive secure protocols without setup under polytime assumptions. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part I. LNCS, vol. 9985, pp. 400–432. Springer, Heidelberg (Oct / Nov 2016)
27. 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)
28. 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)
29. 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)
30. Kiyoshima, S., Manabe, Y., Okamoto, T.: Constant-round black-box construction of composable multi-party computation protocol. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 343–367. Springer, Heidelberg (Feb 2014)
31. Lin, H., Pass, R.: Non-malleability amplification. In: Mitzenmacher, M. (ed.) 41st ACM STOC. pp. 189–198. ACM Press (May / Jun 2009)
32. Lin, H., Pass, R.: Black-box constructions of composable protocols without setup. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 461–478. Springer, Heidelberg (Aug 2012)
33. Lin, H., Pass, R., Venkitasubramaniam, M.: Concurrent non-malleable commitments from any one-way function. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 571–588. Springer, Heidelberg (Mar 2008)
34. Lin, H., Pass, R., Venkitasubramaniam, M.: A unified framework for concurrent security: universal composability from stand-alone non-malleability. In: Mitzenmacher, M. (ed.) 41st ACM STOC. pp. 179–188. ACM Press (May / Jun 2009)
35. Lindell, Y.: Bounded-concurrent secure two-party computation without setup assumptions. In: 35th ACM STOC. pp. 683–692. ACM Press (Jun 2003)
36. Lindell, Y.: Lower bounds for concurrent self composition. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 203–222. Springer, Heidelberg (Feb 2004)

37. Malkin, T., Moriarty, R., Yakovenko, N.: Generalized environmental security from number theoretic assumptions. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 343–359. Springer, Heidelberg (Mar 2006)
38. Micali, S., Pass, R., Rosen, A.: Input-indistinguishable computation. In: 47th FOCS. pp. 367–378. IEEE Computer Society Press (Oct 2006)
39. Naor, M.: Bit commitment using pseudorandomness. Journal of Cryptology 4(2), 151–158 (1991)
40. Naor, M., Yung, M.: Universal one-way hash functions and their cryptographic applications. In: 21st ACM STOC. pp. 33–43. ACM Press (May 1989)
41. Ostrovsky, R., Richelson, S., Scafuro, A.: Round-optimal black-box two-party computation. In: Gennaro, R., Robshaw, M.J.B. (eds.) CRYPTO 2015, Part II. LNCS, vol. 9216, pp. 339–358. Springer, Heidelberg (Aug 2015)
42. Ostrovsky, R., Scafuro, A., Venkitasubramaniam, M.: Resettably sound zero-knowledge arguments from OWFs - the (semi) black-box way. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part I. LNCS, vol. 9014, pp. 345–374. Springer, Heidelberg (Mar 2015)
43. Pass, R.: Simulation in quasi-polynomial time, and its application to protocol composition. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 160–176. Springer, Heidelberg (May 2003)
44. Pass, R.: Bounded-concurrent secure multi-party computation with a dishonest majority. In: Babai, L. (ed.) 36th ACM STOC. pp. 232–241. ACM Press (Jun 2004)
45. Pass, R., Lin, H., Venkitasubramaniam, M.: A unified framework for UC from only OT. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 699–717. Springer, Heidelberg (Dec 2012)
46. 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)
47. Peikert, C., Waters, B.: Lossy trapdoor functions and their applications. SIAM Journal on Computing 40(6), 1803–1844 (2011)
48. Prabhakaran, M., Sahai, A.: New notions of security: Achieving universal composability without trusted setup. In: Babai, L. (ed.) 36th ACM STOC. pp. 242–251. ACM Press (Jun 2004)
49. Venkitasubramaniam, M.: On adaptively secure protocols. In: Abdalla, M., Prisco, R.D. (eds.) SCN 14. LNCS, vol. 8642, pp. 455–475. Springer, Heidelberg (Sep 2014)
50. Wee, H.: Black-box, round-efficient secure computation via non-malleability amplification. In: 51st FOCS. pp. 531–540. IEEE Computer Society Press (Oct 2010)
51. Yao, A.C.C.: How to generate and exchange secrets (extended abstract). In: 27th FOCS. pp. 162–167. IEEE Computer Society Press (Oct 1986)

A Additional Preliminaries

A.1 Shamir's Secret Sharing

We first recall Shamir's secret sharing scheme. (In this paper, we use only the $(6n+1)$-out-of-$10n$ version of it.) To compute a $(6n+1)$-out-of-$10n$ secret sharing $\boldsymbol{s} = (s_1, \ldots, s_{10n})$ of a value $v \in GF(2^n)$, we choose random $a_1, \ldots, a_{6n} \in GF(2^n)$, let $p(z) \stackrel{\text{def}}{=} v + a_1 z + \cdots + a_{6n} z^{6n}$, and set $s_i := p(i)$ for each $i \in [10n]$. Given $\boldsymbol{s}$, we can recover v by obtaining polynomial $p(\cdot)$ thorough interpolation and then computing $p(0)$. We use $\mathsf{Decode}(\cdot)$ to denote the function that recovers v from $\boldsymbol{s}$ as above.

For any positive real number $x \leq 1$ and any $\boldsymbol{s} = (s_1, \ldots, s_{10n})$ and $\boldsymbol{s'} = (s'_1, \ldots, s'_{10n})$, we say that $\boldsymbol{s}$ and $\boldsymbol{s'}$ are *x-close* if $|\{i \in [10n] \text{ s.t. } s_i = s'_i\}| \geq x \cdot 10n$. If $\boldsymbol{s}$ and $\boldsymbol{s'}$ are not x-close, we say that they are *$(1-x)$-far*. Since the shares generated by $(6n+1)$-out-of-$10n$ Shamir's secret sharing scheme are actually a codeword of the Reed-Solomon code with minimum relative distance 0.4, if a (possibly incorrectly generated) sharing $\boldsymbol{s}$ is 0.8-close to a valid codeword $\boldsymbol{w}$, we can recover $\boldsymbol{w}$ from $\boldsymbol{s}$ efficiently by using, for example, the Berlekamp-Welch algorithm.

A.2 Commitment Schemes

Recall that a commitment scheme is a two-party protocol between a *committer* and a *receiver*. We say that a commitment is *accepting* if the receiver does not abort in the commit phase, and *valid* if there exists a value to which the commitment can be decommitted (i.e., if there exists a decommitment that the verifier accepts in the decommit phase). The *committed value* of a commitment is the value to which the commitment can be decommitted. We define the committed value of an invalid commitment as $\perp$.

There exists a two-round statistically binding commitment scheme Com based on one-way functions [39,25], and it uses the underlying one-way function in a black-box way.

A.3 Extractable Commitment Schemes

We next recall the definition of *extractable commitment schemes* from [46]. Roughly speaking, a commitment scheme is *extractable* if there exists an expected polynomial-time oracle machine, an *extractor*, E such that for any adversarial committer C^* that gives a commitment to honest receiver, the extractor E^{C^*} extracts the committed value of the commitment from C^* as long as the commitment is valid. We note that when the commitment is invalid, E can output an arbitrary garbage value; this is called *over-extraction*.

Formally, extractable commitment schemes are defined as follows. A commitment scheme $\langle C, R \rangle$ is *extractable* if there exists an expected polynomial-time extractor E such that for any PPT committer C^*, the extractor E^{C^*} outputs a pair (τ, σ) that satisfies the following properties.

- τ is identically distributed with the view of C^* that interacts with an honest receiver R in the commit phase of $\langle C, R \rangle$. Let c_τ be the commitment that C^* gives in τ.
- If c_τ is accepting, then $\sigma \neq \perp$ except with negligible probability.
- If $\sigma \neq \perp$, then it is statistically impossible to decommit c_τ to any value other than σ.

There exists a four-round extractable commitment scheme ExtCom based on one-way functions [46], and it uses the underlying one-way function in a black-box way. Furthermore, ExtCom satisfies extractability in a stronger sense: It is extractable even against adversarial committers that give polynomially many ExtCom commitments *in parallel.* (The extractor outputs $(\tau, \sigma_1, \sigma_2, \ldots)$ for such committers.)

B Robust Parallel Non-malleable Commitment

We show that any parallel non-malleable commitment can be transformed into a parallel k-robust non-malleable commitment for any constant k. If the original parallel non-malleable commitment is a black-box construction and has a constant number of rounds, the resultant parallel k-robust one is also a black-box construction and has a constant number of rounds.

Theorem 4. *Assume the existence of collision-resistant hash functions and a r-round parallel non-malleable commitment scheme. Then, for any $k \in \mathbb{N}$ there exists a $O(r + k)$-round parallel k-robust non-malleable commitment scheme, and it uses the underlying collision-resistant hash functions and non-malleable commitment schemes in the black-box way.*

B.1 Protocol Description

In the protocol, we use the following building blocks.

- Any parallel non-malleable commitment scheme NMCom.
- The four-round statistically hiding extractable commitment scheme $\mathsf{ExtCom}_{\mathrm{SH}}$ described in Figure 3.[11] We remark that the extractor for $\mathsf{ExtCom}_{\mathrm{SH}}$ obtains the committed value of a commitment by rewinding the committer and sending new receiver challenge to it repeatedly.

Our parallel k-robust non-malleable commitment is described below.

Inputs: Common inputs 1^n and $\mathsf{id} \in \{0,1\}^n$ are given to the committer C and the receiver R. The committer C also takes a secret input $v \in \{0,1\}^n$.

Stage 1: R chooses a random subset $\Gamma \subset [10n]$ of size n and commits to it by using $\mathsf{ExtCom}_{\mathrm{SH}}$.

Stage 2: C computes a $(n+1)$-out-of-$10n$ secret sharing of v, denoted by $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_{10n})$. Then, for each $i \in [10n]$ in parallel, C commits to ρ_i by using NMCom with tag id. Let $d_1, \ldots, d_{10n}$ be the decommitments.

Stage 3: For each $i \in [10n]$ in parallel, C commits to (ρ_i, d_i) by using NMCom with tag id.

Stage 4: For each $j \in [k+1]$ in sequence, C does the following.

- For each $i \in [10n]$ in parallel, C commits to (ρ_i, d_i) by using $\mathsf{ExtCom}_{\mathrm{SH}}$.

[11] $\mathsf{ExtCom}_{\mathrm{SH}}$ satisfies *extractability w.r.t. opening*, which guarantees that any committer cannot open a commitment to a value that is different from the extracted value.

Let $\mathsf{Com}_{\mathrm{SH}}$ be a two-round statistically hiding commitment scheme, which can be obtained from collision-resistant hash functions in the black-box way [40,12].

Commit Phase The committer C and the receiver R receive common inputs 1^n. To commit to $v \in \{0,1\}^n$, the committer C does the following with the receiver R.

`commit` **stage.**
For each $i \in [n]$, the committer C chooses a pair of random n-bit strings (a_i^0, a_i^1) such that $a_i^0 \oplus a_i^1 = v$. Then, for each $i \in [n]$ in parallel, C commits to a_i^0 and a_i^1 by using $\mathsf{Com}_{\mathrm{SH}}$. For each $i \in [n]$ and $b \in \{0,1\}$, let c_i^b be the commitment to a_i^b.

`challenge` **stage.**
R sends random n-bit string $e = (e_1, \ldots, e_n)$ to C.

`reply` **stage.**
For each $i \in [n]$, C decommits $c_i^{e_i}$ to $a_i^{e_i}$.

Decommit Phase C sends v to R and decommits c_i^b to a_i^b for all $i \in [n]$ and $b \in \{0,1\}$. R checks whether $a_1^0 \oplus a_1^1 = \cdots = a_n^0 \oplus a_n^1 = v$.

Fig. 3. The statistically hiding extractable commitment scheme ExtCom.

For each $j \in [k+1]$, we call the j-th parallel $\mathsf{ExtCom}_{\mathrm{SH}}$ commitments *the j-th $\mathsf{ExtCom}_{\mathrm{SH}}$ row.*

Stage 5:

1. R reveals Γ by decommitting the $\mathsf{ExtCom}_{\mathrm{SH}}$ commitment in Stage 1.
2. For every $i \in \Gamma$, C reveals (ρ_i, d_i) by decommitting the i-th NMCom commitment in Stage 3.
3. For every $i \in \Gamma$ and $j \in [k+1]$, C decommits the i-th commitment in the j-th $\mathsf{ExtCom}_{\mathrm{SH}}$ row to (ρ_i, d_i).
4. For every $i \in \Gamma$, R checks whether (ρ_i, d_i) is a valid decommitment of the i-th NMCom commitment in Stage 2.

Decommit Phase:

1. C sends v to R, and also reveals $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_{10n})$ by decommitting the NMCom commitments in Stage 2.
2. R accepts the decommitment if and only if $\mathsf{Value}(\boldsymbol{\rho}, \Gamma)$ is equal to v, where $\mathsf{Value}(\cdot, \cdot)$ is the function that is defined in Fig. 2 (Section 4.1).

We remark that from the definition of the decommitment phase, the committed value of our robust parallel non-malleable commitment is defined as follows.

Definition 8. *Let $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_{10n})$ be the shares that are committed in the NMCom commitment in Stage 2. Then, the committed value of the above scheme is $\mathsf{Value}(\boldsymbol{\rho}, \Gamma)$, where Γ is the subset that is revealed in Stage 5.*

B.2 Proof of Parallel Non-malleability

We first show that the commitment scheme in Appendix B.1 is parallel non-malleable. For simplicity, we prove only standard non-malleability below; parallel non-malleability can be proven analogously.

For any man-in-the-middle adversary $\mathcal{M}$, we consider a sequence of hybrid experiments in which the non-malleability experiment is gradually modified so that the commitment in the left interaction does not contain any information about the committed value in the last hybrid.

Hybrid H_0. H_0 is the same as the real non-malleability experiment (see Section 3.1). Recall that the output of the experiment is the view of $\mathcal{M}$ and the value that it committed in the right interaction.

Hybrid H_1. H_1 is the same as H_0 except that the committed value of the right interaction is defined differently.

- If the NMCom commitments in Stage 2 of the right interaction do not "overlap" with the receiver challenge messages of the $\mathsf{ExtCom}_{\mathrm{SH}}$ commitment in Stage 1 of the left interaction (i.e., $\mathcal{M}$ does not receive the message e of that $\mathsf{ExtCom}_{\mathrm{SH}}$ commitment in the left interaction while giving those NMCom commitments in the right interaction), the committed value of the right interaction is defined as before (i.e., as per Definition 8).
- Otherwise, the committed value of the right interaction is defined as follows. Let $(\rho_1, d_1), \ldots, (\rho_{10n}, d_{10n})$ be the values that are committed in the NMCom commitment in Stage 3 of the right interaction. For each $i \in [10n]$, define $\tilde{\rho}_i$ by $\tilde{\rho}_i := \rho_i$ if (ρ_i, d_i) is a valid decommitment of the i-th NMCom commitment in Stage 2, and by $\tilde{\rho}_i := \bot$ otherwise. Then, the committed value of the right interaction is defined as $\mathsf{Value}(\tilde{\boldsymbol{\rho}}, \mathit{\Gamma})$, where $\tilde{\boldsymbol{\rho}} := (\tilde{\rho}_1, \ldots, \tilde{\rho}_{10n})$.

In what follows, we use *the main* NMCom *row* to denote the NMCom commitments that are used to define the committed values of the right interaction. Hence, in H_1, the main NMCom row is the NMCom commitments in Stage 2 if the committed value of the right interaction is defined as per Definition 8, and it is the NMCom commitments in Stage 3 otherwise.

Lemma 11. *The output of H_1 is statistically indistinguishable from that of H_0.*

Proof. From the definition of H_1, it suffices to show that we have

$$\mathsf{Value}(\tilde{\boldsymbol{\rho}}, \mathit{\Gamma}) = \mathsf{Value}(\boldsymbol{\rho}, \mathit{\Gamma}) \tag{1}$$

whenever the right interaction is accepted. To prove Equality (1), we use Claim 3 in Section 4.3. In fact, Claim 3 implies that to prove Equality (1), it suffice to show that $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ satisfy the following conditions whenever the right interaction is accepted.

1. $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ are 0.99-close, and $\tilde{\rho}_i = \rho_i$ holds for every $i \in \mathit{\Gamma}$.

2. If $\tilde{\rho}_i \neq \bot$, then $\tilde{\rho}_i = \rho_i$.
3. $\tilde{\rho}$ is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i \in [10n]}$ that satisfies $w_i = \tilde{\rho}_i$ for every $i \in \Gamma$ or 0.14-far from any such valid codeword.

Now, we observe that $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ indeed satisfy those conditions whenever the right interaction is accepted.

1. They satisfy the first condition because of the hiding property of $\mathsf{ExtCom}_{\mathrm{SH}}$ in Stage 1 of the right interaction. Indeed, (1) because of the check in Stage 5, $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ satisfy $\tilde{\rho}_i = \rho_i$ for every $i \in \Gamma$ when the right interaction is accepted, and (2) since the subset Γ that is committed in Stage 1 of the right interaction is statistically hidden, the probability that $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ is 0.01-far but satisfy $\tilde{\rho}_i = \rho_i$ for every $i \in \Gamma$ is exponentially small.
2. They satisfy the second condition because of the definition of $\tilde{\boldsymbol{\rho}}$.
3. They satisfy the third condition because of, again, the hiding property of $\mathsf{ExtCom}_{\mathrm{SH}}$ in Stage 1 of the right interaction. Indeed, since Γ is statistically hidden, the probability that $\tilde{\boldsymbol{\rho}}$ is 0.86-close to a valid codeword $\boldsymbol{w} = (w_i)_{i \in [10n]}$ that satisfies $w_i = \tilde{\rho}_i$ for every $i \in \Gamma$ but $\tilde{\boldsymbol{\rho}}$ is also 0.1-far from $\boldsymbol{w}$ is exponentially small.

Hence, we conclude that we have Equation (1) whenever the right interaction is accepted. □

Hybrid H_2. H_2 is the same as H_1 except that in the left interaction, the committed value of the $\mathsf{ExtCom}_{\mathrm{SH}}$ commitment in Stage 1 is extracted by using its extractor (let $\tilde{\Gamma}$ be the extracted value), and the experiment is aborted if the value revealed in Stage 5 is different from the extracted value.

Lemma 12. *The output of H_2 is statistically indistinguishable from that of H_1.*

Proof. From the definition of H_2, it suffices to show that the subset extracted in Stage 1 of the left session is equal to the subset revealed in Stage 5 except with negligible probability. This follows directly from the extractability of $\mathsf{ExtCom}_{\mathrm{SH}}$. □

Remark 7. Since the extraction from a $\mathsf{ExtCom}_{\mathrm{SH}}$ commitment involves rewinding of the committer, in H_2 the man-in-the-middle adversary $\mathcal{M}$ is rewound, and hence the right interaction is also rewound. However, the main NMCom row of the right interaction is either fully rewound (i.e., restarted from its first message) or not rewound at all. (This is because, from its definition, the main NMCom row does not overlap with the receiver challenge message of the $\mathsf{ExtCom}_{\mathrm{SH}}$ commitment in Stage 1 of the left interaction.) We will use this property below.

Hybrid H_3. H_3 is the same as H_2 except that in the left interaction, the value committed in the i-th commitment of the j-th $\mathsf{ExtCom}_{\mathrm{SH}}$ row is switched to $0^{|(\rho_i, d_i)|}$ for every $i \in \tilde{\Gamma}$ and $j \in [k+1]$.

Lemma 13. *The output of H_3 is statistically indistinguishable from that of H_2.*

Proof. The indistinguishability follows directly from the statistical hiding property of $\mathsf{ExtCom}_{\mathrm{SH}}$. (Since $\mathsf{ExtCom}_{\mathrm{SH}}$ is statistically hiding, we can argue that the committed value of the right interaction in H_3 is indistinguishable from that in H_2.) □

Hybrid H_4. H_4 is the same as H_3 except that in Stage 3 of the left interaction, the value committed in the i-th NMCom commitment is switched to $0^{|(\rho_i, d_i)|}$ for every $i \in \tilde{\Gamma}$.

Lemma 14. *The output of* H_4 *is computationally indistinguishable from that of* H_3*.*

Proof. The indistinguishability follows directly from the parallel non-malleability of NMCom. Specifically, by using the parallel non-malleability of NMCom, we can argue that the joint distribution of the view of $\mathcal{M}$ and the committed values of the main NMCom row in H_4 is indistinguishable from that in H_3. (When arguing this indistinguishability, we use the fact that the main NMCom row is either fully rewound or not rewound at all. See Remark 7.) Now, since the committed value of the right interaction can be computed efficiently from the committed values of the main NMCom row, we can conclude that the output of H_4 is also indistinguishable from that of H_3. □

Hybrid H_5. H_5 is the same as H_4 except that in Stage 2 of the left interaction, the value committed in the i-th NMCom commitment is switched to $0^{|\rho_i|}$ for every $i \in \tilde{\Gamma}$.

Lemma 15. *The output of* H_5 *is computationally indistinguishable from that of* H_4*.*

Proof. This lemma can be proven identically with Lemma 14. □

Now, we observe that from the security of Shamir's secret sharing scheme, the left interaction no longer contains any information about the committed value in H_5. Hence, from the above lemmas, we conclude that the output of the non-malleability experiment changes only indistinguishably when the value committed in the left interaction changes. This concludes the proof of non-malleability.

B.3 Proof of Parallel Robust Non-malleability

We next show that the commitment scheme in Appendix B.1 is parallel k-robust. Again, for simplicity we prove only standard k-robustness below; parallel k-robustness can be proven analogously.

For any man-in-the-middle adversary $\mathcal{M}$, we consider the following hybrid experiments.

Hybrid H_0. H_0 is the same as the real robust non-malleability experiment (see Section 3.1). Recall that $\mathcal{M}$ interacts with a machine B in k rounds in the left interaction and with a receiver of the non-malleable commitment in the right interaction, and the output of the experiment is the view of $\mathcal{M}$ and the value that $\mathcal{M}$ committed in the right interaction.

Hybrid H_1. H_1 is the same as H_0 except for the following.

- In the right interaction, we say that a $\mathsf{ExtCom}_{\mathrm{SH}}$ row is *good* if it does not "overlap" with any of the k messages in the left interaction (i.e., if $\mathcal{M}$ does not receive any message in the left interaction while giving that $\mathsf{ExtCom}_{\mathrm{SH}}$ row in the right interaction). Notice that we always have at least one good $\mathsf{ExtCom}_{\mathrm{SH}}$ row since there are $k+1$ $\mathsf{ExtCom}_{\mathrm{SH}}$ rows in total.
  Then, in H_1, the committed values of the first good $\mathsf{ExtCom}_{\mathrm{SH}}$ row are extracted by using its extractor. Let be $(\rho_1, d_1), \ldots, (\rho_{10n}, d_{10n})$ be the extracted values. We remark that, even though the extraction from a $\mathsf{ExtCom}_{\mathrm{SH}}$ row involves rewinding the man-in-the-middle adversary $\mathcal{M}$, in H_1 it is performed in such a way that the left interaction is not rewound. Extracting in this way is possible since a good $\mathsf{ExtCom}_{\mathrm{SH}}$ row does not overlap with any message in the left interaction.[12]
- For each $i \in [10n]$, define $\tilde{\rho}_i$ by $\tilde{\rho}_i := \rho_i$ if (ρ_i, d_i) is a valid decommitment of the i-th NMCom commitment in Stage 2, and by $\tilde{\rho}_i := \bot$ otherwise.
  Then, the committed value of the right interaction is defined as $\mathsf{Value}(\tilde{\boldsymbol{\rho}}, \Gamma)$ rather than as per Definition 8, where $\tilde{\boldsymbol{\rho}} := (\tilde{\rho}_1, \ldots, \tilde{\rho}_{10n})$.

Notice that the output of H_1 is computed in polynomial time.

Lemma 16. *The output of H_0 is statistically indistinguishable from that of H_1.*

This lemma can be proven almost identically with Lemma 11. For completeness, we give a proof below.

Proof. From the definition of H_1, it suffices to show that we have

$$\mathsf{Value}(\tilde{\boldsymbol{\rho}}, \Gamma) = \mathsf{Value}(\boldsymbol{\rho}, \Gamma) \tag{2}$$

whenever the right interaction is accepted. To prove Equality (2), we use Claim 3 in Section 4.3. In fact, Claim 3 implies that to prove Equality (2), it suffice to show that $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ satisfy the following conditions whenever the right interaction is accepted.

1. $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ are 0.99-close, and $\tilde{\rho}_i = \rho_i$ holds for every $i \in \Gamma$.
2. If $\tilde{\rho}_i \neq \bot$, then $\tilde{\rho}_i = \rho_i$.
3. $\tilde{\rho}$ is either 0.9-close to a valid codeword $\boldsymbol{w} = (w_i)_{i \in [10n]}$ that satisfies $w_i = \tilde{\rho}_i$ for every $i \in \Gamma$ or 0.14-far from any such valid codeword.

Now, we observe that $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ indeed satisfy those conditions whenever the right interaction is accepted.

[12] Specifically, if $\mathcal{M}$ requests a message in the left interaction after being rewound, the extractor stops the interaction with $\mathcal{M}$ on the current "thread" and rewinds it again. One can show that the expected number of rewinding is still bounded by a polynomial as long as the extraction is applied to a good $\mathsf{ExtCom}_{\mathrm{SH}}$ row.

1. They satisfy the first condition because of the hiding property of $\mathsf{ExtCom}_{\mathrm{SH}}$ in Stage 1 of the right interaction. Indeed, (1) because of the check in Stage 5 and the extractability of $\mathsf{ExtCom}_{\mathrm{SH}}$, $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ satisfy $\tilde{\rho}_i = \rho_i$ for every $i \in \Gamma$ when the right interaction is accepted, and (2) since the subset Γ that is committed in Stage 1 of the right interaction is statistically hidden, the probability that $\tilde{\boldsymbol{\rho}}$ and $\boldsymbol{\rho}$ is 0.01-far but satisfy $\tilde{\rho}_i = \rho_i$ for every $i \in \Gamma$ is exponentially small.
2. They satisfy the second condition because of the definition of $\tilde{\boldsymbol{\rho}}$.
3. They satisfy the third condition because of, again, the hiding property of $\mathsf{ExtCom}_{\mathrm{SH}}$ in Stage 1 of the right interaction. Indeed, since Γ is statistically hidden, the probability that $\tilde{\boldsymbol{\rho}}$ is 0.86-close to a valid codeword $\boldsymbol{w} = (w_i)_{i\in[10n]}$ that satisfies $w_i = \tilde{\rho}_i$ for every $i \in \Gamma$ but $\tilde{\boldsymbol{\rho}}$ is also 0.1-far from $\boldsymbol{w}$ is exponentially small.

Hence, we conclude that we have Equation (2) whenever the right interaction is accepted. □

Notice that, since the output of H_1 can be computed in polynomial time, the output of H_1 changes only indistinguishably when the left interaction changes indistinguishably. Furthermore, from the above lemma, the same holds w.r.t. the output of H_0. This concludes the proof of k-robustness.

C UC Security and Its SPS Variant

We recall the definition of UC security [4] and its SPS variant [48,2,15]. A part of the text below is taken from [15].

C.1 UC Security

We briefly recall the UC framework. For full details, see [4].

Model for protocol execution. The model for protocol execution consists of an *environment* $\mathcal{Z}$, an *adversary* $\mathcal{A}$, and the parties running protocol π. In the execution of the protocol, the environment $\mathcal{Z}$ is first invoked on external input z. The environment $\mathcal{Z}$ adaptively gives inputs to the parties and receives outputs from them. In addition, $\mathcal{Z}$ communicates freely with $\mathcal{A}$ throughout the execution of the protocol. On inputs from $\mathcal{Z}$, the parties execute π by sending messages to each other. The adversary $\mathcal{A}$ sees all communications between the parties and controls the schedule of the communications. In addition, $\mathcal{A}$ can *corrupt* parties. After corruption, $\mathcal{A}$ receives all internal information of the corrupted parties. Moreover, from now on, $\mathcal{A}$ can fully control the corrupted parties. In this paper, we assume that there exist authenticated communication channels. Thus, the adversary cannot change the contents of messages sent by the honest parties. In addition, in this paper we consider only static adversaries. In other words, we assume that the adversary corrupts parties only at the beginning of the protocol execution. The protocol execution ends when $\mathcal{Z}$ outputs a bit.

Let $\mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}(n,z)$ denote a random variable for the output of $\mathcal{Z}$ on security parameter $n \in \mathbb{N}$ and input $z \in \{0,1\}^*$ with a uniformly-chosen random tape. Let $\mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}$ denote the ensemble $\{\mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}(n,z)\}_{n\in\mathbb{N},z\in\{0,1\}^*}$.

The security of a protocol π is defined using the *ideal protocol*. In the execution of the ideal protocol, all the parties simply hand their inputs to the *ideal functionality* $\mathcal{F}$. The ideal functionality $\mathcal{F}$ carries out the desired task securely and gives outputs to the parties. The parties simply forward these outputs to $\mathcal{Z}$. Let *dummy parties* denote the parties in the ideal protocol. The adversary $\mathcal{S}im$ in the execution of the ideal protocol is often called the *simulator*. Let $\pi(\mathcal{F})$ denote the ideal protocol for functionality $\mathcal{F}$.

Securely realizing an ideal functionality. We say that a protocol π *emulates* protocol ϕ if for any adversary $\mathcal{A}$ there exists an adversary $\mathcal{S}im$ such that no environment $\mathcal{Z}$, on any input, can tell with non-negligible probability whether it is interacting with $\mathcal{A}$ and parties running π or it is interacting with $\mathcal{S}im$ and parties running ϕ. This means that, from the point of view of the environment, running protocol π is just as good as interacting with ϕ. We say that π *securely realizes* an ideal functionality $\mathcal{F}$ if it emulates the ideal protocol $\Pi(\mathcal{F})$. More precise definitions follow. A distribution ensemble is called binary if it consists of distributions over $\{0,1\}$.

Definition 9. *Let π and ϕ be protocols. We say that π* UC-emulates *ϕ if for any adversary $\mathcal{A}$ there exists an adversary $\mathcal{S}im$ such that for any environment $\mathcal{Z}$ that obeys the rules of interaction for UC security, we have* $\mathrm{EXEC}_{\phi,\mathcal{S}im,\mathcal{Z}} \approx \mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}$.

Definition 10. *Let $\mathcal{F}$ be an ideal functionality and let π be a protocol. We say that π* UC-realizes *$\mathcal{F}$ if π UC-emulates the ideal process $\Pi(\mathcal{F})$.*

C.2 UC Security with Super-polynomial Simulation

We next provide a relaxed notion of UC security by giving the simulator access to super-polynomial computational resources.

Definition 11. *Let π and ϕ be protocols. We say that π* UC-SPS-emulates *ϕ if for any adversary $\mathcal{A}$ there exists a super-polynomial-time adversary $\mathcal{S}im$ such that for any environment $\mathcal{Z}$ that obeys the rules of interaction for UC security, we have* $\mathrm{EXEC}_{\phi,\mathcal{S}im,\mathcal{Z}} \approx \mathrm{EXEC}_{\pi,\mathcal{A},\mathcal{Z}}$.

Definition 12. *Let $\mathcal{F}$ be an ideal functionality and let π be a protocol. We say that π* UC-SPS-realizes *$\mathcal{F}$ if π UC-SPS-emulates the ideal process $\Pi(\mathcal{F})$.*

The multi-session extension of an ideal functionality. When showing concurrent security of a protocol π under SPS security, we need to construct a simulator in a setting where parties execute π concurrently. (In other words, unlike in UC security, we cannot rely on the *composition theorem* in SPS security.)

To consider the simulator in such a setting, we use a *multi-session extension* of an ideal functionality. Let $\mathcal{F}$ be an ideal functionality. Recall that $\mathcal{F}$ expects each incoming message to contain a special field consisting of its *session ID (SID)*. All messages received by $\mathcal{F}$ are expected to have the same SID. (Messages that have different SIDs than that of the first message are ignored.) Similarly, all outgoing messages generated by $\mathcal{F}$ carry the same SID.

Below, we recall multi-session extension of an ideal functionality from [8]. The multi-session extension of $\mathcal{F}$, denoted by $\hat{\mathcal{F}}$, is defined as follows. $\hat{\mathcal{F}}$ expects each incoming message to contain two special fields. The first is the usual SID field as in any ideal functionality. The second field is called the *sub-session ID* (SSID) field. Upon receiving a message $(sid, ssid, v)$ (where sid is the SID, $ssid$ is the SSID, and v is an arbitrary value or list of values), $\hat{\mathcal{F}}$ first verifies that sid is the same as that of the first message, otherwise the message is ignored. Next, $\hat{\mathcal{F}}$ checks if there is a running copy of $\mathcal{F}$ whose session ID is $ssid$. If so, then $\hat{\mathcal{F}}$ activates that copy of $\mathcal{F}$ with incoming message $(ssid, v)$, and follows the instructions of this copy. Otherwise, a new copy of $\mathcal{F}$ is invoked (within $\hat{\mathcal{F}}$) and immediately activated with input $(ssid, v)$. From now on, this copy is associated with sub-session ID $ssid$. Whenever a copy of $\mathcal{F}$ sends a message $(ssid, v')$ to some party P_i, $\hat{\mathcal{F}}$ sends $(sid, ssid, v')$ to P_i, and sends $ssid$ to the adversary. (Sending $ssid$ to the adversary implies that $\hat{\mathcal{F}}$ does not hide which copy of $\mathcal{F}$ is being activated within $\hat{\mathcal{F}}$.)

D Omitted Proofs

D.1 The Second Half of Proof of Lemma 1

Case 2. S is corrupted in the $i^*(n)$-th session. We show that when $\mathcal{A}$ cheats, we can break the hiding property of the Com commitment in Stage 1-2 (i.e., the commitment by which Γ_R is committed to). From the definition of the invariant condition (Definition 6), when $\mathcal{A}$ cheats, we have $I_{\rm bad} \cap \Gamma_R = \emptyset$ and either $|I_{\rm bad}| \geq 0.1n$ or $\exists b \in \{0,1\}$ s.t. $\boldsymbol{\rho}_b^{\mathsf{nm}}$ is 0.85-close to but 0.1-far from a valid codeword $\boldsymbol{w} = (w_i)_{i\in\Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{nm}}$ for every $i \in \Gamma_R$, where $I_{\rm bad}$ and $\boldsymbol{\rho}_b^{\mathsf{nm}}$ are defined from the committed values of the NMCom commitments in Stage 7. Similar to Case 1, we first show that we can "approximate" $I_{\rm bad}$ and $\boldsymbol{\rho}_b^{\mathsf{nm}}$ by extracting the committed values of the ExtCom commitments in Stage 7 using its extractability.

First, we observe that if we extract the committed values of the ExtCom commitments in Stage 7 of the $i^*(n)$-th session, the extracted values, $(\hat{a}_1^S, \hat{d}_1^S, \hat{e}_1^S), \ldots, (\hat{a}_{11n}^S, \hat{d}_{11n}^S, \hat{e}_{11n}^S)$, satisfy the following.

- Let $\hat{I}_{\rm bad} \subset [11n]$ be a set such that $i \in \hat{I}_{\rm bad}$ if and only if
  1. $((\hat{a}_1^S, \hat{d}_1^S), \hat{e}_1^S)$ is not a valid decommitment of the i-th NMCom commitment in Stage 7, or
  2. $(\hat{a}_i^S, \hat{d}_i^S)$ is not a valid decommitment of the i-th Com commitment in Stage 2-1, or

3. S does not execute the i-th mS-OT in Stage 3 honestly using $\hat{s}_{i,0} \,\|\, \hat{s}_{i,1} \,\|\, \hat{\tau}_i^S$ as the input and randomness, where $\hat{s}_{i,0} \,\|\, \hat{s}_{i,1} \,\|\, \hat{\tau}_i^S$ is obtained from $\hat{r}_i^S = \hat{a}_i^S \oplus b_i^S$.

Also, for each $b \in \{0,1\}$, let $\hat{\boldsymbol{\rho}}_b = (\hat{\rho}_{b,i})_{i \in \Delta}$ be defined as follows: $\hat{\rho}_{b,i} \stackrel{\text{def}}{=} \beta_{b,i} \oplus \hat{s}_{i,b\oplus\alpha_i}$ if $i \notin \hat{I}_{\text{bad}}$ and $\hat{\rho}_{b,i} \stackrel{\text{def}}{=} \perp$ otherwise. Then, we have

- $\hat{I}_{\text{bad}} \cap \varGamma_R = \emptyset$, and
- either $|\hat{I}_{\text{bad}}| \geq 0.1n$ or there exists $b \in \{0,1\}$ such that $\hat{\rho}_b$ is 0.8-close to but 0.1-far from a valid codeword $\boldsymbol{w} = (w_i)_{i \in \Delta}$ that satisfies $w_i = \hat{\rho}_{b,i}$ for every $i \in \varGamma_R$

with probability at least $1/2p(n)$.

More precisely, we observe that when $\mathcal{A}$ cheats in the $i^*(n)$-th session, the extracted values satisfied the above condition except with negligible probability. Recall that when $\mathcal{A}$ cheats, the cut-and-choose in Stage 8 is accepting but we have

- $|I_{\text{bad}}| \geq 0.1n$, or
- $\exists b \in \{0,1\}$ s.t. ρ_b^{nm} is 0.85-close to but 0.1-far from a valid codeword $\boldsymbol{w} = (w_i)_{i \in \Delta}$ that satisfies $w_i = \rho_{b,i}^{\mathsf{nm}}$ for every $i \in \varGamma_R$.

Also, notice that we have $\hat{I}_{\text{bad}} \cap \varGamma_R = \emptyset$ when the cut-and-choose in Stage 8 is accepting, and have $|\hat{I}_{\text{bad}}| \geq 0.1n$ when $|I_{\text{bad}}| \geq 0.1n$ (this is because we have $I_{\text{bad}} \subseteq \hat{I}_{\text{bad}}$ from the definitions of $I_{\text{bad}}, \hat{I}_{\text{bad}}$). Hence, to show that the extracted values satisfy the above condition when $\mathcal{A}$ cheats, it suffices to show that when $\exists b^* \in \{0,1\}$ s.t. $\boldsymbol{\rho}_{b^*}^{\mathsf{nm}}$ is 0.85-close to but 0.1-far from a valid codeword $\boldsymbol{w} = (w_i)_{i \in \Delta}$ that satisfies $w_i = \rho_{b^*,i}^{\mathsf{nm}}$ for every $i \in \varGamma_R$, we have either $|\hat{I}_{\text{bad}}| \geq 0.1n$ or $\hat{\boldsymbol{\rho}}_{b^*}$ is 0.8-close to but 0.1-far from $\boldsymbol{w}$ and satisfies $w_i = \hat{\rho}_{b^*,i}$ for every $i \in \varGamma_R$. This can be shown as follows.

- If $|\hat{I}_{\text{bad}}| \geq 0.1n$, we are done.
- If $|\hat{I}_{\text{bad}}| < 0.1n$, we have that $\hat{\rho}_{b^*}$ is 0.8-close to but 0.1-far from $\boldsymbol{w}$ and satisfies $w_i = \hat{\rho}_{b^*,i}$ for every $i \in \varGamma_R$. This is because if $|\hat{I}_{\text{bad}}| < 0.1n$,
  1. $\hat{\boldsymbol{\rho}}_{b^*}$ is 0.8-close to $\boldsymbol{w}$ since it is 0.99-close to $\boldsymbol{\rho}_{b^*}^{\mathsf{nm}}$ when $|\hat{I}_{\text{bad}}| < 0.1n$, and $\boldsymbol{\rho}_{b^*}^{\mathsf{nm}}$ is 0.85-close to $\boldsymbol{w}$,
  2. $\hat{\boldsymbol{\rho}}_{b^*}$ is 0.1-far from $\boldsymbol{w}$ since for every i such that $\rho_{b^*,i}^{\mathsf{nm}} \neq w_i$, we have $\hat{\rho}_{b^*,i} \neq w_i$ from the definition of $\hat{\boldsymbol{\rho}}$, and
  3. $\hat{\boldsymbol{\rho}}_{b^*}$ satisfies $w_i = \hat{\rho}_{b^*,i}$ for every $i \in \varGamma_R$ since we have $\hat{\rho}_{b^*,i} = \rho_{b^*,i}^{\mathsf{nm}}$ for every $i \in \varGamma_R$ when the cut-and-choose in Stage 8 is accepting, and $\boldsymbol{\rho}_{b^*}^{\mathsf{nm}}$ satisfies $w_i = \rho_{b^*,i}^{\mathsf{nm}}$ for every $i \in \varGamma_R$.

Based on this observation, we derive contradiction by considering the following adversary $\mathcal{A}_{\mathsf{Com}}$ against the hiding property of Com.

> $\mathcal{A}_{\mathsf{Com}}$ receives a Com commitment c^* in which either $\varGamma_R^0$ or $\varGamma_R^1$ is committed, where $\varGamma_R^0, \varGamma_R^1 \subset [11n]$ are random subsets of size n.
> Then, $\mathcal{A}_{\mathsf{Com}}$ internally executes the experiment H_0 honestly except that in the $i^*(n)$-th session, $\mathcal{A}_{\mathsf{Com}}$ uses c^* as the commitment in Stage 1-2 (i.e.,

> as the Com commitment in which R commits to a subset $\mathit{\Gamma}_R$). When the experiment H_0 reaches Stage 7 of the $i^*(n)$-th session, $\mathcal{A}_{\mathsf{Com}}$ extracts the committed values of the ExtCom commitments in this stage by using its extractability. Let $\hat{I}_{\mathrm{bad}}$ and $\hat{\boldsymbol{\rho}}_b$ ($b \in \{0,1\}$) be defined as above from the extracted values. Then, $\mathcal{A}_{\mathsf{Com}}$ outputs 1 if and only if
> - $\hat{I}_{\mathrm{bad}} \cap \mathit{\Gamma}_R^1 = \emptyset$, and
> - either $|\hat{I}_{\mathrm{bad}}| \geq 0.1n$ or there exists $b \in \{0,1\}$ such that $\hat{\boldsymbol{\rho}}_b$ is 0.8-close to but 0.1-far from a valid codeword $\boldsymbol{w} = (w_i)_{i \in \mathit{\Delta}}$ that satisfies $w_i = \hat{\rho}_{b,i}$ for every $i \in \mathit{\Gamma}_R^1$.

When $\mathcal{A}_{\mathsf{Com}}$ receives a commitment to $\mathit{\Gamma}_R^1$, $\mathcal{A}_{\mathsf{Com}}$ outputs 1 with probability $1/2p(n)$ (this follows from the above observation). It thus suffices to see that when $\mathcal{A}_{\mathsf{Com}}$ receives a commitment to $\mathit{\Gamma}_R^0$, $\mathcal{A}_{\mathsf{Com}}$ outputs 1 with exponentially small probability. This can be seen by observing that when no information about $\mathit{\Gamma}_S^1$ is fed into H_0, the following probabilities are exponentially small.

1. the probability that $|\hat{I}_{\mathrm{bad}}| \geq 0.1n$ but $\hat{I}_{\mathrm{bad}} \cap \mathit{\Gamma}_R^1 = \emptyset$
2. the probability that there exists $b \in \{0,1\}$ such that $\hat{\boldsymbol{\rho}}_b$ is 0.8-close to but 0.1-far from a valid codeword $\boldsymbol{w} = (w_i)_{i \in \mathit{\Delta}}$ that satisfies $w_i = \hat{\rho}_{b,i}$ for every $i \in \mathit{\Gamma}_R^1$

Hence, $\mathcal{A}_{\mathsf{Com}}$ breaks the hiding property of Com.

D.2 Proof of Claim 2

Proof. We first notice that the indistinguishability between $H'_{k-1:7}$ and $H_{k:1}$ can be shown as in the proof of Claim 1. (The only difference is that we use the hiding property of NMCom rather than that of ExtCom.)

We next show that in $H_{k:1}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. Assume for contradiction that in $H_{k:1}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H'_{k-1:7}$ but with non-negligible probability in $H_{k:1}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the non-malleability of NMCom as follows.

> The man-in-the-middle adversary $\mathcal{A}_{\mathsf{NMCom}}$ internally executes $H'_{k-1:7}$ from SM_k using the non-uniform advice. In Stage 4 of session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ sends $(a_i^R, d_i^R)_{i \notin \mathit{\Gamma}_S}$ and $(0,0)_{i \notin \mathit{\Gamma}_S}$ to the external committer, receives back NMCom commitments (in which either $(a_i^R, d_i^R)_{i \notin \mathit{\Gamma}_S}$ or $(0,0)_{i \notin \mathit{\Gamma}_S}$ are committed to), and feeds them into $H'_{k-1:7}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver (specifically, the NMCom commitments in Stage 4 if R is corrupted and in Stage 7 if S is corrupted). After the execution of $H'_{k-1:7}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.

> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
>
> When $\mathcal{A}_{\mathsf{NMCom}}$ receives commitments to $(a_i^R, d_i^R)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H'_{k-1:7}$, whereas when $\mathcal{A}_{\mathsf{NMCom}}$ receives a commitments to $(0,0)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H_{k:1}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H'_{k-1:7}$ but with non-negligible probability in $H_{k:1}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the non-malleability of NMCom.

□

D.3 Proof of Lemma 3

Proof. We first show the indistinguishability between $H_{k:1}$ and $H_{k:2}$. Assume for contradiction that $H_{k:1}$ and $H_{k:2}$ are distinguishable. From an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $H_{k:1}$ and $H_{k:2}$ are still distinguishable. Then, by considering the transcript (including the inputs and randomness of all the parties) and the extracted values up until SM_k as non-uniform advice, we can break the hiding property of Com as follows.

> The adversary $\mathcal{A}_{\mathsf{Com}}$ internally executes $H_{k:1}$ from SM_k using the non-uniform advice. In Stage 2-2 of session $s(k)$, $\mathcal{A}_{\mathsf{Com}}$ chooses random strings $\tilde{\boldsymbol{a}}^R = (\tilde{a}_1^R, \ldots, \tilde{a}_{11n}^R)$ in addition to $\boldsymbol{a}^R = (a_1^R, \ldots, a_{11n}^R)$, sends $(a_i^R)_{i \notin \Gamma_S}$ and $(\tilde{a}_i^R)_{i \notin \Gamma_S}$ to the external committer and receives back Com commitments (in which either $(a_i^R)_{i \notin \Gamma_S}$ or $(\tilde{a}_i^R)_{i \notin \Gamma_S}$ are committed to), and feeds them into $H_{k:1}$; in the subsequent stages, $\mathcal{A}$ proceeds the experiment by computing the outcome of the coin tossing assuming that the committed values of the commitments are $(a_i^R)_{i \notin \Gamma_S}$. After the execution of $H_{k:1}$ finishes, $\mathcal{A}_{\mathsf{Com}}$ outputs whatever $\mathcal{Z}$ outputs in the experiment.
>
> When $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(a_i^R)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H_{k:1}$, whereas when $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(\tilde{a}_i^R)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H_{k:2}$ (this is because when $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(\tilde{a}_i^R)_{i \notin \Gamma_S}$, the value $r_i^R = a_i^R \oplus b_i^R$ for each $i \notin \Gamma_S$ is uniformly random for $\mathcal{A}$ in the experiment and hence the mS-OT for each $i \notin \Gamma_S$ is executed with a random input and true randomness). Hence, from the assumption that $H_{k:1}$ and $H_{k:2}$ are distinguishable, $\mathcal{A}_{\mathsf{Com}}$ distinguishes Com commitments.

We next show that in $H_{k:2}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. Assume for contradiction that in $H_{k:2}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H_{k:1}$ but with non-negligible probability in $H_{k:2}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the robust non-malleability of NMCom as follows.

> The adversary $\mathcal{A}_{\mathsf{NMCom}}$, who interacts with a committer of Com and a receiver of NMCom, internally executes $H_{k:1}$ from SM_k using the non-uniform advice. In Stage 2-2 of session $s(k)$, $\mathcal{A}_{\mathsf{Com}}$ chooses random strings $\tilde{\boldsymbol{a}}^R = (\tilde{a}_1^R, \ldots, \tilde{a}_{11n}^R)$ in addition to $\boldsymbol{a}^R = (a_1^R, \ldots, a_{11n}^R)$, sends $(a_i^R)_{i \notin \Gamma_S}$ and $(\tilde{a}_i^R)_{i \notin \Gamma_S}$ to the external committer and receives back Com commitments (in which either $(a_i^R)_{i \notin \Gamma_S}$ or $(\tilde{a}_i^R)_{i \notin \Gamma_S}$ are committed to), and feeds them into $H_{k:1}$; in the subsequent stages, $\mathcal{A}$ proceeds the experiment by computing the outcome of the coin tossing assuming that the committed values of the commitments are $(a_i^R)_{i \notin \Gamma_S}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver. After the execution of $H_{k:1}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.
>
> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
>
> When $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(a_i^R)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H_{k:1}$, whereas when $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(\tilde{a}_i^R)_{i \notin \Gamma_S}$, the internally executed experiment is identical with $H_{k:2}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H_{k:1}$ but with non-negligible probability in $H_{k:2}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the robust non-malleability of NMCom.

This completes the proof of Lemma 3. □

D.4 Proof of Lemma 5

Proof. Recall that $H_{k:3}$ and $H_{k:4}$ differ only in that in session $s(k)$ of $H_{k:4}$, if S is corrupted and SM_k is $\mathsf{third\ special\ message}$, α_i is a random bit rather than $\alpha_i = u \oplus c_i$ for every $i \in \Delta$ in Stage 6-1.

We first show the indistinguishability between $H_{k:3}$ and $H_{k:4}$. Intuitively, the indistinguishability follows from the security of mS-OT: For every $i \notin \Gamma_S$, the choice bit c_i of the i-th mS-OT in Stage 3 is hidden from $\mathcal{A}$ and hence $\alpha_i = u \oplus c_i$ in $H_{k:3}$ is indistinguishable from a random bit. Formally, we consider the following security game against cheating sender S^* of mS-OT.

> The cheating sender S^* first participates in $10n$ instances of mS-OTs in parallel with an honest receiver R, who uses a random input $c_i \in \{0, 1\}$ in the i-th instance. After the execution with R, S^* receives either the choice bits $\{c_i\}$ or random bits and then guesses which is the case. If S^* guesses correctly, we say that S^* wins the game.

From the security of mS-OT against malicious senders, any cheating S^* wins the game with probability at most $1/2 + \mathsf{negl}(n)$. Now, we assume for contradiction that $H_{k:3}$ and $H_{k:4}$ are distinguishable, and we derive a contradiction by constructing an adversary who wins the above game with probability non-negligibly higher than 1/2. From an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $H_{k:3}$ and $H_{k:4}$ are still distinguishable. Then, by considering the transcript (including the inputs and randomness of all the parties) and the extracted values up until SM_k as non-uniform advice, we can obtain an adversary who wins the above game with probability non-negligibly higher than 1/2 as follows.

> The adversary $\mathcal{A}_{\mathsf{OT}}$ internally executes $H_{k:3}$ from SM_k using the non-uniform advice. In Stage 3 of session $s(k)$, $\mathcal{A}_{\mathsf{OT}}$ executes the i-th mS-OT by itself for every $i \in \Gamma_S$ but obtains the other $10n$ instances of mS-OT from the external receiver. (Recall that in $H_{k:3}$, the subset Γ_S is extracted in Stage 1-1.) Then, in Stage 6 of session $s(k)$, $\mathcal{A}_{\mathsf{OT}}$ receives bits $\{c_i^*\}_{i\in\Delta}$ from the external receiver and uses them to compute $\{\alpha\}_{i\in\Delta}$, i.e., $\alpha_i \overset{\text{def}}{=} u \oplus c_i^*$. After the execution of $H_{k:3}$ finishes, $\mathcal{A}_{\mathsf{OT}}$ outputs whatever $\mathcal{Z}$ outputs in the experiment.
>
> When $\mathcal{A}_{\mathsf{OT}}$ receives the choice bits of the mS-OTs as $\{c_i^*\}_{i\in\Delta}$, the internally executed experiment is identical with $H_{k:3}$, whereas when $\mathcal{A}_{\mathsf{OT}}$ receives random bits as $\{c_i^*\}_{i\in\Delta}$, the internally executed experiment is identical with $H_{k:4}$. Hence, from the assumption that $H_{k:3}$ and $H_{k:4}$ are distinguishable, $\mathcal{A}_{\mathsf{OT}}$ wins the game with probability non-negligibly higher than 1/2.

We next show that in $H_{k:4}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. (The argument below is similar to the one in the proof of Lemma 2.) Assume for contradiction that in $H_{k:4}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H_{k:3}$ but with non-negligible probability in $H_{k:4}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the robust non-malleability of NMCom as follows.

> The adversary $\mathcal{A}_{\mathsf{NMCom}}$, who participates in the above game of mS-OT while interacting with a receiver of NMCom, internally executes $H_{k:3}$ from SM_k using the non-uniform advice. In Stage 3 of session $s(k)$, $\mathcal{A}_{\mathsf{OT}}$ executes the i-th mS-OT by itself for every $i \in \Gamma_S$ but obtains the other $10n$ instances of mS-OT from the external receiver. Then, in Stage 6 of session $s(k)$, $\mathcal{A}_{\mathsf{OT}}$ receives bits $\{c_i^*\}_{i\in\Delta}$ from the external receiver and uses them to compute $\{\alpha\}_{i\in\Delta}$, i.e., $\alpha_i \overset{\text{def}}{=} u \oplus c_i^*$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver. After the execution of $H_{k:3}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.
>
> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
>
> When $\mathcal{A}_{\mathsf{OT}}$ receives the choice bits of the mS-OTs as $\{c_i^*\}_{i\in\Delta}$, the internally executed experiment is identical with $H_{k:3}$, whereas when $\mathcal{A}_{\mathsf{OT}}$

> receives random bits as $\{c^*_i\}_{i\in\varDelta}$, the internally executed experiment is identical with $H_{k:4}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H_{k:3}$ but with non-negligible probability in $H_{k:4}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the robust non-malleability of NMCom.

This completes the proof of Lemma 5. □

D.5 Proof of Lemma 6

Proof. Recall that hybrids $H_{k:4}, H_{k:5}$ differ only in the values committed to in NMCom and ExtCom for the indices outside of $\varGamma_R$. Since the binding property of Com guarantees that the subset opened in Stage 7 is equal to $\varGamma_R$, those commitments are never opened, and the check in Stage 8 does not fail in both hybrids.

We prove the lemma by using a hybrid argument. Specifically, we consider the following intermediate hybrid $H'_{k:4}$.

- $H'_{k:4}$ is the same as $H_{k:4}$ except that in session $s(k)$, if R is corrupted and SM_k is second special message,
  - the committed subset $\varGamma_R$ is extracted by brute force in Stage 1-2, and
  - the value committed in the i-th ExtCom commitment in Stage 7 is switched to an all-zero string for every $i \notin \varGamma_R$.

Claim 8. *Assume that in $H_{k:4}$, $\mathcal{A}$ does not cheat in sessions $s(k),\ldots,s(4m)$ except with negligible probability. Then,*

- *$H_{k:4}$ and $H'_{k:4}$ are indistinguishable, and*
- *in $H'_{k:4}$, $\mathcal{A}$ does not cheat in sessions $s(k),\ldots,s(4m)$ except with negligible probability.*

Proof. We first show the indistinguishability between $H_{k:4}$ and $H'_{k:4}$. Assume for contradiction that $H_{k:4}$ and $H'_{k:4}$ are distinguishable. From an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $H_{k:4}$ and $H'_{k:4}$ are still distinguishable. Then, by considering the transcript (including the inputs and randomness of all the parties) and the extracted values up until SM_k as non-uniform advice, we can break the hiding property of ExtCom as follows.

> The adversary $\mathcal{A}_{\mathsf{ExtCom}}$ internally executes $H_{k:4}$ from SM_k using the non-uniform advice. In Stage 7 of session $s(k)$, $\mathcal{A}_{\mathsf{ExtCom}}$ sends $(a^S_i, d^S_i, e^S_i)_{i\notin\varGamma_R}$ and $(0,0,0)_{i\notin\varGamma_R}$ to the external committer, receives back ExtCom commitments (in which either $(a^S_i, d^S_i, e^S_i)_{i\notin\varGamma_R}$ or $(0,0,0)_{i\notin\varGamma_R}$ are committed to), and feeds them into $H_{k:4}$. After the execution of $H_{k:4}$ finishes, $\mathcal{A}_{\mathsf{ExtCom}}$ outputs whatever $\mathcal{Z}$ outputs in the experiment.
>
> When $\mathcal{A}_{\mathsf{ExtCom}}$ receives commitments to $(a^S_i, d^S_i, e^S_i)_{i\notin\varGamma_R}$, the internally executed experiment is identical with $H_{k:4}$, whereas when $\mathcal{A}_{\mathsf{ExtCom}}$ receives a commitments to $(0,0,0)_{i\notin\varGamma_R}$, the internally executed experiment is identical with $H'_{k:4}$. Hence, from the assumption that $H_{k:4}$ and $H'_{k:4}$ are distinguishable (even after being fixed up until SM_k), $\mathcal{A}_{\mathsf{ExtCom}}$ distinguishes ExtCom commitments.

We next show that in $H'_{k:4}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. Assume for contradiction that in $H'_{k:4}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H_{k:4}$ but with non-negligible probability in $H'_{k:4}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the robust non-malleability of NMCom as follows.

> The man-in-the-meddle adversary $\mathcal{A}_{\mathsf{NMCom}}$ internally executes $H_{k:4}$ from SM_k using the non-uniform advice. In Stage 7 of session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ sends $(a_i^S, d_i^S, e_i^S)_{i \notin \Gamma_R}$ and $(0,0,0)_{i \notin \Gamma_R}$ to the external committer, receives back ExtCom commitments (in which either $(a_i^S, d_i^S, e_i^S)_{i \notin \Gamma_R}$ or $(0,0,0)_{i \notin \Gamma_R}$ are committed to), and feeds them into $H_{k:4}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver. After the execution of $H_{k:4}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.
> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
> When $\mathcal{A}_{\mathsf{NMCom}}$ receives commitments to $(a_i^S, d_i^S, e_i^S)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H_{k:4}$, whereas when $\mathcal{A}_{\mathsf{NMCom}}$ receives a commitments to $(0,0,0)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H'_{k:4}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H_{k:4}$ but with non-negligible probability in $H'_{k:4}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the non-malleability of NMCom.

□

Claim 9. *Assume that in $H'_{k:4}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability. Then,*

- *$H'_{k:4}$ and $H_{k:5}$ are indistinguishable, and*
- *in $H_{k:5}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$ except with negligible probability.*

Proof. We first notice that the indistinguishability between $H'_{k-1:7}$ and $H_{k:1}$ can be shown as in the proof of Claim 8. (The only difference is that we use the hiding property of NMCom rather than that of ExtCom.)

We next show that in $H_{k:5}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. Assume for contradiction that in $H_{k:5}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H'_{k:4}$ but with non-negligible probability in $H_{k:5}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the non-malleability of NMCom as follows.

> The man-in-the-meddle adversary $\mathcal{A}_{\mathsf{NMCom}}$ internally executes $H'_{k:4}$ from SM_k using the non-uniform advice. In Stage 7 of session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ sends $(a_i^S, d_i^S)_{i \notin \Gamma_R}$ and $(0,0)_{i \notin \Gamma_R}$ to the external committer, receives back NMCom commitments (in which either $(a_i^S, d_i^S)_{i \notin \Gamma_R}$ or $(0,0)_{i \notin \Gamma_R}$ are committed to), and feeds them into $H'_{k:4}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver. After the execution of $H'_{k:4}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.
>
> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
>
> When $\mathcal{A}_{\mathsf{NMCom}}$ receives commitments to $(a_i^S, d_i^S)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H'_{k:4}$, whereas when $\mathcal{A}_{\mathsf{NMCom}}$ receives a commitments to $(0,0)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H_{k:5}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H'_{k:4}$ but with non-negligible probability in $H_{k:5}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the non-malleability of NMCom.

□

This completes the proof of Lemma 6. □

D.6 Proof of Lemma 7

Proof. Recall that hybrids $H_{k:5}, H_{k:6}$ differ only in the inputs and the randomness that are used in some of the mS-OTs in Stage 3, where those that are derived from the outcomes of the coin tossing is used in $H_{k:5}$ and random inputs and true randomness are used in $H_{k:6}$. We prove the lemma by relying on the hiding property of Com in the coin tossing.

We first show the indistinguishability between $H_{k:5}$ and $H_{k:6}$. Assume for contradiction that $H_{k:5}$ and $H_{k:6}$ are distinguishable. From an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $H_{k:5}$ and $H_{k:6}$ are still distinguishable. Then, by considering the transcript (including the inputs and randomness of all the parties) and the extracted values up until SM_k as non-uniform advice, we can break the hiding property of Com as follows.

> The adversary $\mathcal{A}_{\mathsf{Com}}$ internally executes $H_{k:5}$ from SM_k using the non-uniform advice. In Stage 2-1 of session $s(k)$, $\mathcal{A}_{\mathsf{Com}}$ chooses random strings $\tilde{\boldsymbol{a}}^S = (\tilde{a}_1^S, \ldots, \tilde{a}_{11n}^S)$ in addition to $\boldsymbol{a}^S = (a_1^S, \ldots, a_{11n}^S)$, sends $(a_i^S)_{i \notin \Gamma_R}$ and $(\tilde{a}_i^S)_{i \notin \Gamma_R}$ to the external committer and receives back Com commitments (in which either $(a_i^S)_{i \notin \Gamma_R}$ or $(\tilde{a}_i^S)_{i \notin \Gamma_R}$ are committed to), and feeds them into $H_{k:5}$; in the subsequent stages, $\mathcal{A}$ proceeds the experiment by computing the outcome of the coin tossing assuming that the committed values of the commitments are $(a_i^S)_{i \notin \Gamma_R}$. After the execution of $H_{k:5}$ finishes, $\mathcal{A}_{\mathsf{Com}}$ outputs whatever $\mathcal{Z}$ outputs in the experiment.

> When $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(a_i^S)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H_{k:5}$, whereas when $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(\tilde{a}_i^S)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H_{k:6}$ (this is because when $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(\tilde{a}_i^S)_{i \notin \Gamma_R}$, the value $r_i^S \stackrel{\text{def}}{=} a_i^S \oplus b_i^S$ for each $i \notin \Gamma_R$ is uniformly random for $\mathcal{A}$ in the experiment and hence the mS-OTs for each $i \notin \Gamma_R$ is executed with random inputs and true randomness). Hence, from the assumption that $H_{k:5}$ and $H_{k:6}$ are distinguishable, $\mathcal{A}_{\mathsf{Com}}$ distinguishes Com commitments.

We next show that in $H_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. Assume for contradiction that in $H_{k:6}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H_{k:5}$ but with non-negligible probability in $H_{k:6}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the robust non-malleability of NMCom as follows.

> The adversary $\mathcal{A}_{\mathsf{NMCom}}$, who interacts with a committer of Com and a receiver of NMCom, internally executes $H_{k:5}$ from SM_k using the non-uniform advice. In Stage 2-1 of session $s(k)$, $\mathcal{A}_{\mathsf{Com}}$ chooses random strings $\tilde{\boldsymbol{a}}^S = (\tilde{a}_1^S, \ldots, \tilde{a}_{11n}^S)$ in addition to $\boldsymbol{a}^S = (a_1^S, \ldots, a_{11n}^S)$, sends $(a_i^S)_{i \notin \Gamma_R}$ and $(\tilde{a}_i^S)_{i \notin \Gamma_R}$ to the external committer and receives back Com commitments (in which either $(a_i^S)_{i \notin \Gamma_R}$ or $(\tilde{a}_i^S)_{i \notin \Gamma_R}$ are committed to), and feeds them into $H_{k:5}$; in the subsequent stages, $\mathcal{A}$ proceeds the experiment by computing the outcome of the coin tossing assuming that the committed values of the commitments are $(a_i^S)_{i \notin \Gamma_R}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver. After the execution of $H_{k:5}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.
> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
> When $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(a_i^S)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H_{k:5}$, whereas when $\mathcal{A}_{\mathsf{Com}}$ receives commitments to $(\tilde{a}_i^S)_{i \notin \Gamma_R}$, the internally executed experiment is identical with $H_{k:6}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H_{k:5}$ but with non-negligible probability in $H_{k:6}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the robust non-malleability of NMCom.

This completes the proof of Lemma 7. □

D.7 Proof of Claim 5

Proof. Recall that $H_{k:6}$ and $H'_{k:6}$ differ only in that in session $s(k)$ of $H'_{k:6}$, if R is corrupted and SM_k is $\mathsf{fourth\ special\ message}$, $\beta_{b,i}$ is a random bit rather than $\beta_{b,i} = \rho_{b,i} \oplus s_{i,b \oplus \alpha_i}$ for every $b \in \{0,1\}$ and $i \in \Delta \setminus I_b$.

First, we show the indistinguishability between $H_{k:6}$ and $H'_{k:6}$. Roughly, we prove the indistinguishability using the security of mS-OT: For every $i \in \Delta \setminus I_b$, $\mathcal{A}$ executed the i-th mS-OT honestly with choice bit $(1-b) \oplus \alpha_i$, and the sender's input and randomness of this mS-OT are not revealed in Stage 8; therefore, the value of $s_{i,b\oplus\alpha_i}$ is hidden from $\mathcal{A}$ and thus $\beta_{b,i} = \rho_{b,i} \oplus s_{i,b\oplus\alpha_i}$ is indistinguishable from a random bit. Formally, we consider the following security game against cheating receiver R^* of mS-OT.

> The cheating receiver R^* gets random input-randomness pairs $(c_i, \tau_i^R)_i$ of mS-OT instances as input. R^* then participates in $9n$ instances of mS-OTs in parallel with an honest sender S, who uses a random input $(s_{i,0}, s_{i,1})$ in the i-th instance. After the execution with S, R^* receives bits $(s^*_{i,0}, s^*_{i,1})_i$ that are defined as follows: Let $b^* \in \{0,1\}$ be a randomly chosen bit; if $b^* = 0$, then for every i, $s^*_{i,0} \overset{\text{def}}{=} s_{i,0}$ and $s^*_{i,1} \overset{\text{def}}{=} s_{i,1}$; if $b^* = 1$, then for every i such that R^* behaved honestly in the i-th mS-OT using (c_i, τ_i^R) as input and randomness, $s^*_{i,c_i} \overset{\text{def}}{=} s_{i,c_i}$ but $s^*_{i,1-c_i}$ is a random bit, and for every other i, $s^*_{i,0} \overset{\text{def}}{=} s_{i,0}$ and $s^*_{i,1} \overset{\text{def}}{=} s_{i,1}$. Then, R^* guesses the value of b^*, and if the guess is correct, we say that R^* wins the game.

From the security of mS-OT against semi-honest receivers, any cheating R^* wins the game with probability at most $1/2 + \mathsf{negl}(n)$. Now, we assume for contradiction that $H_{k:6}$ and $H'_{k:6}$ are distinguishable, and we derive a contradiction by constructing an adversary who wins the above game with probability non-negligibly higher than $1/2$. From an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $H_{k:6}$ and $H'_{k:6}$ are still distinguishable. Then, by considering the transcript (including the inputs and randomness of all the parties) and the extracted values up until SM_k as non-uniform advice, we can obtain an adversary who wins the above game with probability non-negligibly higher than $1/2$ as follows.

> The adversary R^* gets random input-randomness pairs $(c_i, \tau_i^R)_{i\in\Delta\setminus\Gamma_R}$ of mS-OT instances as its input, and internally executes $H'_{k:6}$ from SM_k using the non-uniform advice. In Stage 2-2, R^* chooses $\boldsymbol{b}^R = (b_1^R, \dots, b_{11n}^R)$ in such a way that $\boldsymbol{r}^R = (r_1^R, \dots, r_{11n}^R)$ satisfies $r_i^R = c_i \parallel \tau_i^R$ for every $i \in \Delta \setminus \Gamma_R$, namely, chooses $\boldsymbol{b}^R$ such that $b_i^R = a_i^R \oplus (c_i \parallel \tau_i^R)$ for every $i \in \Delta \setminus \Gamma_R$. (Recall that in $H'_{k:6}$, the subset Γ_R and the strings $\boldsymbol{a}^R = (a_1^R, \dots, a_{11n}^R)$ are extracted by brute force and they are included in the non-uniform advice.) In Stage 3 of session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ obtains the i-th mS-OT from the external sender for every $i \in \Delta \setminus \Gamma_R$ and executes other instances of mS-OT by itself. Then, in Stage 6 of session $s(k)$, R^* receives bits $(s^*_{i,0}, s^*_{i,1})_{i\in\Delta\setminus\Gamma_R}$ from the external sender and uses them to compute $\beta_{b,i}$ for every $i \in \Delta \setminus \Gamma_R$, i.e., $\beta_{b,i} := \rho_{b,i} \oplus s^*_{i,b\oplus\alpha_i}$. After the execution of $H'_{k:6}$ finishes, R^* outputs whatever $\mathcal{Z}$ outputs in the experiment.
>
> When $b^* = 0$ in the security game (and hence $s^*_{i,b\oplus\alpha_i} = s_{i,b\oplus\alpha_i}$ for every i and b), the internally executed experiment is identical with $H_{k:6}$,

whereas when $b^* = 1$ (and hence $s^*_{i,b\oplus\alpha_i}$ is a random bit if $i \in \Delta \setminus I_b$ and $s^*_{i,b\oplus\alpha_i} = s_{i,b\oplus\alpha_i}$ otherwise), the internally executed experiment is identical with $H'_{k:6}$. Hence, from the assumption that $H_{k:6}$ and $H'_{k:6}$ are distinguishable, R^* wins the game with probability non-negligibly higher than $1/2$.

Next, we show that in $H'_{k:6}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. (The argument below is similar to the one in the proof of Lemma 2.) Assume for contradiction that in $H'_{k:6}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H_{k:6}$ but with non-negligible probability in $H'_{k:6}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the robust non-malleability of NMCom as follows.

The adversary $\mathcal{A}_{\mathsf{NMCom}}$, who participates in the above game while interacting with a receiver of NMCom, gets random input-randomness pairs $(c_i, \tau_i^R)_{i\in\Delta\setminus\Gamma_R}$ of mS-OT instances as its input, and internally executes $H'_{k:6}$ from SM_k using the non-uniform advice. In Stage 2-2, $\mathcal{A}_{\mathsf{NMCom}}$ chooses $\boldsymbol{b}^R = (b_1^R, \ldots, b_{11n}^R)$ in such a way that $\boldsymbol{r}^R = (r_1^R, \ldots, r_{11n}^R)$ satisfies $r_i^R = c_i \parallel \tau_i^R$ for every $i \in \Delta \setminus \Gamma_R$, namely, chooses $\boldsymbol{b}^R$ such that $b_i^R = a_i^R \oplus (c_i \parallel \tau_i^R)$ for every $i \in \Delta \setminus \Gamma_R$. In Stage 3 of session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ obtains the i-th mS-OT from the external sender for every $i \in \Delta \setminus \Gamma_R$ and executes other instances of mS-OT by itself. Then, in Stage 6 of session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ receives bits $(s^*_{i,0}, s^*_{i,1})_{i\in\Delta\setminus\Gamma_R}$ from the external sender and uses them to compute $\beta_{b,i}$ for every $i \in \Delta \setminus \Gamma_R$, i.e., $\beta_{b,i} := \rho_{b,i} \oplus s^*_{i,b\oplus\alpha_i}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver. After the execution of $H'_{k:6}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs its view.
The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
When $b^* = 0$ in the security game (and hence $s^*_{i,b\oplus\alpha_i} = s_{i,b\oplus\alpha_i}$ for every i and b), the internally executed experiment is identical with $H_{k:6}$, whereas when $b^* = 1$ (and hence $s^*_{i,b\oplus\alpha_i}$ is a random bit if $i \in \Delta \setminus I_b$ and $s^*_{i,b\oplus\alpha_i} = s_{i,b\oplus\alpha_i}$ otherwise), the internally executed experiment is identical with $H'_{k:6}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H_{k:6}$ but with non-negligible probability in $H'_{k:6}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the robust non-malleability of NMCom.

This completes the proof. □

D.8 Proof of Lemma 9

Proof (of Lemma 9). We first show the indistinguishability between $H_{k:7}$ and $H_{k:8}$. Assume for contradiction that $H_{k:7}$ and $H_{k:8}$ are distinguishable. From an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $H_{k:7}$ and $H_{k:8}$ are still distinguishable. Then, by considering the transcript (including the inputs and randomness of all the parties) and the extracted values up until SM_k as non-uniform advice, we can break the UC security of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$ as follows.

> The environment $\mathcal{Z}$ internally executes $H_{k:7}$ from SM_k using the non-uniform advice while externally participating in a single session of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$ via the dummy adversary that corrupts S. In session $s(k)$, $\mathcal{Z}$ forwards all the messages of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$ from the internal $\mathcal{A}$ to the external dummy adversary (including the query to $\mathcal{F}_{OT}$),[13] and those from the external dummy adversary to the internal $\mathcal{A}$. After the execution of $H_{k:7}$ finishes, $\mathcal{Z}$ outputs the output of the internally emulated experiment.
>
> When $\mathcal{Z}$ interacts with the dummy adversary, the internally executed experiment is identical with $H_{k:7}$, whereas when $\mathcal{Z}$ interacts with the simulator of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$, the internally executed experiment is identical with $H_{k:8}$. Hence, from the assumption that $H_{k:7}$ and $H_{k:8}$ are distinguishable, $\mathcal{Z}$ breaks the security of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$

We next show that in $H_{k:8}$, $\mathcal{A}$ does not cheat in sessions $s(k), \ldots, s(4m)$. Assume for contradiction that in $H_{k:8}$, $\mathcal{A}$ cheats in one of those sessions, say, session $s(j)$, with non-negligible probability. Then, from an average argument, we can fix the execution of the experiment up until SM_k (inclusive) in such a way that even after being fixed, $\mathcal{A}$ cheats in session $s(j)$ only with negligible probability in $H_{k:7}$ but with non-negligible probability in $H_{k:8}$. Then, by considering the transcript and the extracted values up until SM_k as non-uniform advice, we can break the robust non-malleability of NMCom as follows.

> The adversary $\mathcal{A}_{\mathsf{NMCom}}$, who participates in an execution of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$ as the environment (where the dummy adversary corrupts S) while interacting with a receiver of NMCom, internally executes $H_{k:7}$ from SM_k using the non-uniform advice. In session $s(k)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards all the messages of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$ from the internal $\mathcal{A}$ to the external dummy adversary (including the query to $\mathcal{F}_{OT}$), and those from the external dummy adversary to the internal $\mathcal{A}$. Also, in session $s(j)$, $\mathcal{A}_{\mathsf{NMCom}}$ forwards the NMCom commitments from $\mathcal{A}$ to the external receiver. After the execution of $H_{k:7}$ finishes, $\mathcal{A}_{\mathsf{NMCom}}$ outputs the output of the internally emulated experiment.
>
> The distinguisher $\mathcal{D}_{\mathsf{NMCom}}$ takes as input the view of $\mathcal{A}_{\mathsf{NMCom}}$ and the values committed by $\mathcal{A}_{\mathsf{NMCom}}$ (which are equal to the values committed to by $\mathcal{A}$ in session $s(j)$ in the internally executed experiment). $\mathcal{D}_{\mathsf{NMCom}}$ then outputs 1 if and only if $\mathcal{A}$ cheated in session $s(j)$.
>
> When $\mathcal{A}_{\mathsf{NMCom}}$ interacts with the dummy adversary in the execution of $\Pi_{\text{2PC}}^{\mathcal{F}_{\text{OT}}}$, the internally executed experiment is identical with $H_{k:7}$, whereas when $\mathcal{A}_{\mathsf{NMCom}}$ interacts with the simulator there, the internally executed

[13] Note that these messages appear after SM_k

experiment is identical with $H_{k:8}$. Hence, from the assumption that $\mathcal{A}$ cheats in session $s(j)$ with negligible probability in $H_{k:7}$ but with non-negligible probability in $H_{k:8}$, $\mathcal{A}_{\mathsf{NMCom}}$ breaks the robust non-malleability of NMCom.

This completes the proof of Lemma 9. □

E UC-secure OT-hybrid 2PC/MPC Protocol with Appropriate Properties

As stated in Section 5, the protocol of Ishai et al. [28] itself does not satisfy the property that is required for our purpose, but it can be modified to satisfy it. Specifically, we replace each invocation of $\mathcal{F}_{\mathrm{O}T}$ with the following protocol, which uses $\mathcal{F}_{\mathrm{O}T}$ with random inputs.

1. Let (v_0, v_1) be the sender's input, and u be the receiver's one.
2. The sender and the receiver invokes $\mathcal{F}_{\mathrm{O}T}$ with random input (s_0, s_1) and c.
3. The receiver sends $\alpha := u \oplus c$.
4. The sender sends $\beta_b := v_b \oplus s_{b \oplus \alpha}$ for each $b \in \{0, 1\}$.
5. The receiver outputs $\beta_u \oplus s_c$, where s_c is obtained from $\mathcal{F}_{\mathrm{O}T}$ in the second step.

Now, since $\mathcal{F}_{\mathrm{O}T}$ is invoked only with random input, it can be invoked at the beginning of the protocol.