# Round-optimal Black-box Commit-and-prove with Succinct Communication 

Susumu Kiyoshima<br>NTT Research<br>susumu.kiyoshima@ntt-research.com

July 6, 2020


#### Abstract

We give a four-round black-box construction of a commit-and-prove protocol with succinct communication. Our construction is WI and has constant soundness error, and it can be upgraded into a one that is ZK and has negligible soundness error by relying on a round-preserving transformation of Khurana et al. (TCC 2018). Our construction is obtained by combining the MPC-in-the-head technique of Ishai et al. (SICOMP 2009) with the two-round succinct argument of Kalai et al. (STOC 2014), and the main technical novelty lies in the analysis of the soundness-we show that, although the succinct argument of Kalai et al. does not necessarily provide soundness for $\mathcal{N} \mathcal{P}$ statements, it can be used in the MPC-in-the-head technique for proving the consistency of committed MPC views. Our construction is based on sub-exponentially hard collision-resistant hash functions, two-round PIRs, and two-round OTs.


[^0]
## Contents

1 Introduction ..... 4
1.1 Our Result ..... 5
1.2 Overview of Our Commit-and-prove Protocol ..... 6
2 Preliminaries ..... 7
2.1 Notations and Conventions ..... 7
2.2 Witness-indistinguishable Commit-and-prove Protocols ..... 7
2.3 Secure Multi-party Computation ..... 8
2.4 Probabilistically Checkable Proofs (PCPs) ..... 9
2.5 Definitions from Kalai et al. [KRR14] and Subsequent Works ..... 9
3 Outline of Proof of Theorem 1 ..... 10
3.1 Building Block: Perfect 2-private MPC protocol $\Pi$ ..... 10
4 Overview of Step 1 (Non-WI Scheme with Soundness against CNS Well-behaving Provers) ..... 11
4.1 Preliminary: Overview of Analysis of KRR No-signaling PCP ..... 11
4.2 Protocol Description ..... 12
4.3 Proof of Soundness ..... 12
5 Overview of Step 2 (Non-WI Scheme with Soundness against CNS Provers) ..... 17
6 Overview of Subsequent Steps of Proof of Theorem 1 ..... 18
A On Verification Time ..... 20
B On Definition of Commit-and-prove Protocols ..... 20
B. 1 Differences from Definition in Khurana et al. [KOS18] ..... 20
B. 2 Rationale behind Our Definition ..... 21
C Additional Preliminaries ..... 21
C. 1 Oblivious Transfer Protocols ..... 21
C. 2 Private Information Retrieval ..... 21
C. 3 Computational No-signaling (Parallel Version) ..... 22
C. 4 Low-degree Extension (LDE) ..... 22
C. 5 Threshold Verifiers ..... 22
D Step 0: PCP for Checking View Consistency ..... 22
D. 1 Preliminaries: Results from Kalai et al. [KRR14] and Subsequent Works ..... 22
D. 2 Our PCP system (PCP.P, PCP.V) ..... 25
E Step 1: Non-WI Scheme with (1 - negl)-Soundness against Well-behaving CNS Provers ..... 28
E. 1 Protocol Description ..... 28
E. 2 Proof of Binding ..... 28
E. 3 Proof of Soundness ..... 33
F Step 2: Non-WI Scheme with (1 - negl)-Soundness against CNS Provers ..... 39
F. 1 Protocol Description ..... 40
F. 2 Proof of Binding ..... 40
F. 3 Proof of Soundness ..... 42
G Step 3: Non-WI Scheme with negl-Soundness against CNS Provers ..... 43
G. 1 Protocol Description ..... 43
G. 2 Proof of Binding ..... 43
G. 3 Proof of Soundness ..... 45
H Step 4: Non-WI Scheme with Standard negl-Soundness ..... 46
H. 1 Preliminaries ..... 46
H. 2 Protocol Description ..... 47
H. 3 Proof of Binding ..... 47
H. 4 Proof of 2-Privacy ..... 49
H. 5 Proof of Soundness ..... 50
I Step 5: WI Scheme with Standard $O(1)$-Soundness ..... 51
I. 1 Protocol Description ..... 51
I. 2 Proof of Binding ..... 51
I. 3 Proof of Witness Indistinguishability ..... 53
I. 4 Proof of Soundness ..... 54
J ZK Scheme with Standard negl-Soundness ..... 57
J. 1 Protocol Description ..... 57
J. 2 Proof Sketch of Binding ..... 57
J. 3 Proof Sketch of Zero-knowledge ..... 58
J. 4 Proof Sketch of Soundness ..... 58
K Lemmas from Kalai et al. [KRR14] and Subsequent Works ..... 59
K. 1 Lemmas on SelfCorr ..... 59
K. 2 Soundness Amplification Lemma ..... 61

## 1 Introduction

In this paper, we obtain a new commit-and-prove protocol by relying on techniques in the area of succinct arguments. We start by giving some backgrounds.

Succinct arguments. Informally speaking, a succinct argument is an argument system with small communication complexity and fast verification time-typically, when a statement about $T$-time deterministic or non-deterministic computation is proven, the communication complexity and the verification time are required to be polylogarithmic in $T$. (The security requirements are, as usual, completeness and computational soundness.) Succinct arguments are useful when resources for communication and verification are limited; for example, a direct application of succinct arguments is delegating computation [GKR15] (or verifiable computation [GGP10]), where a computationally weak client delegates heavy computations to a powerful server and the client uses succinct arguments to verify the correctness of the server's computation efficiently. It was shown that a four-round succinct argument for all statements in $\mathcal{N} \mathcal{P}$ can be obtained from collision-resistance hash functions [Kil92]. Since then, succinct arguments have been actively studied, and protocols with various properties have been proposed.

Among existing succinct arguments, the most relevant to this work is the one by Kalai et al. [KRR14] (KRR succinct argument in short), which has several desirable properties such as (1) being doubly efficient [GKR15] (i.e., not only the verifier but also the prover is efficient), (2) being a two-round protocol (i.e., the scheme consists of a single query message from the verifier and a single answer message from the prover), and (3) being proven secure under standard assumptions, especially without relying on unfalsifiable assumptions and random oracles. More concretely, when the statement is about the correctness of a $T$-time computation, the communication complexity and the verifier running time is polylogarithmic in $T$ while the prover running time is polynomial in $T$, and the security is proven assuming the existence of private information retrieval (PIR) or fully homomorphic encryption (FHE).

Given the powerful properties of KRR succinct argument, it is natural to expect that it has many cryptographic applications. For example, since argument systems have been extensively used in the design of cryptographic protocols, one might expect that the efficiency of such cryptographic protocols can be improved by simply plugging in KRR succinct argument.

However, using KRR succinct argument in cryptographic applications is actually non-trivial. One difficulty is that the soundness of KRR succinct argument is currently proven only for some specific types of $\mathcal{N} \mathcal{P}$ statements [KP16, BHK17, $\mathrm{BKK}^{+} 18$ ] (originally, its soundness was proven for statements in $\mathcal{P}$ [KRR14]). Another difficulty is that it does not provide any privacy on witnesses when it is used for $\mathcal{N} \mathcal{P}$ statements.

Nonetheless, recent works showed that KRR succinct argument can be used in some cryptographic applications. For example, by cleverly combining KRR succinct argument with other cryptographic primitives, Bitansky et al. [BBK $\left.{ }^{+} 16\right]$ obtained a three-round zero-knowledge argument against uniform cheating provers, Brakerski and Kalai [BK20] obtained a succinct private access control protocol for the access structures that can be expressed by monotone formulas, and Morgan et al. [MPP20] obtained a succinct non-interactive secure two-party computation protocol.

The number of applications is, however, still limited. A potential reason for this limitation is that the current techniques inherently use cryptographic primitives in non-black-box ways. Concretely, to hide the prover's witness, the current techniques use KRR succinct argument under other cryptographic protocols (such as garbling schemes) and thus require non-black-box accesses to the codes of the cryptographic primitives that underlies KRR succinct argument. Consequently, the current techniques cannot be used for applications where black-box uses of cryptographic protocols are desirable, such as the application to commit-and-prove protocols, which we discuss next.

Commit-and-prove protocols. Informally speaking, a commit-and-prove protocol is a commitment scheme in which the committer can prove a statement about the committed value without opening the commitment. Proofs by the committer are required to be zero-knowledge ( ZK ) or witness-indistinguishable (WI), where the former requires that the views of the receiver in the commit and prove phases can be simulated in polynomial time without knowing the committed value, and the latter requires that for any two messages and any statement such that both of the messages satisfy the statement, the receiver cannot tell which of the messages is committed even after receiving a proof on the statement. Commit-and-prove protocols were implicitly used by Goldreich et al. [GMW87] for obtaining a secure multi-party computation protocol with malicious security, and later formalized by Canetti et al. [CLOS02].

A desirable property of commit-and-prove protocols is that they are constructed in a black-box way, i.e., in a way that uses the underlying cryptographic primitives as black-box by accessing them only through their input/output interfaces. Indeed, this black-box construction property is essential when commit-and-prove protocols are used as a tool for enforcing honest behaviors on malicious parties without relying on non-black-box uses of the underlying cryptographic primitives (see, e.g., [GLOV12, LP12, GOSV14]).

Very recently, Hazay and Venkitasubramaniam [HV18] and Khurana et al. [KOS18] gave four-round black-box constructions of ZK commit-and-prove protocols, where the round complexity of a commit-and-prove protocol is defined as the sum of that of the commit phase and that of the prove phase. Their protocols are round optimal since the commit and prove phases of their commit-and-prove protocols can be thought of as black-box ZK arguments (where the prover first
commits to a witness and then proves the validity of the committed witness) and black-box ZK arguments are known to require at least four rounds [GK96]. Their protocols also have the delayed-input property, i.e., the property that statements to be proven on committed values can be chosen adaptively in the last round of the prove phase.

The commit-and-prove protocols by Hazay and Venkitasubramaniam [HV18] and Khurana et al. [KOS18] are not succinct in the sense that when the statement is expressed as a $T$-time predicate on the committed value, the communication complexity depends at least linearly on $T$. This is because both of their protocols were obtained via transformations from the three-round constant-sound commit-and-prove protocol of Hazay and Venkitasubramaniam [HV16], which is not succinct in the above sense.

### 1.1 Our Result

Our main result is a four-round black-box construction of a constant-sound WI commit-and-prove protocol with succinct communication complexity.

Theorem 1. Assume the existence of sub-exponentially hard versions of the following cryptographic primitives: a collisionresistant hash function family, a two-round oblivious transfer protocol, and a two-round private information retrieval protocol. Then, there exists a constant-sound WI commit-and-prove protocol with the following properties.

1. The round complexity is 4 , and the protocol satisfies the delayed-input property and uses the above cryptographic primitives in a black-box way.
2. When the length of the committed value is $n$ and the statement to be proven on the committed value is a $T$-time predicate, the communication complexity depends polynomially on $\log n, \log T$, and the security parameter.

Our commit-and-prove protocol uses a variant of KRR succinct argument (which is obtained from the private information retrieval protocol), and succinctness of our commit-and-prove protocol is inherited from that of KRR succinct argument. We assume sub-exponential hardness on the cryptographic primitives since we use complexity leveraging.

ZK and negligible soundness error. Given our constant-sound WI commit-and-prove protocol, we can use (a minor variant of) a transformation of Khurana et al. [KOS18] to transform it into a 4-round ZK commit-and-prove protocol with negligible soundness error. The resultant commit-and-prove protocol still satisfies the delayed-input property, the black-box uses of the underlying primitives, and the succinct communication complexity. (See Appendix J for details.)

Verification time. The verification of our commit-and-prove protocol is not succinct, i.e., the verifier running time depends polynomially on $T$. Although we might be able to make it succinct by appropriately modifying our protocol (see Appendix A for details), we do not explore this possibility in this work so that we can focus on our main purpose, i.e., on showing how to use KRR succinct argument in black-box constructions of commit-and-prove protocols.

Complexity leveraging. As mentioned above, we use complexity leveraging in the proof of Theorem 1. Although we might be able to avoid the use of complexity leveraging by using known techniques (e.g., by relying on extractable commitments [PW09]), we do not explore this possibility in this work for the same reason as above.

Comparison with existing schemes. As explained above, Hazay and Venkitasubramaniam [HV18] and Khurana et al. [KOS18] gave four-round black-box ZK commit-and-prove protocols with the delayed-input property. Their schemes rely on a weak primitive (injective one-way functions) but do not have succinct communication.

Goyal et al. [GOSV14] and Ishai and Weiss [IW14] studied black-box commit-and-prove protocols with succinct communication under slightly different definitions than ours. ${ }^{1}$ If their techniques are used to obtain schemes under our definitions, the resultant schemes will rely on a weak primitive (collision-resistant hash functions) but have round complexity larger than $4 .{ }^{2}$

Kalai and Paneth [KP16] observed that when messages are committed by using Merkle tree-hash, KRR succinct argument can be used for proving statements on the committed messages. The resultant scheme is succinct in terms of both communication complexity and verification time, but uses the underlying hash function in a non-black-box way and does not have privacy properties (which are not needed for the purpose of [KP16]).

[^1]
### 1.2 Overview of Our Commit-and-prove Protocol

The overall approach is to combine KRR succinct argument with the MPC-in-the-head technique [IKOS09].
Let us first recall how we can obtain a non-succinct WI commit-and-prove protocol by using the MPC-in-the-head technique. Let $M \in \mathbb{N}$ be an arbitrary constant, $\Pi$ be any 2 -private semi-honest secure $M$-party computation protocol with perfect completeness, ${ }^{3}$ OT be any two-round 1-out-of- $M^{2}$ oblivious transfer (OT) protocol, SBCom be any statistically binding commitment scheme, and SHCom be any statistically hiding commitment scheme. We assume that the hiding property of SBCom can be broken in a quasi-polynomial time $T_{\mathrm{SB}}$, and the security of the other primitives holds against poly $\left(T_{\mathrm{SB}}\right)$-time adversaries.
Commit phase. To commit to a message $x_{\mathrm{CoM}}$, the committer (1) chooses random $x_{\mathrm{MPC}}^{1}, \ldots, x_{\mathrm{MPC}}^{M}$ such that $x_{\mathrm{MPC}}^{1} \oplus \cdots \oplus x_{\mathrm{MPC}}^{M}=$ $x_{\mathrm{COM}}$, (2) chooses randomness $r_{\mathrm{MPC}}^{1}, \ldots, r_{\mathrm{MPC}}^{M}$ for the $M$ parties of $\Pi$, and (3) commits to $\mathrm{st}_{0}^{\mu}:=\left(x_{\mathrm{MPC}}^{\mu}, r_{\mathrm{MPC}}^{\mu}\right)$ for each $\mu \in[M]$ by using SHCom. (Note that each st ${ }_{0}^{\mu}$ can be thought of as an initial state of a party of $\Pi$.) For each $\mu \in[M]$, let $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ denote the decommitment of SHCom for revealing st ${ }_{0}^{\mu}$.

Prove phase. In the first round, the receiver computes a receiver message of OT by using random $(\alpha, \beta) \in[M] \times[M]$ as the input, ${ }^{4}$ and sends it to the committer.
In the second round, to prove $f\left(x_{\text {Сом }}\right)=1$ for a predicate $f$, the committer does the following. (1) Execute $\Pi$ in the head by using $\mathrm{st}_{0}^{1}, \ldots, \mathrm{st}_{0}^{M}$ as the initial states of the $M$ parties and using $f^{\prime}:\left(y^{1}, \ldots, y^{M}\right) \mapsto f\left(y^{1} \oplus \cdots \oplus y^{M}\right)$ as the functionality to be computed. Let view ${ }^{1}, \ldots$, view $^{M}$ be the views of the parties in this execution of $\Pi$. (2) For each $\mu \in[M]$, compute a commitment to $\left(\operatorname{dec}_{\mathrm{sH}}^{\mu}, \mathrm{view}^{\mu}\right)$ by using SBCom. Let dec $\mathrm{S}_{\mathrm{sB}}^{\mu}$ be the decommitment of SBCom for revealing $\left(\operatorname{dec}_{\mathrm{sH}}^{\mu}\right.$, view $\left.^{\mu}\right)$. (3) Compute a sender message of OT by using $\left\{\left(\operatorname{dec}_{\mathrm{sB}}^{\mu}, \operatorname{dec}_{\mathrm{sB}}^{\nu}\right)\right\}_{\mu, v \in[M]}$ as the input. (4) Send the commitments and the OT message to the receiver.
In the verification, the receiver (1) recovers $\operatorname{dec}_{\mathrm{sB}}^{\alpha}$, $\operatorname{dec}_{\mathrm{sB}}^{\beta}$ from the OT message, (2) checks that they are valid decommitments of SBCom for revealing $\operatorname{dec}_{\mathrm{sH}}^{\alpha}, \mathrm{view}^{\alpha}, \operatorname{dec}_{\mathrm{sH}}^{\beta}, \mathrm{view}^{\beta}$ and that $\operatorname{dec}_{\mathrm{SH}}^{\alpha}, \operatorname{dec}_{\mathrm{sH}}^{\beta}$ are valid decommitments of SHCom for revealing $\mathrm{st}_{0}^{\alpha}$, $\mathrm{st}_{0}^{\beta}$, and (3) checks the following two conditions on $\mathrm{st}_{0}^{\alpha}, \mathrm{view}^{\alpha}$, $\mathrm{st}_{0}^{\beta}$, view ${ }^{\beta}$.

1. The views view ${ }^{\alpha}$, view $^{\beta}$ are consistent in the sense that the messages that the party $P^{\alpha}$ receives from the party $P^{\beta}$ in view ${ }^{\alpha}$ is equal to the messages that $P^{\beta}$ sends to $P^{\alpha}$ in view ${ }^{\beta}$ and vice versa.
2. For each $\xi \in\{\alpha, \beta\}$, the view view ${ }^{\xi}$ indicates that the initial state of $P^{\xi}$ is st ${ }_{0}^{\xi}$ and the output is 1 .

First, the constant soundness follows from the receiver security of OT and the perfect completeness of $\Pi$. Roughly speaking, this is because (1) the receiver security of OT guarantees that the committer can convince the verifier with high probability only when it commits to initial states and views that satisfy the above two conditions for every $\alpha, \beta \in[M],{ }^{5}$ and (2) when the committed initial states and views satisfy the above two conditions for every $\alpha, \beta \in[M]$, the perfect completeness of $\Pi$ guarantees $f\left(x_{\text {сом }}\right)=1$, where $x_{\text {сом }}$ is derived from the committed initial states. Next, the witness-indistinguishability follows from the receiver security of OT and the 2-privacy of $\Pi$. This is because the former guarantees that the receiver only learns the committed initial states and views of two parties and the latter guarantees the committed initial states and views of any two parties do not reveal any information about $x_{\text {сом }}$. Finally, this scheme is not succinct since the committer sends the initial states and views of $\Pi$ (or more precisely the decommitments to them) via OT.

Now, to make the above scheme succinct, we combine it with KRR succinct argument. The idea is to let the committer send succinct arguments about the initial states and views (instead of the initial states and views themselves) via OT. That is, we let the committer prove that the above two conditions hold on the committed initial states and views of each pair of the parties, where a separate instance of KRR succinct argument is used for each pair of the parties, and let it send the resultant $M^{2}$ succinct arguments via OT. (Note that KRR succinct argument can naturally be combined with OT since it is a two-round protocol.) As a minor modification, we also let the committer use a succinct commitment scheme to commit to the initial states and the views.

Unfortunately, although the modifications are intuitive, proving the soundness of the resultant scheme is non-trivial. (In contrast, the WI property can be proven similarly to the WI property of the original scheme. The key point is that, although KRR succinct argument does not provide any witness privacy, we can still prove WI of the whole scheme since in each instance of KRR succinct argument, the witness-initial states and views of a pair of the parties-does not reveal any secret information anyway.)

A natural approach for proving the soundness would be to first prove the soundness of each instance of KRR succinct argument and then derive the soundness of the whole scheme from it. Indeed, if we can show that each of the $M^{2}$ instances of KRR succinct argument provides an argument-of-knowledge property (which allows us to extract the committed initial states and views from the cheating committer), we can easily prove the soundness of the whole scheme.

The problem of this approach is that KRR succinct argument is not known to provide soundness for all statements in $\mathcal{N} \mathcal{P}$, and hence, does not necessarily provide soundness when it is used as above.

[^2]Our actual approach is to show that, while each of the instances of KRR succinct argument does not necessarily provide soundness, they as a whole provide a meaningful notion of the soundness, which can be used to prove the soundness of the whole scheme. Specifically, by getting into the security proof of the soundness of KRR succinct argument, we show that when $M^{2}$ instances of KRR argument are used in parallel for proving the consistency of each pair of the committed views etc. as above, then they as a whole guarantee that the committed views are mutually consistent etc.

We give more detailed overviews of our approach from Section 3 to Section 6 after giving necessary definitions in Section 2.

## 2 Preliminaries

We assume familiarity with basic cryptographic primitives. Several additional definitions are given in Appendix C.

### 2.1 Notations and Conventions

We denote the security parameter by $\lambda$. We assume that every algorithm takes the security parameter as input, and often do not write it explicitly.

We identify a bit-string with a function in the following manner: a bit-string $x=\left(x_{1}, \ldots, x_{n}\right)$ is thought of as a function $x:[n] \rightarrow\{0,1\}$ such that $x(i)=x_{i}$. More generally, for any finite field $\boldsymbol{F}$, we identify a string over $\boldsymbol{F}$ with a function in the same manner. For a vector $\boldsymbol{v}=\left(v_{1}, \ldots, v_{n}\right)$ and a set $S \subseteq[n]$, we define $\left.\boldsymbol{v}\right|_{S}$ by $\left.\boldsymbol{v}\right|_{S}:=\left\{v_{i}\right\}_{i \in S}$. Similarly, for a function $f: D \rightarrow R$ and a set $S \subseteq D$, we define $\left.f\right|_{S}$ by $\left.f\right|_{S}:=\{f(i)\}_{i \in S}$.

For any two probabilistic interactive Turing machines $A$ and $B$ and any input $x_{A}$ to $A$ and $x_{B}$ to $B$, we denote by (out ${ }_{A}$, out ${ }_{B}$ ) $\leftarrow\left\langle A\left(x_{A}\right), B\left(x_{B}\right)\right\rangle$ that the output of an interaction between $A\left(x_{A}\right)$ and $B\left(x_{B}\right)$ is (out ${ }_{A}$, out ${ }_{B}$ ), where out ${ }_{A}$ is the output from $A$ and out ${ }_{B}$ is the output from $B$.

### 2.2 Witness-indistinguishable Commit-and-prove Protocols

We give the definition of witness-indistinguishable commit-and-prove protocols. Our definition is based on the definition by Khurana et al. [KOS18] but is slightly different from it; see Appendix B for the differences.

A witness-indistinguishable (WI) commit-and-prove protocol $\langle C, R\rangle$ is a protocol between a committer $C=$ (C.Com, C.Dec, C.Prv) and a receiver $R=$ (R.Com, R.Dec, R.Prv), and it consists of three phases.

1. In the commit phase, C.Com takes a message $x \in\{0,1\}^{n}$ as input and interacts with R.Com to commit to $x$. ${ }^{6}$ At the end of the interaction, C.Com outputs its internal state $s t_{C}$ and R.Com outputs the commitment com, which is the transcript of the commit phase.
2. In the prove phase, C.Prv takes a predicate $f$ as input along with $\mathrm{st}_{C}$, and interacts with R.Prv to prove that $f(x)=1$ holds, where R.Prv takes (com, $f$ ) as input, At the end of the interaction, R.Prv outputs either 1 (accept) or 0 (reject).
3. In the open phase, C.Dec takes an index $i \in[n]$ as input along with $s t_{C}$, and interacts with R.Dec to reveal the $i$-th bit of $x$, where R.Dec takes (com, i) as input. At the end of the interaction, R.Dec outputs either a bit $x_{i}$ as the decommitted bit, or $\perp$ (reject).

In this paper, we focus on a WI commit-and-prove protocol such that (1) both the prove phase and the open phase consist of two rounds, (2) the first round of the prove phase does not depend on the commitment com and the predicate $f,{ }^{7}$ and (3) the first round of the open phase does not depend on the commitment com. Because of (1) and (2), R.Prv can be split into two algorithms, R.Prv.Q and R.Prv.D, such that the prove phase proceeds as follows: $\left(Q, \mathrm{st}_{R}\right) \leftarrow$ R.Prv.Q; $\pi \leftarrow$ C.Prv(st $\left.{ }_{C}, f, Q\right) ; b \leftarrow$ R.Prv.D(st $\left.{ }_{R}, \operatorname{com}, f, \pi\right)$. Similarly, because of (1) and (3), R.Dec can be split into two algorithms, R.Dec.Q and R.Dec.D, such that the open phase proceeds as follows: $\left(Q, \mathrm{st}_{R}\right) \leftarrow$ R.Dec. $\mathrm{Q}(i)$; dec $\leftarrow \mathrm{C} . \operatorname{Dec}\left(\mathrm{st}_{C}, i, Q\right)$; $b \leftarrow$ R.Dec.D(st ${ }_{R}$, com, dec).

WI commit-and-prove protocols need to satisfy the following security notions.
Definition 1 (Completeness). A commit-and-prove protocol $\langle C, R\rangle$ is complete if for any polynomial $n: \mathbb{N} \rightarrow \mathbb{N}$ and any $\lambda \in \mathbb{N}, x \in\{0,1\}^{n(\lambda)}$, and $i \in[n(\lambda)]$,

$$
\operatorname{Pr}\left[\begin{array}{l|l}
x_{i}=\tilde{x}_{i} & \begin{array}{l}
\left(\mathrm{st}_{C}, \operatorname{com}\right) \leftarrow\langle\operatorname{CC} \cdot \operatorname{Com}(x), \text { R.Com }\rangle \\
\left(\perp, \tilde{x}_{i}\right) \leftarrow\left\langle\mathrm{C} \cdot \operatorname{Dec}\left(\mathrm{st}_{C}, i\right), \text { R.Dec }(\operatorname{com}, i)\right\rangle
\end{array}
\end{array}\right]=1 .
$$

Definition 2 (Binding). A commit-and-prove protocol $\langle C, R\rangle$ is (computationally) binding if for any polynomial $n: \mathbb{N} \rightarrow$ $\mathbb{N}$, any pPT cheating committer $C^{*}=\left(\mathrm{C} . \mathrm{Com}^{*}, \mathrm{C} . \mathrm{Dec}^{*}\right)$, and any $\lambda \in \mathbb{N}$, the following binding condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}^{*}\right.$, R.Com $\rangle$.

[^3]- Binding Condition: For every $i \in[n(\lambda)]$, it holds $\operatorname{Pr}\left[b_{B A D}=1\right] \leq \operatorname{negl}(\lambda)$ in the following probabilistic experiment EXP ${ }^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}^{*}, \mathrm{st}_{C}\right.$, com, $i$ ).

1. For each $b \in\{0,1\}$, sample $Q_{b}$ by $\left(Q_{b}, \mathrm{st}_{b}\right) \leftarrow$ R.Dec. $\mathrm{Q}(i)$.
2. Run $\left\{\operatorname{dec}_{b}\right\}_{b \in\{0,1\}} \leftarrow \operatorname{C.Dec}{ }^{*}\left(\operatorname{st}_{C}, i,\left\{Q_{b}\right\}_{b \in\{0,1\}}\right)$.
3. For each $b \in\{0,1\}$, let $x_{b}^{*} \leftarrow$ R.Dec.D $\left(\mathrm{st}_{b}, \mathrm{com}, \mathrm{dec}_{b}\right)$.
4. Output $b_{B A D}:=1$ if and only if $x_{0}^{*} \neq \perp \wedge x_{1}^{*} \neq \perp \wedge x_{0}^{*} \neq x_{1}^{*}$ holds.

Definition 3 (Soundness). Let $\epsilon: \mathbb{N} \rightarrow[0,1]$ be a function. A commit-and-prove protocol $\langle C, R\rangle$ is (computationally) $\epsilon$-sound if for any constant $c \in \mathbb{N}$, there exists a PPT oracle Turing machine $E$ (called an extractor) such that for any polynomial $n: \mathbb{N} \rightarrow \mathbb{N}$, any PPT cheating committer $C^{*}=\left(\mathrm{C} . \mathrm{Com}^{*}, \mathrm{C} . \mathrm{Prv}^{*}\right)$, and any sufficiently large $\lambda \in \mathbb{N}$, the following soundness condition holds with overwhelming probability over the choice of ( $\left.\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}{ }^{*}, \mathrm{R} . \mathrm{Com}\right\rangle$.

- Soundness Condition ${ }^{8}$ : If it holds

$$
\left.\begin{array}{l|l}
\operatorname{Pr}[b=1 & \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q} ;(f, \pi) \leftarrow \operatorname{C\cdot } \cdot \operatorname{Prv}^{*}\left(\mathrm{st}_{C}, Q\right) ; \\
b \leftarrow \operatorname{R.Prv} \cdot \mathrm{D}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi\right)
\end{array}
\end{array}\right] \geq \epsilon(\lambda)+\frac{1}{\lambda^{c}},
$$

then there exists $x^{*}=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n(\lambda)}$ such that

$$
\forall i \in[n(\lambda)], \operatorname{Pr}\left[x_{i}=x_{i}^{*} \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E^{\mathrm{C} . \operatorname{Prv}^{*}\left(\mathrm{st}_{C}, \cdot\right)}(\operatorname{com}, i), \operatorname{R.Dec}(\operatorname{com}, i)\right\rangle\right] \geq 1-\operatorname{negl}(\lambda)
$$

and

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \operatorname{R.Prv.Q} ;(f, \pi) \leftarrow \operatorname{C.Prv}^{*}\left(\mathrm{st}_{C}, Q\right) ; \\
\wedge f\left(x^{*}\right)=0 & b \leftarrow \operatorname{R.Prv} \cdot \mathrm{D}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi\right)
\end{array}\right] \leq \epsilon(\lambda)+\operatorname{negl}(\lambda)
$$

$\langle C, R\rangle$ is said to be sound if it is $\epsilon$-sound for a negligible function $\epsilon$.
Definition 4 (Witness Indistinguishability). $\langle C, R\rangle$ is witness-indistinguishable if for any polynomial $n: \mathbb{N} \rightarrow \mathbb{N}$, any two sequences $\left\{x_{\lambda}^{0}\right\}_{\lambda \in \mathbb{N}}$ and $\left\{x_{\lambda}^{1}\right\}_{\lambda \in \mathbb{N}}$ such that $x_{\lambda}^{0}, x_{\lambda}^{1} \in\{0,1\}^{n(\lambda)}$, any PPT cheating receiver $R^{*}=\left(\right.$ R.Com ${ }^{*}$, R.Prv.Q*), the outputs of Experiment 0 and Experiment 1 are computationally indistinguishable.

- Experiment $b(b \in\{0,1\})$.

1. $\left(\mathrm{st}_{C}, \mathrm{st}_{R}\right) \leftarrow\left\langle\operatorname{C.Com}\left(x_{\lambda}^{b}\right), \operatorname{R.Com}{ }^{*}\left(x_{\lambda}^{0}, x_{\lambda}^{1}\right)\right\rangle$.
2. $\left(f, Q, \mathrm{st}_{R}^{\prime}\right) \leftarrow$ R.Prv. $Q^{*}\left(\mathrm{st}_{R}\right)$. If $f\left(x_{\lambda}^{0}\right) \neq 1$ or $f\left(x_{\lambda}^{1}\right) \neq 1$, abort.
3. $\pi \leftarrow \mathrm{C} \cdot \operatorname{Prv}\left(\mathrm{st}_{C}, f, Q\right)$.
4. Output $\left(\mathrm{st}_{R}^{\prime}, \pi\right)$.

### 2.3 Secure Multi-party Computation

We recall the definition of secure multi-party computation (MPC) protocols based on the description by Ishai et al. [IKOS09]. (We assume that the readers are familiar with the concept of secure MPC protocols.)

The basic model that is used in this paper is the following. The number of parties is denoted by $M$. We focus on MPC protocols that realize any deterministic $M$-party functionality that outputs a single bit (which is obtained by all the parties), given the synchronous communication over secure point-to-point channels. We assume that every party implicitly takes as input the $M$-party functionality to be computed.

Recall that the view of a party in an execution of an MPC protocol consists of its input, its randomness, and all the incoming messages that it received from the other parties during the execution of the protocol. The consistency between a pair of views is defined as follows.
Definition 5 (Consistent Views). A pair of views $\mathrm{view}^{i}{\text {, } \mathrm{view}^{j} \text { is consistent (w.r.t. an MPC protocol } \Pi \text { for a functionality }}^{\text {a }}$ f) if the outgoing messages that are implicitly reported in view ${ }^{i}$ are identical to the incoming messages that are reported in view ${ }^{j}$ and vice versa.

We consider security against semi-honest adversaries. Concretely, we use the following two security notions.
Definition 6 (Perfect correctness). We say that an MPC protocol $\Pi$ satisfies perfect correctness if for any deterministic $M$-party functionality $f$ and for any private inputs to the parties, the probability that the output of some party in an honest execution of $\Pi$ is different from the output of $f$ is 0 .
Definition 7 (2-privacy). We say that an MPC protocol $\Pi$ satisfies perfect 2-privacy if for any deterministic M-party functionality $f$, there exists a PPT simulator $\mathcal{S}_{\text {MPC }}$ such that for any private inputs $x_{1}, \ldots, x_{M}$ to the parties and every pair of corrupted parties, $T \subset[M]$ such that $|T|=2$, the joint view $\operatorname{View}_{T}\left(x_{1}, \ldots, x_{M}\right)$ of the parties in $T$ is identically distributed with $\mathcal{S}_{\text {MPC }}\left(T,\left\{x_{i}\right\}_{i \in T}, f\left(x_{1}, \ldots, x_{M}\right)\right)$.

[^4]
### 2.4 Probabilistically Checkable Proofs (PCPs)

We recall the definition of probabilistically checkable proofs (PCPs) based on the description by Brakerski et al. [BHK17]. Roughly speaking, PCPs are proof systems with which one can probabilistically verify the correctness of statements by reading only a few bits or symbols of the proof strings. A formal definition is given below.

Definition 8. A к-query PCP system $(\mathrm{P}, \mathrm{V})$ for an NP language $L$, where $\mathrm{V}=(\mathrm{Q}, \mathrm{D})$, satisfies the following.
-(Completeness) For all $\lambda \in \mathbb{N}$ and $x \in L$ (with witness $w$ ) such that $|x| \leq 2^{\lambda}$,

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\mathrm{D}\left(\mathrm{st}, x,\left.\pi\right|_{Q}\right)=1 & \begin{array}{l}
(Q, \mathrm{st}) \leftarrow \mathrm{Q}\left(1^{\lambda}\right) \\
\pi \leftarrow \mathrm{P}\left(1^{\lambda}, x, w\right)
\end{array}
\end{array}\right]=1 .
$$

The PCP proof $\pi$ is a string of characters over some alphabet $\Sigma$, and it can be thought that this string is indexed by a set $\Gamma$ (by identifying $\Gamma$ with $[N]$ in a canonical way, where $N$ is the length of the string) and $Q \subseteq \Gamma$. Alternatively, $\pi$ can be thought of as a function from $\Gamma$ to $\Sigma$.

- (Soundness) For all $\lambda \in \mathbb{N}$, all $x \notin L$ such that $|x| \leq 2^{\lambda}$, and all proof string $\pi^{*}$,

$$
\operatorname{Pr}\left[\mathrm{D}\left(\mathrm{st}, x,\left.\pi^{*}\right|_{Q}\right) \mid \quad(Q, \mathrm{st}) \leftarrow \mathrm{Q}\left(1^{\lambda}\right)\right] \leq \frac{1}{2} .
$$

- (Query Efficiency) If $(Q, \mathrm{st}) \leftarrow \mathrm{Q}\left(1^{\lambda}\right)$, then $|Q| \leq \kappa(\lambda)$ and the combined run-time of Q and D is poly $(\lambda)$.
- (Prover Efficiency) The prover P runs in polynomial time, where its input is $\left(1^{\lambda}, x, w\right)$.


### 2.5 Definitions from Kalai et al. [KRR14] and Subsequent Works

### 2.5.1 Computational no-signaling (CNS).

We recall the definition of adaptive (computational) no-signaling [KRR14, BHK17].
Definition 9. Fix any alphabet $\left\{\Sigma_{\lambda}\right\}_{\lambda \in \mathbb{N}}$, any $\left\{N_{\lambda}\right\}_{\lambda \in \mathbb{N}}$ such that $N_{\lambda} \in \mathbb{N}$, any function $\kappa_{\max }: \mathbb{N} \rightarrow \mathbb{N}$ such that $\kappa_{\max }(\lambda) \leq N_{\lambda}$, and any algorithm Algo such that for any $\lambda \in \mathbb{N}$, on input a subset $Q \subset\left[N_{\lambda}\right]$ of size at most $\kappa_{\max }(\lambda)$, Algo outputs (the truth table of) a function $A: Q \rightarrow \Sigma \cup\{\perp\}$ with an auxiliary output out.

Then, the algorithm Algo is adaptive $\kappa_{\max }$-computational no-signaling (CNS) if for any pPT distinguisher $\mathcal{D}$, any sufficiently large $\lambda \in \mathbb{N}$, any $Q, S \subset\left[N_{\lambda}\right]$ such that $Q \subseteq S$ and $|S| \leq \kappa_{\max }(\lambda)$, and any $z \in\{0,1\}^{\mathrm{poly}(\lambda)}$,

$$
\mid \operatorname{Pr}[\mathcal{D}(\text { out }, A, z)=1 \mid(\text { out }, A) \leftarrow \operatorname{Algo}(Q)]-\operatorname{Pr}\left[\mathcal{D}\left(\text { out },\left.A\right|_{Q}, z\right)=1 \mid(\text { out }, A) \leftarrow \operatorname{Algo}(S)\right] \mid \leq \operatorname{negl}(\lambda) .
$$

We remark that the above definition can be naturally extended for the case that Algo takes auxiliary inputs, as well as for the case that Algo takes multiple subsets as input and then outputs multiple functions (see Section C.3) .

### 2.5.2 Adaptive local assignment generator.

We recall the definition of adaptive local assignment generators [PR14, BHK17].
Definition 10. For any function $\kappa_{\max }: \mathbb{N} \rightarrow \mathbb{N}$, an adaptive $\kappa_{\text {max }}$-local assignment generator Assign on variables $\left\{V_{\lambda}\right\}_{\lambda \in \mathbb{N}}$ is an algorithm that takes as input a security parameter $1^{\lambda}$ and a set of at most $\kappa_{\max }(\lambda)$ queries $W \subseteq\left\{1, \ldots,\left|V_{\lambda}\right|\right\}$, and outputs a 3CNF formula $\varphi$ on variables $V_{\lambda}$ and assignments $A: W \rightarrow\{0,1\}$ such that the following two properties hold.

- Everywhere Local Consistency. For every $\lambda \in \mathbb{N}$ and every set $W \subseteq\left\{1, \ldots,\left|V_{\lambda}\right|\right\}$ such that $|W| \leq \kappa_{\max }(\lambda)$, with probability at least $1-\operatorname{negl}(\lambda)$ over sampling $(\varphi, A) \leftarrow \operatorname{Assign}\left(1^{\lambda}, W\right)$, the assignment $A$ is "locally consistent" with the formula $\varphi$. That is, for any $i_{1}, i_{2}, i_{3} \in W$, if $\varphi$ has a clause whose variables are $v_{i_{1}}, v_{i_{2}}, v_{i_{3}}$, then this clause is satisfied with the assignment $A\left(i_{1}\right), A\left(i_{2}\right), A\left(i_{3}\right)$ with probability at least $1-\operatorname{negl}(\lambda)$.
- Computational No-signaling. Assign is adaptive $\kappa_{\max }-C N S$.


### 2.5.3 No-signaling PCPs.

We recall the definition of (computational) no-signaling PCPs [KRR14, BHK17]. Essentially, no-signaling PCPs are PCP systems that are sound against no-signaling cheating provers. Specifically, for any function $\kappa_{\max }: \mathbb{N} \rightarrow \mathbb{N}$, a PCP system $(\mathrm{P}, \mathrm{V})$ for a language $L$, where $\mathrm{V}=(\mathrm{Q}, \mathrm{D})$, is adaptive $\kappa_{\max }$-no-signaling sound with negligible soundness error if it satisfies the following.

- (No-signaling Soundness) For any adaptive $\kappa_{\max }$-CNS cheating prover $P^{*}$ and any $\lambda \in \mathbb{N}$,

$$
\operatorname{Pr}\left[\begin{array}{l|l}
x^{*} \notin L \wedge \mathrm{D}\left(\mathrm{st}, x^{*}, \pi^{*}\right)=1 & \underset{\left(x^{*}, \pi^{*}\right) \leftarrow P^{*}\left(1^{\lambda}, Q\right)}{(Q, \mathrm{st}) \leftarrow \mathrm{Q}\left(1^{\lambda}\right)}
\end{array}\right] \leq \operatorname{negl}(\lambda) .
$$

## 3 Outline of Proof of Theorem 1

As mentioned in Section 1.2, our commit-and-prove protocol uses the succinct argument of Kalai et al. [KRR14] (KRR succinct argument in short). Unfortunately, we do not use it modularly-we slightly modify a building block of KRR succinct argument (namely, their no-signaling PCP system) when constructing our protocol, and we see low-level parts of the analysis of KRR succinct argument when analyzing our protocol.

At a high level, KRR succinct argument is obtained in three steps, starting from a scheme with a weak soundness notion.

1. Obtain a PCP system such that no CNS adversary can break the soundness with overwhelming success probability.
2. Obtain a PCP system such that no CNS adversary can break the soundness with non-negligible success probability.
3. Obtain a succinct argument such that no adversary can break the soundness with non-negligible success probability.

Somewhat similarly, our commit-and-prove protocol is obtained in five steps, starting from a non-WI scheme with a weak soundness notion.

1. Obtain a non-WI scheme, $\left\langle C_{1}, R_{1}\right\rangle$, such that no CNS "well-behaving" adversary can break the soundness with overwhelming success probability. (Well-behaving adversaries is the class of adversaries that we introduce later.)
2. Obtain a non-WI scheme, $\left\langle C_{2}, R_{2}\right\rangle$, such that no CNS adversary can break the soundness with overwhelming success probability.
3. Obtain a non-WI scheme, $\left\langle C_{3}, R_{3}\right\rangle$, such that no CNS adversary can break the soundness with non-negligible success probability.
4. Obtain a non-WI scheme, $\left\langle C_{4}, R_{4}\right\rangle$, such that no adversary can break the soundness with non-negligible success probability.
5. Obtain a WI scheme, $\left\langle C_{5}, R_{5}\right\rangle$, such that no adversary can break the soundness with constant success probability.

The most technically interesting step is the first step, and an extensive overview of this step is given in Section 4. Overviews of the other steps are given in Section 5 and Section 6. The formal proof is given in appendices (from Appendix D to Appendix I).

### 3.1 Building Block: Perfect 2-private MPC protocol $\Pi$

In addition to the cryptographic primitives that are listed in Theorem 1, we use a 2-private semi-honest secure $M$-party computation protocol $\Pi$ with perfect completeness, where $M$ is an arbitrary constant. (Note that such an MPC protocol can be obtained unconditionally; cf. Footnote 3.) We denote the parties of $\Pi$ by $P^{1}, \ldots P^{M}$.

Simplifying assumptions on $\Pi$. For editorial simplicity, we make several simplifying assumptions on $\Pi$.

- The length of the initial state of each party is denoted by $n_{\mathrm{st}}=n+n_{\text {MPC }}$, where $n$ is the input length and $n_{\text {MPC }}$ is the randomness length, and each party has $n_{\text {st }}$-bit internal state at the beginning of each round.
- Every party uses the same next-message function in every round. ${ }^{9}$
- Every party sends a 1-bit message to each party at the end of each round.
- Every party receives dummy incoming messages from all the parties at the beginning of the first round, and every party sends a dummy outgoing message to itself at the end of each round. (This assumption is made so that the nextmessage function always takes an $\left(n_{\text {st }}+M\right)$-bit input, where the last $M$ bits are the concatenation of the incoming messages.)
- The first bit of the final state of each party denotes the output of that party.

[^5]
## 4 Overview of Step 1 (Non-WI Scheme with Soundness against CNS Wellbehaving Provers)

We give an extensive overview of our non-WI commit-and-prove protocol $\left\langle C_{1}, R_{1}\right\rangle$, which is ( $1-$ negl)-sound against CNS "well-behaving" provers. At a high level, we follow the approach that we outline in Section 1.2. That is, we implement the MPC-in-the-head technique with the MPC protocol $\Pi$ and a succinct argument. However, instead of using KRR succinct argument, we use a variant of the no-signaling PCP system (PCP. $P_{\text {KRR }}, P C P . V_{\text {KRR }}$ ) of Kalai et al. [KRR14] (which is the main building block of KRR succinct argument and is referred to as KRR no-signaling PCP in what follows), and we do not use any cryptographic primitives in this step so that we can focus on information theoretical arguments in the analysis. As a result, we can prove soundness only against very restricted provers, which we define as CNS well-behaving provers.

For simplicity, in this overview, we focus on static soundness, where the statement to be proven by the cheating prover is fixed at the beginning of the prove phase. We will also make several implicit oversimplifications in this overview.

### 4.1 Preliminary: Overview of Analysis of KRR No-signaling PCP

We start by briefly recalling the analysis of KRR no-signaling PCP (i.e., the analysis of its no-signaling soundness for statements in $\mathcal{P}$ ), focusing on the parts that are relevant to this work. ${ }^{10}$

We first remark that KRR no-signaling PCP is a PCP system for 3SAT, so at the beginning the statement to be proven is converted into a 3SAT instance. Specifically, given any statement in $\mathcal{P}$ of the form " $(f, x)$ satisfies $f(x)=1$ " for some public function $f$ and input $x$, first the function $f$ is converted into a carefully designed Boolean circuit C that computes $f$, and next the statement is converted into a 3SAT instance $\varphi$ that has the following properties.

1. $\varphi$ has a variable for each of the wires in C , and the values that are assigned to these variables are interpreted as an assignment to the corresponding wires in C .
2. The clauses of $\varphi$ checks that (1) for each gate in C , the assignment to its input and output wires is consistent with the computation of the gate, (2) the assignment to the input wires of C is equal to $x$, and (3) the assignment to the output wire of $C$ is equal to 1 .

Now, the analysis of KRR no-signaling PCP roughly consists of three parts.
The first part of the analysis shows that any successful CNS cheating prover for a statement $(f, x)$ can be converted into a local assignment generator for the 3SAT instance $\varphi$ that is obtained from $(f, x)$ as above. That is, it shows that any successful CNS cheating prover can be converted into a probabilistic algorithm Assign such that (1) Assign takes as input a small-size subset of the variables of $\varphi$ and it outputs an assignment to these variables, and (2) Assign is guaranteed to satisfy the following everywhere local consistency.

Everywhere local consistency. Assign does not make an assignment that violates any clause of $\varphi$. Specifically, when Assign is asked to make an assignment to the three variables that appear in a clause of $\varphi$, it makes an assignment that satisfies this clause.
(Actually, Assign is also guaranteed to be CNS, but we ignore it in this overview for simplicity. ${ }^{11}$ ) We note that Assign does not necessarily comply with a single global assignment, that is, Assign can assign different values to the same variable depending on the randomness and the input. We also note that this part of the analysis holds even for statements in $\mathcal{N P}$. For simplicity, in this overview we assume that Assign does not err (i.e., the everywhere local consistency holds with probability 1 ).

The second part of the analysis shows that the local assignment generator Assign that is obtained in the first part is guaranteed to comply with a single global "correct" assignment. A bit more precisely, this part shows the following.

Let the correct assignment to a wire in C (or, equivalently, to a variable in $\varphi$ ) be defined as the assignment that is obtained by evaluating $C$ on $x$, and let Assign be called correct on a wire in $C$ (or variable in $\varphi$ ) if Assign makes the correct assignment to it whenever Assign is asked to make an assignment to it. Then, Assign is correct on any wire in $C$ ( or variable in $\varphi$ ), and in particular correct on the output wire of C .

Roughly speaking, the above is shown in two steps.

1. First, it is shown, by relying on a specific structure of $C$, that Assign is correct on any wire in $C$ if Assign is correct on each input wire of $C$.
2. Next, it is observed that Assign is indeed correct on each input wire of $C$ due to the everywhere local consistency and the definition of $\varphi$ (which has clauses that check that the assignment to the input wires of C is equal to $x$ ).
[^6]Finally, the last part of the analysis obtains the soundness by combining what are shown by the preceding two parts. In particular, it is observed that the existence of Assign as above implies $f(x)=1$ since (1) on the one hand, Assign always assigns 1 to the output wire of $C$ due to the everywhere local consistency and the definition of $\varphi$ (which has clauses that check that the assignment to the output wire of C is 1 ), and (2) on the other hand, Assign always assigns $f(x)$ to the output wire of C since what is shown by the second part implies that Assign is correct on the output wire of C .
Remark 1 (Difficulty in the case of $\mathcal{N} \mathcal{P}$ statements). The above analysis does not work in general for statements in $\mathcal{N} \mathcal{P}$. A difficulty is that when the statement is in $\mathcal{N P}$, it is unclear how we should define the correct assignment in the second part of the analysis. Indeed, on the one hand, the correct assignment can be naturally defined in the case of statements in $\mathcal{P}$ since there exists a unique assignment that any successful prover is supposed to use (namely the assignment that is derived from $x$ ); on the other hand, in the case of statements in $\mathcal{N} \mathcal{P}$, there does not exist a single such assignment. Jumping ahead, below we define well-behaving provers so that we can define the correct assignment naturally (while at the same time so that we can use cryptographic primitives later to force any prover to be well-behaving).

### 4.2 Protocol Description

In this overview, we consider the following protocol $\left\langle C_{1}, R_{1}\right\rangle=\left(\right.$ C.Com $_{1}$, C. Prv $_{1}$, R.Com ${ }_{1}$, R.Prv.Q ${ }_{1}$, R.Prv. $\left.D_{1}\right)$, which is slightly oversimplified from the actual protocol in Appendix E. (At this point, we temporarily ignore the open phase.) We warn that $\left\langle C_{1}, R_{1}\right\rangle$ is not biding at all in the standard sense since the committer sends no message in the commit phase.

## Commit Phase:

Round 1: Given $x_{\text {сом }}$ as the value to be committed, C.Com ${ }_{1}$ does the following.

1. Sample random $x_{\mathrm{MPC}}^{1}, \ldots, x_{\mathrm{MPC}}^{M}$ such that $x_{\mathrm{MPC}}^{1} \oplus \cdots \oplus x_{\mathrm{MPC}}^{M}=x_{\mathrm{CoM}}$.
2. For each $\mu \in[M]$, define $x_{1, \text { in }}^{\mu}$ as follows: sample random $r_{\mathrm{MPC}}^{\mu} \in\{0,1\}^{n_{\mathrm{MPc}}}$ and let $\mathrm{st}_{0}^{\mu}:=x_{\mathrm{MPC}}^{\mu} \| r_{\mathrm{MPC}}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}:=$ $0^{M}, x_{1, \mathrm{n}}^{\mu}:=\mathrm{st}_{0}^{\mu} \| \mathrm{i}-\mathrm{msgs}_{1}^{\mu}$.
3. Output an empty string as the commitment and store $\left\{x_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}$ as the internal state.

## Prove Phase:

Round 1 R.Prv. $Q_{1}$ does the following.

1. For each $\mu, v \in[M]$, obtain a set of queries $Q^{\mu, \nu}$ by running the verifier of KRR no-signaling PCP.
2. Output $\left\{Q^{\mu, \nu}\right\}_{\mu, v \in[M]}$ as the query.

Round 2: Given the statement $f$ and the query $\left\{Q^{\mu, \nu}\right\}_{\mu, v \in[M]}$ as input, C.Prv ${ }_{1}$ does the following.

1. Run the MPC protocol $\Pi$ in the head for functionality $f^{\prime}$ and initial states $\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i} \text {-msgs }{ }_{1}^{\mu}\right)\right\}_{\mu \in[M]}{ }^{12}$ where $f^{\prime}$ is defined as $f^{\prime}:\left(y^{1}, \ldots, y^{M}\right) \mapsto f\left(y^{1} \oplus \cdots \oplus y^{M}\right)$ and each $\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)$ is recovered from the internal state of the commit phase. Let $\left\{\text { view }^{\mu}\right\}_{\mu \in[M]}$ be the view of the parties in this execution.
2. For each $\mu, v \in[M]$, obtain a PCP proof $\pi^{\mu: v}$ by running the prover of KRR no-signaling PCP on the 3SAT instance $\varphi^{\mu: v}$ that we will carefully design later-roughly speaking, $\varphi^{\mu: v}$ takes views of the parties $P^{\mu}, P^{v}$ of $\Pi$ as input, and checks that the views are consistent and that $P^{\mu}$ and $P^{v}$ output 1 in the views. (In an honest execution, C.Prv ${ }_{1}$ uses (view ${ }^{\mu}$, view ${ }^{\nu}$ ) to obtain a satisfying assignment to $\varphi^{\mu: v}$ and then uses it to obtain $\pi^{\mu: \nu}$.)
3. Output $\left\{\left.\pi^{\mu: v}\right|_{Q^{\mu v:}}\right\}_{\mu, v \in[M]}$ as the proof.

Verification: Given the statement $f$ and the proof $\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}$ as input, R.Prv. $D_{1}$ does the following.

1. Verify each $\pi^{* \mu: v}$ by running the verifier of KRR no-signaling PCP, and let $b^{\mu: v}$ be the verification result.
2. Output 1 if and only if $b^{\mu: \nu}=1$ for every $\mu, v \in[M]$.

### 4.3 Proof of Soundness

We give an overview of the proof of the soundness. To focus on the main technical idea, in this overview we consider a weak version of the soundness where the extractor is only required to extract a committed value (rather than decommit the commitment as required in Definition 3). Thus, for any successful cheating prover, the extractor is required to extract a value such that the cheating prover cannot prove false statements on it.

[^7]
### 4.3.1 Overall approach.

At a very high level, the proof consists of two parts.
The first part is to obtain an extractor. Toward this end, we first observe that, by borrowing analyses from Kalai et al. [KRR14], we can convert any successful CNS cheating prover against $\left\langle C_{1}, R_{1}\right\rangle$ into a parallel local assignment generator p-Assign, which gives $M^{2}$ local assignments to the 3SAT instances $\left\{\varphi^{\mu: v}\right\}_{\mu, v \in[M]}$ in parallel when it is given $M^{2}$ subsets of the variables as input. (To see this, observe that the prove phase of $\left\langle C_{1}, R_{1}\right\rangle$ consists of $M^{2}$ parallel executions of KRR no-signaling PCP.) Then, we obtain an extractor by using $\mathrm{p}-\mathrm{Assign}$ as follows.

- Note that since each $\varphi^{\mu: v}$ is a 3SAT instance that takes views of $P^{\mu}, P^{v}$ as input, for any particular parts of $P^{\mu}$ and $P^{v}$ s views, $\varphi^{\mu: v}$ has variables that are supposed to be assigned with these parts. In the following, when we say that p -Assign makes an assignment to particular parts of $P^{\mu}$ and $P^{\nu}$ s views in $\varphi^{\mu: \nu}$, we mean that p -Assign makes an assignment to the variables that are supposed to be assigned with these parts in $\varphi^{\mu: v}$.
- Now, to extract the $i$-th bit of the committed value, the extractor obtains the $i$-th bit of each party's MPC input by asking p -Assign to make an assignment to the $i$-th bit of $P^{\mu}$ 's input in $\varphi^{\mu: \mu}$ for every $\mu \in[M]$, and then takes XOR of the obtained bits.

The second part is to show that any cheating prover cannot prove false statements on the extracted value. In this part, the analysis proceeds similarly to the analysis of KRR no-signaling PCP. That is, we first define the correct assignment for each of $\varphi^{\mu: v}$, and next show that p -Assign always makes the correct assignment to any variable in any of $\varphi^{\mu: v}$.

Unfortunately, we do not know how to prove the second part against CNS cheating provers in general, and thus, we further restrict the provers to be "well-behaving."

### 4.3.2 Well-behaving provers.

Roughly speaking, we define well-behaving provers as follows. Recall that the extractor is obtained by converting the cheating prover into a parallel local assignment generator. Now, we define well-behaving provers so that when we convert a successful CNS well-behaving prover into a parallel local assignment generator p -Assign, it satisfies the following two consistency properties.

Consistency on the initial states: Once the commit phase is completed, there exists a unique set of MPC initial states $\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}$ such that p -Assign always makes assignments that are consistent with it (i.e., for any $\mu, v \in[M]$, when p -Assign is asked to make an assignment to any bit of the initial state of $P^{\mu}$ or $P^{\nu}$ in $\varphi^{\mu: v}$, then p-Assign always assigns the corresponding bit of $\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)$ or $\left.\left(\mathrm{st}_{0}^{\nu}, \mathrm{i}-\mathrm{msgs}_{1}^{\nu}\right)\right)$.

Consistency on the views: For every $\mu, v, \xi \in[M]$, when p -Assign is asked to make an assignment to any bit of $P^{\mu}$ 's view in both $\varphi^{\mu: v}$ and $\varphi^{\mu: \xi}$, then the value that p -Assign assigns to it in $\varphi^{\mu: v}$ is identical with the value that p -Assign assigns to it in $\varphi^{\mu: \xi}$. (The same holds for $\varphi^{v: \mu}$ and $\varphi^{\xi: \mu}$ and for $\varphi^{\mu: v}$ and $\varphi^{\xi: \mu}$ etc.)

Remark 2 (Intuition of the two consistency properties of p-Assign). Essentially, the above two consistency properties guarantee that $p$-Assign behaves as if it were obtained from an honest prover. This is because when $p$-Assign is indeed obtained from an honest prover, we can show that p-Assign always assigns the same MPC initial states once the commit phase is fixed, and assigns the same $P^{\mu}$ 's view in any $\varphi^{\mu: \nu}$ and $\varphi^{\mu: \xi}$. (Roughly speaking, this is because in an honest execution of $\left\langle C_{1}, R_{1}\right\rangle$, a set of MPC initial states are fixed in the commit phase, and the same $P^{\mu}$ 's view is used for computing PCPs on any $\varphi^{\mu: v}$ and $\varphi^{\mu: \xi}$ in the prove phase.)

Before giving more details on the definition of well-behaving provers, we show that by restricting the provers to be well-behaving, we can complete the second part of the above overall approach, where our goal is to show that any cheating prover cannot prove false statements on the extracted value.

### 4.3.3 Showing that cheating prover cannot prove false statements.

As stated earlier, the analysis proceeds similarly to the analysis of KRR no-signaling PCP. That is, we first define the correct assignment for each of $\varphi^{\mu: \nu}$, and next show that p -Assign always makes the correct assignment to any variable in any of $\varphi^{\mu: \nu}$.

Step 1: Defining the correct assignments. We define the correct assignments for $\left\{\varphi^{\mu: v}\right\}_{\mu, v \in[M]}$ by relying on that p -Assign satisfies the consistency on the initial states. Recall that it guarantees that once the commit phase is completed, there exists a unique set of MPC initial states $\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}$ such that p -Assign always makes local assignments that are consistent with it. Then, we first define the correct views $\left\{\mathrm{view}^{\mu}\right\}_{\mu \in[M]}$ as the views that are obtained by executing $\Pi$ on these unique initial states $\left\{\left(\mathrm{St}_{0}^{\mu}, \mathrm{i}-\operatorname{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}$, and then define the correct assignment for $\varphi^{\mu: v}(\mu, v \in[M])$ as the assignment that is derived from the correct views (view ${ }^{\mu}$, view ${ }^{v}$ ) of $P^{\mu}, P^{v}$. (Recall that $\varphi^{\mu: v}$ is a 3SAT instance that takes views of $P^{\mu}$, $P^{v}$ as input.)

From the definition, it is clear that p-Assign is correct on the initial states in every $\varphi^{\mu: v}$ (i.e., p-Assign always assigns the correct assignment to any bit of the initial states of $P^{\mu}, P^{v}$ in $\varphi^{\mu i v}$ for every $\mu, v \in[M]$. Also, since the extractor extracts the committed value by taking XOR of the MPC inputs that are obtained from p-Assign, p-Assign's correctness on the initial states implies that the value that the extractor extracts is unique and is equal to the XOR of the MPC inputs that are used in the correct views.

Step 2: Showing that $p$-Assign is correct on every variable. At a high level, our approach is to apply the second part of the analysis of KRR no-signaling PCP (Section 4.1) on each party's next-message computation in a "round-by-round" manner. More concretely, our approach is to first show that $p$-Assign is correct on each of the variables that correspond to the internal states and incoming/outgoing messages of Round 1 of $\Pi$ in every $\varphi^{\mu: v}$, next show it on each of the variables that correspond to those of Round 2 of $\Pi$ in every $\varphi^{\mu: \nu}$, and so on.

Toward this end, we first remark that we design each 3SAT instance $\varphi^{\mu: v}$ carefully so that it has the following specific structure.

1. Let $N_{\text {round }}$ be the round complexity of $\Pi$. Then, $\varphi^{\mu: v}$ has variables that can be partitioned into $4 N_{\text {round }}$ sequences of variables, $\boldsymbol{w}_{1, \text { in }}^{\xi}, \boldsymbol{w}_{1, \text { out }}^{\xi}, \ldots, \boldsymbol{w}_{N_{\text {round }, \mathrm{i}}}^{\xi}, \boldsymbol{w}_{N_{\text {round }, \text { out }}^{\xi}}^{\xi}$ for $\xi \in\{\mu, v\}$, such that for each $\ell \in\left[N_{\text {round }}\right]$ :

- $\boldsymbol{w}_{\ell, \text { in }}^{\xi}$ is a sequence of variables such that the values that are assigned to them are interpreted as an internal state and incoming messages of $P^{\xi}$ at the beginning of Round $\ell{ }^{13}$
- $\boldsymbol{w}_{\ell, \text { out }}^{\xi}$ is a sequence of variables such that the values that are assigned to them are interpreted as an internal state and outgoing messages of $P^{\xi}$ at the end of Round $\ell$.

2. $\varphi^{\mu: v}$ has clauses that check the following.

- In each round, for each of $P^{\mu}$ and $P^{\nu}$, its end state (i.e., its internal state at the end of the round) and outgoing messages are correctly derived from its start state (i.e., its internal state at the beginning of the round) and incoming messages.
- In each round, for each of $P^{\mu}$ and $P^{\nu}$, its start state is equal to its end state of the previous round.
- In each round, $P^{\mu}$ 's incoming message from $P^{v}$ at the beginning of the round is equal to $P^{v}$,s outgoing message to $P^{\mu}$ at the end of the previous round, and vise versa.
- Both $P^{\mu}$ and $P^{v}$ output 1 in the last round.

We note that given consistent views of $P^{\mu}, P^{v}$ in which they output 1 , we can compute a satisfying assignment to the variables in $\varphi^{\mu: v}$ efficiently by obtaining each party's end state and outgoing messages of each round through the nextmessage function.

Now, we first show that if in every $\varphi^{\mu: \nu}$, p -Assign is correct on $P^{\mu}$ and $P^{\nu}$ 's start states and incoming messages in Round 1, then in every $\varphi^{\mu, v}$, p-Assign is also correct on $P^{\mu}$ and $P^{\nu}$ s end states and outgoing messages in Round 1. A key observation on this step is that, essentially, what we need to show is that in every $\varphi^{\mu: \nu}$, for each $\xi \in\{\mu, \nu\}$, if $p$-Assign is correct on the input of $P^{\xi}$ 's next-message computation of Round 1 , then $p$-Assign is also correct on the output of it. Given this observation (and by designing the details of $\varphi^{\mu: v}$ appropriately), we can complete this step by just reusing the second part of the analysis of KRR no-signaling PCP, where it is shown that if Assign is correct on the input, then Assign is also correct on the output.

We next show that in every $\varphi^{\mu: \nu}$, if p-Assign is correct on $P^{\mu}$ and $P^{\nu}$ 's end states and outgoing messages in Round 1 , then in every $\varphi^{\mu: \nu}$, p -Assign is also correct on $P^{\mu}$ and $P^{\nu}$, start states and incoming messages in Round 2. In this step, we consider three cases for each $\varphi^{\mu: \nu}$.

Case 1. We first consider the correctness on $P^{\xi}$ 's start state of Round $2(\xi \in\{\mu, v\})$. This case is easy and we just need to use the everywhere local consistency of p-Assign and the definition of $\varphi^{\mu: \nu}$. Specifically, since $\varphi^{\mu: \nu}$ has clauses that check that $P^{\xi}$ 's start state of Round 2 is equal to its end state of Round 1, the everywhere local consistency of p -Assign guarantees that p -Assign assigns the same value on $P^{\xi}$ 's start state of Round 2 and on $P^{\xi}$ 's end state of Round 1 , and thus, if p-Assign is correct on the latter, it is also correct on the former.

Case 2. We next consider the correctness on $P^{\mu}$ 's incoming message from $P^{\nu}$ and $P^{\nu}$ 's incoming message from $P^{\mu}$ at the beginning of Round 2. Again, this case is easy and we just need to use the everywhere local consistency of p-Assign and the definition of $\varphi^{\mu: v}$ (which has clauses that check that the message that $P^{\mu}$ receives from $P^{v}$ at the beginning of Round 2 is equal to the one that $P^{v}$ sends to $P^{\mu}$ at the end of Round 1 , and vise versa).

[^8]Case 3. We finally consider the correctness on $P^{\mu}$ and $P^{\nu}$ 's incoming messages from the parties other than $P^{\mu}$ and $P^{v}$ at the beginning of Round 2. This case is not straightforward, and we rely on that p-Assign satisfies the consistency on the views, which is guaranteed since $p$-Assign is obtained from a well-behaving prover. Let us consider, for example, $P^{\mu}$ 's incoming message from $P^{\xi}(\xi \notin\{\mu, v\})$. Then, since the consistency on the views guarantees that p-Assign assigns the same value in $\varphi^{\mu: v}$ and $\varphi^{\mu: \xi}$ as $P^{\mu}$ 's incoming message from $P^{\xi}$, if $p$-Assign is correct on it in $\varphi^{\mu: \xi}$, then p-Assign is also correct on it in $\varphi^{\mu: \nu}$. Then, since we showed in Case 2 that p-Assign is indeed correct on it in $\varphi^{\mu: \xi}$, we conclude that p -Assign is correct on it in $\varphi^{\mu i v} .{ }^{14}$

By proceeding identically (and observing that, by definition, p -Assign is correct on $P^{\mu}$ and $P^{\nu}$, start states and incoming messages in Round 1 in every $\varphi^{\mu: v}$ ), we conclude that $p$-Assign is correct on any variable, and in particular correct on $P^{\mu}$ and $P^{v}$ s final states in every $\varphi^{\mu: \nu}$.

Step 3: Obtaining soundness. On the one hand, the value that $p$-Assign assigns as the output of any party $P^{\mu}$ is always 1 due to the everywhere local consistency of p-Assign (recall that $\varphi^{\mu i v}$ has a clause that checks that $P^{\mu}$ 's output is 1 ). On the other hand, since p-Assign is correct on the output of $P^{\mu}$, it is also equal to the value that $P^{\mu}$ outputs in the correct views. Thus, $P^{\mu}$ outputs 1 in the correct view, which means that the statement proven by the prover is true on the XOR of the MPC inputs of the correct views. From the definition of the extractor, it follows that the prover cannot prove false statements on the extracted value.

### 4.3.4 More details of well-behaving provers.

It remains to give an overview of the concrete definition of well-behaving provers. As we mentioned earlier, we define well-behaving provers so that when we convert a CNS well-behaving prover into a parallel local assignment generator p-Assign, then p-Assign has the aforementioned two consistency properties.

Before giving the definition of well-behaving provers, we give a few details about the construction of KRR no-signaling PCP.

- When a PCP proof $\pi$ for a 3SAT instance $\varphi$ is created by using a satisfying assignment $\boldsymbol{x}$ to $\varphi$, the PCP proof $\pi$ contains an encoding of $\boldsymbol{x},{ }^{15}$ i.e., there is a set of queries $D(X)$ such that $\left.\pi\right|_{D(X)}$ is an encoding of $\boldsymbol{x}$.
- Furthermore, we can make sure that in our protocol, each PCP proof $\pi^{\mu: v}$ for $\varphi^{\mu: v}$ (where $\pi^{\mu: v}$ is created by using $\left(\right.$ view $^{\mu}$, view $\left.\left.^{\nu}\right)\right)$ contains encodings of $x_{1, \mathrm{in}}^{\mu}, x_{1, \mathrm{in}}^{\nu}$, view ${ }^{\mu}$, and view ${ }^{\nu}$, i.e., there are sets of queries $D\left(X_{1, \mathrm{in}}^{\mu}\right), D\left(X_{1, \mathrm{in}}^{\nu}\right)$, $D\left(X^{\mu}\right), D\left(X^{\nu}\right)$ such that:
$-\left.\pi^{\mu: v}\right|_{D\left(X_{1, \mathrm{i})}^{\mu}\right)}$ and $\left.\pi^{\mu: v}\right|_{D\left(X_{1, \mathrm{i})}^{v}\right)}$ are encodings of $x_{1, \mathrm{in}}^{\mu}$ and $x_{1, \mathrm{in}}^{\nu}$, respectively.
- $\left.\pi^{\mu: \nu}\right|_{D\left(X^{\mu}\right)}$ and $\left.\pi^{\mu: \nu}\right|_{D\left(X^{\nu}\right)}$ are encodings of view ${ }^{\mu}$ and view ${ }^{\nu}$, respectively.
(Recall that $x_{1, \mathrm{in}}^{\mu}:=\mathrm{st}_{0}^{\mu} \| \mathrm{i}-\mathrm{msgs}_{1}^{\mu}$ and $x_{1, \mathrm{in}}^{\nu}:=\mathrm{st}_{0}^{\nu} \| \mathrm{i}-\mathrm{msgs}_{1}^{\nu}$ are the initial states and dummy incoming messages that are computed in the commit phase.)

Then, informally speaking, a CNS prover is said to be well-behaving if it satisfies the following two consistency properties.

Consistency on $D\left(X_{1, \text { in }}^{\mu}\right)$. Once the commit phase is completed, the prover gives the same response to a query in $D\left(X_{1, \text { in }}^{\mu}\right)$ $(\mu \in[M])$ in different invocations. More concretely, for any queries $\left\{Q_{0}^{\mu: v}\right\}_{\mu, v \in[M]},\left\{Q_{1}^{\mu: v}\right\}_{\mu, v \in[M]}$, any $\alpha, \beta, \gamma, \delta \in[M]$ such that $\exists \xi \in\{\alpha, \beta\} \cap\{\gamma, \delta\}$, and any $q \in Q_{0}^{\alpha ; \beta} \cap Q_{1}^{\gamma: \delta} \cap D\left(X_{1, \text { in }}^{\xi}\right)$, we have $\pi_{0}^{* \alpha ; \beta}(q)=\pi_{1}^{* \gamma: \delta}(q)$, where $\pi_{0}^{* \alpha ; \beta}$ and $\pi_{1}^{* \gamma: \delta}$ are generated as follows.

1. $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\right.$ C.Com $_{1}^{*}$, R.Com $\left.{ }_{1}\right\rangle$
2. $\left(f_{0},\left\{\pi_{0}^{* \mu: \nu}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C} \cdot \operatorname{Prv}_{1}^{*}\left(\operatorname{st}_{C},\left\{Q_{0}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$
3. $\left(f_{1},\left\{\pi_{1}^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C} \cdot \operatorname{Prv}_{1}^{*}\left(\operatorname{st}_{C},\left\{Q_{1}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$

Consistency on $D\left(X^{\mu}\right)$. The prover gives the same responses to a query in $D\left(X^{\mu}\right)(\mu \in[M])$ in a single invocation. More concretely, for any queries $\left\{Q^{\mu: \nu}\right\}_{\mu, v \in[M]}$, any $\alpha, \beta, \gamma, \delta \in[M]$ such that $\exists \xi \in\{\alpha, \beta\} \cap\{\gamma, \delta\}$, and any $q \in Q^{\alpha: \beta} \cap Q^{\gamma: \delta} \cap$ $D\left(X^{\xi}\right)$, we have $\pi^{* \alpha ; \beta}(q)=\pi^{* \gamma: \delta}(q)$, where $\pi^{*}$ is generated as follows.

1. $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\right.$ C.Com $_{1}^{*}$, R.Com $\left.{ }_{1}\right\rangle$
2. $\left(f,\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C} \cdot \operatorname{Prv}_{1}^{*}\left(\operatorname{st}_{C},\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right)$
[^9]To show that the above definition indeed implies the aforementioned two consistency properties of p-Assign, we need to see the details of $p$-Assign. Specifically, we rely on that p-Assign obtains local assignments by applying a procedure called self-correction on the cheating prover. In this overview, we do not give the details of self-correction, and we just note that p -Assign obtains local assignments in the following manner: p -Assign first creates some queries $Q^{\mu: v}$ for each $\mu, v \in[M]$ based on its input, next queries $\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}$ to the prover, and finally obtains the local assignments based on the prover's responses.

Now, at first sight, it seems trivial to show that the above definition of well-behaving provers implies the two consistency properties of $p$-Assign. Consider, for example, showing that the above definition implies that $p$-Assign has the consistency on the initial states. Then, since $p$-Assign obtains local assignments based on the prover's responses, and well-behaving provers are guaranteed to give unique responses to any queries on the initial states (i.e., any queries in $D\left(X_{1, \text { in }}^{\mu}\right)(\mu \in[M])$ ), it seems trivial to show that p -Assign makes unique assignments on the initial states.

However, this intuition is wrong. For example, in the case of showing the consistency on the initial states, the problem is that even when making assignments on the initial states, p -Assign's queries to the prover includes those that are not in $D\left(X_{1, \text { in }}^{\mu}\right)(\mu \in[M])$, and well-behaving provers' responses to such queries are not necessarily unique.

Fortunately, this problem can be solved relatively easily by using a technique in a previous work [HR18]. Specifically, by letting the verifier of KRR no-signaling PCP do several additional tests on the prover, we can show that it suffices to consider a modified version of p -Assign, which obtains local assignments on the initial states (resp., the views) based solely on the prover's responses to the queries in $D\left(X_{1, \text { in }}^{\mu}\right)$ (resp., in $D\left(X^{\mu}\right)$ ). ${ }^{16}$ On this modified version of p -Assign, it is indeed easy to show that the two consistency properties of well-behaving provers imply the two consistency properties of p -Assign by relying on analyses given in [KRR14].

### 4.3.5 Towards formal proof.

Finally, we discuss what modifications are needed to turn the above proof idea into a formal proof.
First, we need to modify the extractor so that it can open the commitment (instead of just extracting a committed value) as required in Definition 3; along the way, we also need to define the open phase of the protocol appropriately. Recall that in the above, the extractor uses the parallel local assignment generator p -Assign to extract a committed value. Motivated by this construction of the extractor, we follow the following overall approach: we define the open phase so that running p-Assign jointly with the receiver is sufficient for the committer to succeed in the open phase. To implement this approach, we rely on that, as mentioned above, p -Assign obtains local assignments in the following manner: p -Assign first creates some queries $Q^{\mu: v}$ for each $\mu, v \in[M]$ based on its input, next queries $\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}$ to the prover, and finally obtains the local assignments based on the prover's responses. Given this structure of p -Assign, we define the open phase as follows.

1. In the first round, the receiver computes queries as in p-Assign and sends them to the committer.
2. In the second round, the committer gives responses to the queries.
3. Finally, the receiver computes the local assignments from the responses as in p-Assign and then uses them to extract a committed value as in the extractor.

Then, we modify the extractor so that it simply forwards the queries from the receiver to the cheating prover and next forwards the responses from the cheating prover to the receiver. Since the extracted value is computed from the output of p -Assign just as before (the only difference is that now p -Assign is executed jointly between the extractor and the receiver), we can still prove that any CNS well-behaving cheating prover cannot prove false statements on the extracted value. Furthermore, we can show that the above open phase is strong enough to guarantee a meaningful binding property. Specifically, by letting the receiver make additional queries in the open phase, ${ }^{17}$ we can prove the binding property against CNS well-behaving decommitters, which are defined similarly to CNS well-behaving provers. (The proof of the binding property proceeds essentially in the same way as we show that p-Assign satisfies the consistency on the initial states in the proof of the soundness against well-behaving provers, where we show that once the commit phase is completed, the assignments by p-Assign on the MPC initial states-which define the committed value-are unique.)

Second, we need to consider the case that p-Assign can err (i.e., the everywhere local consistency does not necessarily hold with probability 1). Fortunately, this case is already handled in Kalai et al. [KRR14], and we can handle it identically. (Concretely, when showing that p-Assign is correct on every variable in the round-by-round way, we only show that p -Assign is correct on average, i.e., instead of showing that p -Assign is correct on any variables that correspond to, say, the start state and incoming message of a round, we only show that p -Assign is correct on randomly chosen $\omega(\log \lambda)$ such variables. It is shown in [KRR14] that showing such average-case correctness is sufficient to prove the soundness.)

Third, we need to consider adaptive soundness, where the cheating prover chooses the statement to prove at the last round of the prove phase. Fortunately, adaptive soundness is already considered in previous works (e.g., [BHK17]), and we can handle it identically.

[^10]
## 5 Overview of Step 2 (Non-WI Scheme with Soundness against CNS Provers)

We give an overview of our non-WI commit-and-prove protocol $\left\langle C_{2}, R_{2}\right\rangle$, which is ( $1-$ negl)-sound against CNS provers.
Our high-level approach is to upgrade the protocol $\left\langle C_{1}, R_{1}\right\rangle$ that we give in Step 1 so that the soundness holds against any (not necessarily well-behaving) CNS provers. Recall that, roughly speaking, an adversary is well-behaving if for every $\mu \in[M]$,

1. it does not give different responses to a query in $D\left(X_{1, \mathrm{in}}^{\mu}\right)$ in different invocations, and
2. it does not give different responses to a query in $D\left(X^{\mu}\right)$ in a single invocation,
where $D\left(X_{1, \text { in }}^{\mu}\right)$ and $D\left(X^{\mu}\right)$ are sets of queries such that in $\left\langle C_{1}, R_{1}\right\rangle$, the prover is supposed to create PCPs $\left\{\pi^{\mu: v}\right\}_{\mu, v \in[M]}$ such that $\left.\pi^{\mu: \nu}\right|_{D\left(X_{1 \text { in }}^{\mu}\right)}$ is an encoding of $x_{1, \text { in }}^{\mu}$ and $\left.\pi^{\mu: v}\right|_{D\left(X^{\mu}\right)}$ is an encoding of view ${ }^{\mu}$ for every $v \in[M]$, where $x_{1, \text { in }}^{\mu}$ is the value that is fixed in the commit phase and view ${ }^{\mu}$ is the view that is fixed in the prove phase. Naturally, we enforce this behavior on the prover by relying on collision-resistant hash functions: we require the prover to publish the roots of the tree-hash of the encodings of $\left\{x_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}$ and $\left\{\text { view }^{\mu}\right\}_{\mu \in[M]}$, and also require it to give responses along with appropriate certificates when it is queried on these values.

More concretely, we consider the following protocol (which is slightly oversimplified from the actual protocol in Appendix F). In the following, for a hash function hf, we denote by TreeHash ${ }_{h f}$ an algorithm that computes the Merkle tree-hash of the input.

## Commit Phase

Round 1: R.Com 2 sends a hash function hf $\in \mathcal{H}$ to $\mathrm{C}^{\mathrm{C}} \mathrm{Com}_{2}$.
Round 2: Given ( $x_{\text {сом }}$, hf) as input, C.Com ${ }_{2}$ obtains $\left\{x_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$ by running C.Com ${ }_{1}\left(x_{\text {Сом }}\right)$, computes encodings $\left\{X_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}$ of them, and then outputs $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}:=\operatorname{TreeHash}_{\mathrm{hf}}\left(X_{1, \mathrm{in}}^{\mu}\right)\right\}_{\mu \in[M]}$ as the commitment and store (hf, $\left\{X_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$ ) as the internal state.

## Prove Phase

Round 1: R.Prv. $Q_{2}$ works identically with R.Prv. $Q_{1}$. That is, R.Prv. $Q_{2}$ obtains $\left\{Q^{\mu, v}\right\}_{\mu, v \in[M]}$ just like R.Prv. Q ${ }_{1}$ does, and outputs $\left\{Q^{\mu, \nu}\right\}_{\mu, v \in[M]}$ as the query.
Round 2: Given the statement $f$ and the query $\left\{Q^{\mu, v^{\nu}}\right\}_{\mu, v \in[M]}$ as input, C. $\operatorname{Prv}_{2}$ does the following.

1. Obtain $\left\{\text { view }^{\mu}\right\}_{\mu \in[M]}$ and $\left\{\pi^{\mu: v}\right\}_{\mu, v \in[M]}$ just like C.Prv ${ }_{1}$ does.
2. Compute encodings $\left\{X^{\mu}\right\}_{\mu \in[M]}$ of $\left\{\text { view }{ }^{\mu}\right\}_{\mu \in[M]}$, and compute $\left\{\mathrm{r}^{\mu}:=\operatorname{TreeHash}_{\mathrm{hf}}\left(X^{\mu}\right)\right\}_{\mu \in[M]}$.
3. Augment each $\pi^{\mu: v}$ as follows.

- Augment each symbol in $\left.\pi^{\mu: v}\right|_{D\left(X_{1, \mathrm{in}}^{\xi}\right)}(\xi \in\{\mu, v\})$ with a certificate for opening $\mathrm{rt}_{1, \mathrm{in}}^{\xi}$ to it.
- Augment each symbol in $\left.\pi^{\mu: \nu}\right|_{D\left(X^{\xi}\right) \backslash D\left(X_{1, \mathrm{i}}^{\xi}\right)}(\xi \in\{\mu, v\})$ with a certificate for opening $\mathrm{rt}^{\xi}$ to it.

4. Output $\left(\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\left.\pi^{\mu: \nu}\right|_{Q^{\mu: v}}\right\}_{\mu, v \in[M]}\right)$ as the proof.

Verification: Given the commitment $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$, the statement $f$, and the proof $\left(\left\{\mathrm{r}^{\mu}\right\}_{\mu \in[M],},\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right)$ as input, R.Prv. $D_{2}$ works identically with R.Prv. $D_{1}$ except that before the verification, each $\pi^{* \mu: v}$ is "filtered" as follows.

- Replace each symbol ( $x$, cert) in $\left.\pi^{* \mu: v}\right|_{D\left(X_{1, \mathrm{in}}^{\xi}\right)}\left(\xi \in\{\mu, \nu\}\right.$ ) with $x$ if cert is a valid certificate for opening $\mathrm{rt} \mathrm{t}_{1, \text { in }}^{\xi}$ to $x$, and replace it with $\perp$ otherwise.
- Replace each symbol ( $x$, cert) in $\left.\pi^{* \mu: \nu}\right|_{D\left(X^{\xi}\right) \backslash D\left(X_{1, \mathrm{in}}^{\xi}\right)}(\xi \in\{\mu, v\})$ with $x$ if cert is a valid certificate for opening $\mathrm{rt}^{\xi}$ to $x$, and replace it with $\perp$ otherwise.

We prove the soundness of $\left\langle C_{2}, R_{2}\right\rangle$ by relying on the soundness of $\left\langle C_{1}, R_{1}\right\rangle$. Specifically, for any cheating committer-prover $C_{2}^{*}=\left(\mathrm{C}^{\mathrm{Com}}{ }_{2}^{*}, \mathrm{C} . \operatorname{Pr} v_{2}^{*}\right)$ against $\left\langle C_{2}, R_{2}\right\rangle$, we consider the following cheating committer-prover $C_{1}^{*}=$ (C.Com ${ }_{1}^{*}$, C.Prv ${ }_{1}^{*}$ ) against $\left\langle C_{1}, R_{1}\right\rangle$.

- Committer. C.Com ${ }_{1}^{*}$ runs $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{\mathrm{C}} \mathrm{Com}_{2}^{*}\right.$, R.Com $\left.{ }_{2}\right\rangle$ internally, sends an empty string to R.Com ${ }_{1}$ as the commitment, and stores ( $\mathrm{com}, \mathrm{st}_{C}$ ) as the internal state.
- Prover. Given (com, st $C_{C}$ ) and $\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}$ as input, C.Prv ${ }_{1}^{*}$ first runs $\left(f,\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow$ C.Prv${ }_{2}^{*}\left(\operatorname{st}_{C},\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right)$. Then, C.Prv ${ }_{1}^{*}$ filters each $\pi^{* \mu: v}$ as in the verification of $\left\langle C_{2}, R_{2}\right\rangle$, and sends $\left(f,\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right)$ to R.Prv ${ }_{1}$ as the proof.

It is straightforward to show that (1) $C_{1}^{*}$ is successful if $C_{2}^{*}$ is successful and (2) $C_{1}^{*}$ is well-behaving CNS. (The latter follows from the the binding property of TreeHash ${ }_{\text {hf }}$.)

## 6 Overview of Subsequent Steps of Proof of Theorem 1

In Step 3, we upgrade the soundness to the one with negligible soundness error. Fortunately, this type of soundness amplification is already studied by Kalai et al. [KRR14] as mentioned in Section 3, and it suffices to apply their soundness amplification on the protocol $\left\langle C_{2}, R_{2}\right\rangle$ that we obtained in Step 2. Concretely, in this step, we just borrow a soundness amplification technique from [KRR14, BHK17], which amplifies soundness by letting the verifier use a smaller threshold parameter for the PCP decision algorithm (i.e., letting the verifier tolerate a smaller number of failures on the tests that it applies on the prover).

In Step 4, we upgrade the soundness to the one against any (not necessarily CNS) adversaries. Again, this type of soundness amplification is already studied by Kalai et al. [KRR14] as mentioned in Section 3, and it suffices to apply their soundness amplification on the protocol $\left\langle C_{3}, R_{3}\right\rangle$ that we obtained in Step 3. Concretely, in this step, we just borrow a transformation from [KRR14], which enforces CNS behavior on the committer by encrypting the verifier queries by PIR. (Intuitively, encrypting the verifier queries by PIR is helpful to enforce CNS behavior since it forces the prover to answer each query independently of the other queries.)

In Step 5, we add the WI property while tolerating that the soundness error increases to a constant. Toward this end, we augment the protocol $\left\langle C_{4}, R_{4}\right\rangle$ that we obtained in Step 4 with commitment schemes and OT by using these two primitives as in the non-succinct protocol that we sketched in Section 1.2. The soundness and WI of the resultant protocol $\left\langle C_{5}, R_{5}\right\rangle$ can be shown similarly to those of the non-succinct protocol in Section 1.2. That is, the soundness follows from the security of OT and the soundness of $\left\langle C_{4}, R_{4}\right\rangle,{ }^{18}$ and the WI property follows from the 2-privacy of $\left\langle C_{4}, R_{4}\right\rangle$, which roughly guarantees that the verifier does not learn any secret information if it only obtains one of the $M^{2}$ KRR no-signaling PCP strings. (The 2-privacy of $\left\langle C_{4}, R_{4}\right\rangle$, in turn, follows from the 2-privacy of the underlying MPC protocol $\Pi$.)

## References

[AL17] Gilad Asharov and Yehuda Lindell. A full proof of the BGW protocol for perfectly secure multiparty computation. Journal of Cryptology, 30(1):58-151, January 2017.
$\left[\mathrm{BBK}^{+} 16\right]$ Nir Bitansky, Zvika Brakerski, Yael Tauman Kalai, Omer Paneth, and Vinod Vaikuntanathan. 3-message zero knowledge against human ignorance. In Martin Hirt and Adam D. Smith, editors, TCC 2016-B, Part I, volume 9985 of LNCS, pages 57-83. Springer, Heidelberg, October / November 2016.
[BGW88] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic faulttolerant distributed computation (extended abstract). In 20th ACM STOC, pages 1-10. ACM Press, May 1988.
[BHK16] Zvika Brakerski, Justin Holmgren, and Yael Kalai. Non-interactive RAM and batch NP delegation from any PIR. Cryptology ePrint Archive, Report 2016/459, 2016. http://eprint.iacr.org/2016/459.
[BHK17] Zvika Brakerski, Justin Holmgren, and Yael Tauman Kalai. Non-interactive delegation and batch NP verification from standard computational assumptions. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, 49th ACM STOC, pages 474-482. ACM Press, June 2017.
[BK18] Zvika Brakerski and Yael Tauman Kalai. Monotone batch NP-delegation with applications to access control. Cryptology ePrint Archive, Report 2018/375, 2018. https://eprint.iacr.org/2018/375.
[BK20] Zvika Brakerski and Yael Kalai. Witness indistinguishability for any single-round argument with applications to access control. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, PKC 2020, Part II, volume 12111 of $L N C S$, pages 97-123. Springer, Heidelberg, May 2020.
[ $\left.\mathrm{BKK}^{+} 18\right]$ Saikrishna Badrinarayanan, Yael Tauman Kalai, Dakshita Khurana, Amit Sahai, and Daniel Wichs. Succinct delegation for low-space non-deterministic computation. In Ilias Diakonikolas, David Kempe, and Monika Henzinger, editors, 50th ACM STOC, pages 709-721. ACM Press, June 2018.
[CLOS02] Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multiparty secure computation. In 34th ACM STOC, pages 494-503. ACM Press, May 2002.
[GGP10] Rosario Gennaro, Craig Gentry, and Bryan Parno. Non-interactive verifiable computing: Outsourcing computation to untrusted workers. In Tal Rabin, editor, CRYPTO 2010, volume 6223 of LNCS, pages 465-482. Springer, Heidelberg, August 2010.
[GK96] Oded Goldreich and Hugo Krawczyk. On the composition of zero-knowledge proof systems. SIAM Journal on Computing, 25(1):169-192, 1996.

[^11][GKR15] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: Interactive proofs for muggles. Journal of the ACM, 62(4):27:1-27:64, 2015.
[GLOV12] Vipul Goyal, Chen-Kuei Lee, Rafail Ostrovsky, and Ivan Visconti. Constructing non-malleable commitments: A black-box approach. In 53rd FOCS, pages 51-60. IEEE Computer Society Press, October 2012.
[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. SIAM Journal on Computing, 18(1):186-208, 1989.
[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In Alfred Aho, editor, 19th ACM STOC, pages 218-229. ACM Press, May 1987.
[GOSV14] Vipul Goyal, Rafail Ostrovsky, Alessandra Scafuro, and Ivan Visconti. Black-box non-black-box zero knowledge. In David B. Shmoys, editor, 46th ACM STOC, pages 515-524. ACM Press, May / June 2014.
[HR18] Justin Holmgren and Ron Rothblum. Delegating computations with (almost) minimal time and space overhead. In Mikkel Thorup, editor, 59th FOCS, pages 124-135. IEEE Computer Society Press, October 2018.
[HV16] Carmit Hazay and Muthuramakrishnan Venkitasubramaniam. On the power of secure two-party computation. In Matthew Robshaw and Jonathan Katz, editors, CRYPTO 2016, Part II, volume 9815 of LNCS, pages 397429. Springer, Heidelberg, August 2016.
[HV18] Carmit Hazay and Muthuramakrishnan Venkitasubramaniam. Round-optimal fully black-box zero-knowledge arguments from one-way permutations. In Amos Beimel and Stefan Dziembowski, editors, TCC 2018, Part I, volume 11239 of LNCS, pages 263-285. Springer, Heidelberg, November 2018.
[IKOS09] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Zero-knowledge proofs from secure multiparty computation. SIAM Journal on Computing, 39(3):1121-1152, 2009.
[IW14] Yuval Ishai and Mor Weiss. Probabilistically checkable proofs of proximity with zero-knowledge. In Yehuda Lindell, editor, TCC 2014, volume 8349 of LNCS, pages 121-145. Springer, Heidelberg, February 2014.
[Kil92] Joe Kilian. A note on efficient zero-knowledge proofs and arguments (extended abstract). In 24th ACM STOC, pages 723-732. ACM Press, May 1992.
[Kiy18] Susumu Kiyoshima. No-signaling linear PCPs. Cryptology ePrint Archive, Report 2018/649, 2018. https: //eprint.iacr.org/2018/649.
[KOS18] Dakshita Khurana, Rafail Ostrovsky, and Akshayaram Srinivasan. Round optimal black-box "commit-andprove". In Amos Beimel and Stefan Dziembowski, editors, TCC 2018, Part I, volume 11239 of LNCS, pages 286-313. Springer, Heidelberg, November 2018.
[KP16] Yael Tauman Kalai and Omer Paneth. Delegating RAM computations. In Martin Hirt and Adam D. Smith, editors, TCC 2016-B, Part II, volume 9986 of LNCS, pages 91-118. Springer, Heidelberg, October / November 2016.
[KRR13] Yael Tauman Kalai, Ran Raz, and Ron D. Rothblum. How to delegate computations: The power of no-signaling proofs. Cryptology ePrint Archive, Report 2013/862, 2013. http://eprint.iacr.org/2013/862.
[KRR14] Yael Tauman Kalai, Ran Raz, and Ron D. Rothblum. How to delegate computations: the power of no-signaling proofs. In David B. Shmoys, editor, 46th ACM STOC, pages 485-494. ACM Press, May / June 2014.
[LP12] Huijia Lin and Rafael Pass. Black-box constructions of composable protocols without set-up. In Reihaneh Safavi-Naini and Ran Canetti, editors, CRYPTO 2012, volume 7417 of $L N C S$, pages 461-478. Springer, Heidelberg, August 2012.
[MPP20] Andrew Morgan, Rafael Pass, and Antigoni Polychroniadou. Succinct non-interactive secure computation. In Anne Canteaut and Yuval Ishai, editors, EUROCRYPT 2020, Part II, volume 12106 of LNCS, pages 216-245. Springer, Heidelberg, May 2020.
[ORS15] Rafail Ostrovsky, Silas Richelson, and Alessandra Scafuro. Round-optimal black-box two-party computation. In Rosario Gennaro and Matthew J. B. Robshaw, editors, CRYPTO 2015, Part II, volume 9216 of LNCS, pages 339-358. Springer, Heidelberg, August 2015.
[PR14] Omer Paneth and Guy N. Rothblum. Publicly verifiable non-interactive arguments for delegating computation. Cryptology ePrint Archive, Report 2014/981, 2014. http://eprint.iacr.org/2014/981.
[PR17] Omer Paneth and Guy N. Rothblum. On zero-testable homomorphic encryption and publicly verifiable noninteractive arguments. Cryptology ePrint Archive, Report 2017/903, 2017. http://eprint.iacr.org/ 2017/903.
[PW09] Rafael Pass and Hoeteck Wee. Black-box constructions of two-party protocols from one-way functions. In Omer Reingold, editor, TCC 2009, volume 5444 of LNCS, pages 403-418. Springer, Heidelberg, March 2009.
[Sud00] Madhu Sudan. Probabilistically checkable proofs, 2000. http://madhu.seas.harvard.edu/MIT/pcp/ pcp.ps.

## A On Verification Time

The verification of our protocol is not succinct since we use a simpler version of KRR succinct argument where the verifier naively evaluates a low-degree extension (LDE) of the indicator function of a 3CNF formula whose size is polynomially related to the complexity of the statement. In Kalai et al. [KRR14] and subsequent works [BHK17, HR18], the verification is made succinct by observing that when the statement to be proven satisfies some conditions, the evaluation of the LDE can be either recursively delegated to the prover succinctly or locally performed by the verifier efficiently. In our protocol, KRR succinct argument is used for proving statements that are related to the next-message function of the underlying perfect 2-privacy MPC protocol (cf. Section 1.2, Section 4, and Appendix D). Thus, if we can show that the above-mentioned conditions are satisfied for a specific perfect 2-privacy MPC protocol, the verification of our protocol can be made succinct.

## B On Definition of Commit-and-prove Protocols

## B. 1 Differences from Definition in Khurana et al. [KOS18]

Our definition of commit-and-prove protocols in Section 2.2 has several differences from the definition in Khurana et al. [KOS18]. First, our definition has several syntactical differences.

- Instead of thinking the prove phase as a part of the commit phase, we separate the prove phase from the commit phase.
- We focus on the case that each of the prove phase and the open phase consists of two rounds.

Next, our definition is stronger than the definition of Khurana et al. [KOS18] in the following points.

- We explicitly define the soundness and witness indistinguishability in the delayed-input setting, where the statement to be proven is chosen at the last round of the prove phase.
- In the definition of the soundness, we require the extractor to decommit the commitment to a value on which any committer cannot prove false statements. (In the definition in [KOS18], the extractor just outputs such a value without decommitment, with the guarantee that any committer cannot decommit the commitment to a value other than the extracted one.)
We think that requiring the extractor to decommit the commitment is important, as otherwise the definition would not prevent an attack where the committer gives an accepting proof on an invalid commitment (i.e., a commitment that cannot be opened to any value). ${ }^{19}$ (This is because even if such an attack is possible, we can still show that any committer cannot decommit the commitment to a value other than the extracted one, since an invalid commitment cannot be opened to any value.) We remark that such an attack is possible if a commit-and-prove protocol is naively executed in parallel multiple times.

Finally, our definition is weaker than the definition of Khurana et al. [KOS18] in the following points.

- In the definitions of the binding and the soundness, the extractor succeeds only on an overwhelming fraction of the executions of the commit phase, rather than on any execution of the commit phase.
- In the definition of the soundness, the extractor is allowed to depend on the success probability of the cheating committer.

[^12]
## B. 2 Rationale behind Our Definition

Since our definition of commit-and-prove protocols in Section 2.2 might look too cumbersome, we explain the rationale behind it.

First, binding and witness-indistinguishability are defined naturally, and the only complication is that we allow the open phase to be interactive in the definition of binding. To guarantee a stronger notion of binding, our definition considers an adversary that obtains two sets of receiver decommitment queries simultaneously (rather than obtain each of them separately).

Next, soundness is defined similarly to proof-of-knowledge of interactive proofs [GMR89]. A complication is that we define it so that it guarantees the adaptive delayed-input property, i.e., it holds against an adversary that chooses the statement to prove at the last round of the prove phase. ${ }^{20}$ To guarantee proof-of-knowledge with the adaptive delayed-input property, our definition requires that if a cheating prover convinces the verifier for a commitment com with sufficiently high probability, then there exists a value $x^{*}$ such that (1) the extractor can decommit com to $x^{*}$ and (2) the cheating prover cannot prove false statements about $x^{*}$. (We remark that the extractor is required to succeed in the decommitment of each bit of $x^{*}$ with overwhelming probability so that we can obtain the whole $x^{*}$ by repeatedly using the extractor.)

## C Additional Preliminaries

## C. 1 Oblivious Transfer Protocols

We recall the definition of 1-out-of- $n$ oblivious transfer (OT) protocols for a constant $n$, based on the description by Brakerski and Kalai [BK18] while straightforwardly generalizing the description from 1-out-of-2 OT to 1-out-of- $n$ OT.

We focus on two-round OT protocols. They consist of three ppt algorithms, $\left(\mathrm{OT}_{1}, \mathrm{OT}_{2}, \mathrm{OT}_{3}\right)$, such that $\mathrm{OT}_{1}$ is used by the receiver to send the first-round message, $\mathrm{OT}_{2}$ is used by the sender to send the second-round message, and $\mathrm{OT}_{3}$ is used by the receiver to compute the output from the sender message.

Definition 11 (Correctness). We say that a 1-out-of-n OT protocol $\mathrm{OT}=\left(\mathrm{OT}_{1}, \mathrm{OT}_{2}, \mathrm{OT}_{3}\right)$ satisfies correctness if for all $\lambda \in \mathbb{N}, i \in[n]$, and $\left(m_{1}, \ldots, m_{n}\right) \in\left(\{0,1\}^{\lambda}\right)^{n}$,

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\mathrm{OT}_{3}\left(\mathrm{st}_{o r}, \mathrm{ot}_{3}\right)=m_{i} & \begin{array}{l}
\left(\mathrm{ot}_{1}, \mathrm{st}_{o r}\right) \leftarrow \mathrm{OT}_{1}\left(1^{\lambda}, i\right) \\
\mathrm{ot}_{2} \leftarrow \mathrm{OT}_{2}\left(1^{\lambda}, \mathrm{ot}_{1},\left(m_{1}, \ldots, m_{n}\right)\right)
\end{array}
\end{array}\right]=1 .
$$

Definition 12 (Receiver privacy ${ }^{21}$ ). We say that a 1-out-of-n OT protocol $\mathrm{OT}=\left(\mathrm{OT}_{1}, \mathrm{OT}_{2}, \mathrm{OT}_{3}\right)$ satisfies receiver privacy if for any PPT distinguisher $\mathcal{D}$ and any $\lambda \in \mathbb{N}$,

$$
\operatorname{Pr}\left[\mathcal{D}\left(\mathrm{ot}_{1}, i_{b}\right)=b \left\lvert\, \begin{array}{l}
\text { Sample random } b \in\{0,1\} \text { and } i_{0}, i_{1} \in[n] \\
\left(\mathrm{ot}_{1}, \mathrm{st}_{o T}\right) \leftarrow \mathrm{OT}_{1}\left(1^{\lambda}, i_{0}\right)
\end{array}\right.\right] \leq \frac{1}{2}+\operatorname{neg}(\lambda)
$$

Definition 13 (Sender privacy). We say that a 1-out-of-n OT protocol $\mathrm{OT}=\left(\mathrm{OT}_{1}, \mathrm{OT}_{2}, \mathrm{OT}_{3}\right)$ satisfies sender privacy if for all $\lambda \in \mathbb{N}$ and for all $\mathrm{ot}_{1}^{*} \in\{0,1\}^{\mathrm{lot}_{1} \mathrm{l}}$, there exists $i^{*} \in[n]$ such that for any $\left(m_{1}, \ldots, m_{n}\right) \in\left(\{0,1\}^{\lambda}\right)^{n}$ and $\left(m_{1}^{\prime}, \ldots, m_{n}^{\prime}\right) \in\left(\{0,1\}^{\lambda}\right)^{n}$ such that $m_{i^{*}}=m_{i^{*}}^{\prime}$, the distributions $\mathrm{OT}_{2}\left(1^{\lambda}, \mathrm{ot}_{1}^{*},\left(m_{1}, \ldots, m_{n}\right)\right)$ and $\mathrm{OT}_{2}\left(1^{\lambda}, \mathrm{ot}_{1}^{*},\left(m_{1}^{\prime}, \ldots, m_{n}^{\prime}\right)\right)$ are computationally indistinguishable. We denote by Extot an arbitrary computationally unbounded procedure that produces $i^{*}$ as above from $\mathrm{ot}_{1}^{*}$, i.e., $i^{*}:=\operatorname{Ext}_{\mathrm{ot}}\left(1^{\lambda}, \mathrm{ot}_{1}^{*}\right)$

## C. 2 Private Information Retrieval

We recall the definition of private information retrieval (PIR) schemes, based on the description by Brakerski et al. [BHK17].

Definition 14. A 2-message private information retrieval (PIR) scheme over alphabet $\left\{\Sigma_{\lambda}\right\}_{\lambda \in \mathbb{N}}$, with $\log \left|\Sigma_{\lambda}\right| \leq \operatorname{poly}(\lambda)$, is a tuple of PPT algorithms (PIR.Enc, PIR.Res, PIR.Dec) with the following syntax.
$\bullet\left(\mathbb{q}, \operatorname{st}_{P I R}\right) \leftarrow \operatorname{PIR} . \operatorname{Enc}\left(1^{\lambda}, q, N\right):$ given $1^{\lambda}$ and some $q, N \in \mathbb{N}$ such that $q \leq N$, PIR.Enc outputs a query string $q$ and a secret state st $_{\text {PIR }}$.
$\bullet \mathfrak{\prec} \leftarrow \operatorname{PIR} . \operatorname{Res}\left(1^{\lambda}, \llbracket, D B\right):$ given a query string $q$ and a database $D B \in \Sigma_{\lambda}^{N}$, PIR.Res outputs a response string $火$.

- $x \leftarrow \mathrm{PIR} . \operatorname{Dec}\left(\mathrm{st}_{P I R}, \mathfrak{x}\right)$ : given a secret state $\mathrm{st}_{\text {PIR }}$ and a response string $\mathfrak{x}$, PIR.Dec outputs an element $x \in \Sigma_{\lambda}$.

We say that the scheme is a polylogarithmic PIR if $|x|=\operatorname{poly}(\lambda, \log N)$. We say that it is (perfectly) correct iffor all $q \leq N \leq$ $2^{\lambda}$ and $D B \in \Sigma_{\lambda}^{N}$ it holds that when setting $\left(\mathbb{q}, \mathrm{st}_{P I R}\right) \leftarrow \operatorname{PIR} . \operatorname{Enc}\left(1^{\lambda}, q, N\right)$, $\mathfrak{\varkappa} \leftarrow \operatorname{PIR} \cdot \operatorname{Res}\left(1^{\lambda}, \mathfrak{q}, D B\right), x \leftarrow \operatorname{PIR} . \operatorname{Dec}\left(\mathrm{st}_{P I R}, \mathfrak{x}\right)$, then $x=D B(q)$ holds with probability 1 . We say that the scheme is secure if for any sequence of $\left\{N_{\lambda}, q_{\lambda}, q_{\lambda}^{\prime}\right\}_{\lambda \in \mathbb{N}}$ such that $q_{\lambda}, q_{\lambda}^{\prime} \leq N_{\lambda}$, it holds $\mathbb{q} \approx \mathbb{q}^{\prime}$, where $\left(\mathbb{q}, \mathrm{st}_{P I R}\right) \leftarrow \operatorname{PIR} . E n c\left(1^{\lambda}, q_{\lambda}, N_{\lambda}\right)$ and $\left(\mathbb{q}^{\prime}, \mathrm{st}_{P I R}^{\prime}\right) \leftarrow \operatorname{PIR} . \operatorname{Enc}\left(1^{\lambda}, q_{\lambda}^{\prime}, N_{\lambda}\right)$.

[^13]
## C. 3 Computational No-signaling (Parallel Version)

Definition 15. Fix any constant $M \in \mathbb{N}$, any alphabet $\left\{\Sigma_{\lambda}\right\}_{\lambda \in \mathbb{N}}$, any $\left\{N_{\lambda}\right\}_{\lambda \in \mathbb{N}}$ such that $N_{\lambda} \in \mathbb{N}$, any function $\kappa_{\max }: \mathbb{N} \rightarrow \mathbb{N}$ such that $\kappa_{\max }(\lambda) \leq N_{\lambda}$, and any algorithm Algo such that for any $\lambda \in \mathbb{N}$, on input subsets $Q^{1}, \ldots, Q^{M} \subset\left[N_{\lambda}\right]$ of size at most $\kappa_{\max }(\lambda)$, Algo outputs (the truth tables of) functions $A^{1}, \ldots, A^{M}: Q \rightarrow \Sigma \cup\{\perp\}$ with an auxiliary output out.

Then, the algorithm Algo is adaptive $\kappa_{\max }$-computational no-signaling (CNS) if for any pPT distinguisher $\mathcal{D}$, any sufficiently large $\lambda \in \mathbb{N}$, any $Q^{i}, S^{i} \subset\left[N_{\lambda}\right]$ such that $Q^{i} \subseteq S^{i}$ and $\left|S^{i}\right| \leq \kappa_{\max }(\lambda)$ for $\forall i \in[M]$, and any $z \in\{0,1\}^{\text {poly }(\lambda)}$,

$$
\left|\begin{array}{l}
\operatorname{Pr}\left[\mathcal{D}\left(\text { out, }\left\{A^{i}\right\}_{i \in[M]}, z\right)=1 \mid\left(\text { out, }\left\{A^{i}\right\}_{i \in[M]}\right) \leftarrow \operatorname{Algo}\left(\left\{Q^{i}\right\}_{i \in[M]}\right)\right] \\
-\operatorname{Pr}\left[\mathcal{D}\left(\text { out, }\left\{\left.A^{i}\right|_{Q^{i}}\right\}_{i \in[M]}, z\right)=1 \mid\left(\text { out, }\left\{A^{i}\right\}_{i \in[M]}\right) \leftarrow \operatorname{Algo}\left(\left\{S^{i}\right\}_{i \in[M]}\right)\right]
\end{array}\right| \leq \operatorname{negl}(\lambda) .
$$

We remark that the above definition can be naturally extended for the case that Algo takes auxiliary inputs.

## C. 4 Low-degree Extension (LDE)

Let $\boldsymbol{F}$ be a finite field, and let $\boldsymbol{H}$ be a subset of $\boldsymbol{F}$, and let $m$ be an integer. Any function $f: \boldsymbol{H}^{m} \rightarrow\{0,1\}$ can be extended into a function $\hat{f}: \boldsymbol{F}^{m} \rightarrow \boldsymbol{F}$ such that $\hat{f}$ satisfies $\left.\hat{f}\right|_{\boldsymbol{H}^{m}} \equiv f$ and is an $m$-variate polynomial of degree at most $(|\boldsymbol{H}|-1)$ in each variable; the function $\hat{f}$ is called a low-degree extension (LDE) of $f$.

Low-degree extensions of strings. An LDE of a binary string $x$ of length $N$ can be obtained by choosing $\boldsymbol{H}$ and $m$ such that $N \leq|\boldsymbol{H}|^{m}$, identifying $\left\{1, \ldots,|\boldsymbol{H}|^{m}\right\}$ with $\boldsymbol{H}^{m}$ in a canonical way, and viewing $x$ as a function $x: \boldsymbol{H}^{m} \rightarrow\{0,1\}$ such that $x(i)=x_{i}$ for $\forall i \in[N]$ and $x(i)=0$ for $\forall i \in\left\{N+1, \ldots,|\boldsymbol{H}|^{m}\right\}$

## C. 5 Threshold Verifiers

We recall the definition of threshold verifiers [BHK17], which are used in the description and analysis of the PCP system of Kalai et al. [KRR14].

Definition 16. Given a PCP verifier $V=(\mathrm{Q}, \mathrm{D})$, the $t$-of- $n$ threshold verifier $\left(\mathrm{Q}^{\otimes n}, \mathrm{D}^{\geq t}\right)$ (where both $n$ and $t$ may be functions of $\lambda$ ) is defined as follows. $Q^{\otimes n}$ takes a security parameter $1^{\lambda}$ as input and does the following:

1. Compute $\left(Q_{i}, \mathrm{st}_{i}\right) \leftarrow \mathrm{Q}\left(1^{\lambda}\right)$ for each $i \in[n]$.
2. Output $\cup_{i=1}^{n} Q_{i}$ as the query and $\left(\left(Q_{1}, \mathrm{st}_{1}\right), \ldots,\left(Q_{n}, \mathrm{st}_{n}\right)\right)$ as the state.
$\mathrm{D}^{\geq t}$ takes input $\left(\left(\left(Q_{1}, \mathrm{st}_{1}\right), \ldots,\left(Q_{n}, \mathrm{st}_{n}\right)\right), x, \pi\right)$ and does the following:
3. Compute $y_{i} \leftarrow \mathrm{D}\left(\mathrm{st}_{i}, x,\left.\pi\right|_{Q_{i}}\right)$ for each $i \in[n]$.
4. Output 1 if at least $t$ of the $y_{i}$ 's are 1 ; otherwise outputs 0 .

For the special case where $t=n$, we write $\mathrm{D}^{\otimes n}$ instead of $\mathrm{D}^{\geq n}$.

## D Step 0: PCP for Checking View Consistency

Before starting to construct our commit-and-prove protocol, we first introduce the main building block, the PCP system (PCP.P, PCP.V) for checking the consistency of a pair of views of the MPC protocol $\Pi$. This PCP system is a variant of the PCP system of Kalai et al. [KRR14] (which is the main building block of their succinct argument), and the differences are that the statement to be proven is restricted to a specific form of 3SAT instances and that the verification includes several additional tests.

## D. 1 Preliminaries: Results from Kalai et al. [KRR14] and Subsequent Works

We recall some results from Kalai et al. [KRR14] and its subsequent works.
The first is a lemma that gives a PCP system (PCP. P $_{\text {KRR }}, ~ P C P . V_{\text {KRR }}$ ) for 3SAT such that any successful adaptive CNS prover can be converted into an adaptive local assignment generator.

Lemma D.1. There exists a PCP system (PCP. $\left.\mathrm{P}_{\mathrm{KRR}}, \mathrm{PCP} . \mathrm{V}_{\mathrm{KRR}}\right)$ for $3 S A T$, where $\mathrm{PCP} . \mathrm{V}_{\mathrm{KRR}}=\left(\mathrm{PCP} . \mathrm{Q}_{\text {KRR }}, \mathrm{PCP} . \mathrm{D}_{\text {KRR }}\right)$, that satisfies the following soundness property: There exists a PPT oracle machine Assign and a polynomial ${ }^{22} \kappa_{0}$ such that

[^14]for every negligible function $\epsilon$, every polynomial $\kappa_{\max }$, and every adaptive ( $\kappa_{0} \cdot \kappa_{\max }$ )-CNS cheating prover PCP. ${ }^{*}$, if it holds
\[

\operatorname{Pr}\left[$$
\begin{array}{l|l}
b=1 & \begin{array}{l}
(\mathrm{st}, Q) \leftarrow \operatorname{PCP} . \mathrm{Q}_{\mathrm{KRR}}^{\otimes \lambda}\left(1^{\lambda}\right) ; ~ \\
b:=\operatorname{PCP} . \mathrm{D}_{\mathrm{KRR}}^{\geq \lambda-\zeta}(\mathrm{st}, \varphi, \pi)
\end{array}
\end{array}
$$\right] \geq 1-\epsilon(\lambda),
\]

for $\zeta=\omega(\log \lambda)$ for infinitely many $\lambda \in \mathbb{N}($ let $\Lambda$ be the set of such $\lambda)$, then Assign ${ }^{\text {PCP. } P^{*}}$ is an adaptive $\kappa_{\max }$-local assignment generator for every sufficiently large $\lambda \in \Lambda$. Moreover, the distribution of $\varphi$ that is generated by $(\varphi, A) \leftarrow$ Assign ${ }^{P^{P C P} . P^{*}}(W)$ for any $W$ is computationally indistinguishable from the distribution of $\varphi$ that is generated by PCP.P* as above.
Proof. See, e.g., [BHK16, Lemma 6].
The next is a lemma that describes properties of (PCP. $P_{\text {KRR }}$, PCP. $V_{\text {KRR }}$ ).
Lemma D.2. The PCP system (PCP.P ${ }_{\text {KRR }}$, PCP. $\mathrm{V}_{\text {KRR }}$ ) and the oracle machine Assign given in Lemma D. 1 satisfy the following properties, where in the following, $N$ is used to denote the number of the variables in the 3SAT instance that is given to (PCP. $\mathrm{P}_{\mathrm{KRR}}, \mathrm{PCP} . \mathrm{V}_{\mathrm{KRR}}$ ) as the statement.

- (PCP. $\left.\mathrm{P}_{\mathrm{KRR}}, \mathrm{PCP} . \mathrm{V}_{\mathrm{KRR}}\right)$ has $\boldsymbol{F}, \boldsymbol{H}$, and $m$ as parameters, where $\boldsymbol{F}$ is a finite field of size $O\left(\log ^{2} N\right) \leq|\boldsymbol{F}| \leq$ polylog $(N)$, and $\boldsymbol{H}$ and $m$ are defined as $\boldsymbol{H}:=\{0, \ldots, \Theta(\log N)-1\}$ and $m:=\lceil\log N / \log |\boldsymbol{H}|\rceil$ so that $|\boldsymbol{H}|^{m} \geq N$.
- When PCP. $\mathrm{P}_{\text {KRR }}$ generates a proof $\pi$ for a 3 SAT instance $\varphi$ by using a satisfying assignment $\boldsymbol{x} \in\{0,1\}^{N}$ as a witness, the proof $\pi$ consists of several low-degree polynomials, and one of these polynomials, denoted as $X$, is an LDE (w.r.t. $\boldsymbol{F}, \boldsymbol{H}, m$ ) of $\boldsymbol{x}$.
- The query complexity of PCP. $\mathrm{V}_{\mathrm{KRR}}$ is at most $O\left(m|\boldsymbol{F}|^{2}\right) \leq \operatorname{polylog}(N)$, and the polynomial $\kappa_{0}$ given in Lemma D. 1 satisfies $\kappa_{0}=O\left(\lambda|\boldsymbol{F}|^{2}\right)$.
- Assign is defined as Assign := SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}$ for the oracle machine SelfCorr in Algorithm 1, where $D(X):=\boldsymbol{F}^{m}$ is the domain of $X .{ }^{23}$

Proof. See, e.g., the proof of [BHK16, Lemma 6].
The last is a technical lemma. Roughly speaking, it says that any function $f$ can be converted into a 3CNF formula $\varphi$ with the following property: $\varphi$ is a 3CNF formula that checks whether the $n$-bit value $x$ given as "the input value" and the $n$-bit value $y$ given as "the output value" satisfy $y=f(x)$; furthermore, it is carefully designed so that for any local assignment generator Assign for $\varphi$, if the assignment by Assign to "the input value" agrees with $x$, then the assignment by Assign to "the output value" must agree with $f(x)$. The formal statement is given below.

Lemma D.3. Fix any polynomials $N^{\prime}$ and $n$. There exist PPT algorithms Aug, $\mathrm{Aug}^{-1}$ and a polynomial ${ }^{24} \kappa_{\max }$ that satisfy the following properties.

Syntax. Fix the security parameter $\lambda$, and let $N^{\prime}:=N^{\prime}(\lambda)$ and $n:=n(\lambda)$. Aug takes as input a function $f:\{0,1\}^{n} \rightarrow\{0,1\}^{n}$ such that $\operatorname{Time}(f)=N^{\prime}$, and outputs a $3 C N F$ formula $\varphi$ of $N=\operatorname{poly}\left(N^{\prime}\right)$ variables. Aug ${ }^{-1}$ takes as input a $3 C N F$ formula $\varphi$, and outputs a function $f$ such that $\operatorname{Aug}(f)=\varphi$ if such $f$ exists, and outputs $\perp$ otherwise.

Satisfiability. Given any $f$ and $x=\left(x_{1}, \ldots, x_{n}\right) \in\{0,1\}^{n}$, one can efficiently compute a satisfying assignment to the variables in $\varphi:=\operatorname{Aug}(f)$. Furthermore, the resultant satisfying assignment satisfies the following.

- The assignment to the first $n$ variables, $\boldsymbol{w}_{\text {in }}=\left(w_{1}, \ldots, w_{n}\right)$, is $x=\left(x_{1}, \ldots, x_{n}\right)$.
- There exists a sequence of variables $\boldsymbol{w}_{\mathrm{in}, \mathrm{LDE}}$ such that the assignment to $\boldsymbol{w}_{\mathrm{in}, \mathrm{LDE}}$ is the LDE of $x$ w.r.t. some predetermined $\boldsymbol{F}^{\prime}, \boldsymbol{H}^{\prime}, m^{\prime}$.
- There exists a sequence of variables $\boldsymbol{w}_{\text {out,LDE }}$ such that the assignment to $\boldsymbol{w}_{\text {Out,LDE }}$ is the $\operatorname{LDE}$ of $f(x)=\left(y_{1}, \ldots, y_{n}\right)$ w.r.t. $\boldsymbol{F}^{\prime}, \boldsymbol{H}^{\prime}, m^{\prime}$, and therefore there exists a sequence of $n$ variables $\boldsymbol{w}_{\text {out }} \subset \boldsymbol{w}_{\text {out,LDE }}$ such that the assignment to $\boldsymbol{w}_{\text {out }}$ is $f(x)=\left(y_{1}, \ldots, y_{n}\right)$.

Let $I_{\text {in }}=\{1, \ldots, n\}$ denote the set of indices of the variables in $\boldsymbol{w}_{\text {in }}, I_{\text {in,LDE }}$ denote the set of indices of the variables in $\boldsymbol{w}_{\mathrm{in}, \mathrm{LDE}}, I_{\text {Out,LDE }}$ denote the set of indices of the variables in $\boldsymbol{w}_{\text {out,LDE }}$, and $I_{\text {out }}$ denote the set of indices of the variables in $\boldsymbol{w}_{\text {out }}{ }^{25}$

Security. If there exists a PPT machine Assign that is an adaptive $\kappa_{\max }$-local assignment generator for infinitely many $\lambda \in \mathbb{N}$ (let $\Lambda$ be the set of such $\lambda$ ) and it always outputs a $3 C N F$ formula $\varphi$ such that $\operatorname{Aug}^{-1}(\varphi) \neq \perp$, then the following three claims hold.

[^15]```
Algorithm 1 Self-Correction Procedure SelfCorrr \({ }_{d, D}^{\text {PCP.P }}\)
Parameter: \(d \in \mathbb{N}\) is an integer, and \(D \subseteq D(X)\) is a subset of the domain \(D(X)=\boldsymbol{F}^{m}\) of \(X\). (In this paper, \(D\) is always of
the form \(\left\{\left(c_{1}, \ldots, c_{k}, z_{k+1}, \ldots, z_{m}\right) \mid z_{k+1}, \ldots, z_{m} \in \boldsymbol{F}\right\} \subseteq \boldsymbol{F}^{m}\) for some constants \(k \in\{0, \ldots, m\}\) and \(c_{1}, \ldots, c_{k} \in \boldsymbol{F}\).)
Input: \(Q \subseteq[N]\), which is a subset of the indices of the variables in a 3SAT instance, and is thought of as a subset of
\(\boldsymbol{H}^{m} \subset \boldsymbol{F}^{m}=D(X)\) by a canonical mapping.
```

1. Run $\left(\tilde{Q}, s t_{Q}\right) \leftarrow$ SelfCorr. $Q_{D}(Q)$.
2. $\operatorname{Run}\left(\varphi, \pi^{*}\right) \leftarrow \operatorname{PCP}^{\left(P^{*}\right.}(\tilde{Q})$.
3. Run $A:=\operatorname{SelfCorrr} \operatorname{Rec}_{d}\left(\mathrm{st}_{Q}, \pi^{*}\right)$.
4. Output $(\varphi, A)$.

## $\underline{\text { Subroutine SelfCorr. } Q_{D}(Q):}$

1. For each $\boldsymbol{q} \in Q$, choose $\lambda$ random lines $L_{\boldsymbol{q}, 1}, \ldots, L_{\boldsymbol{q}, \lambda}: \boldsymbol{F} \rightarrow D$ such that each $L \in\left\{L_{\boldsymbol{q}, 1}, \ldots, L_{\boldsymbol{q}, \lambda}\right\}$ satisfies $L(0)=\boldsymbol{q}$. (Abort if $\boldsymbol{q} \notin D$.)
2. Output $\left(\tilde{Q}\right.$, st $\left._{Q}\right)$, where $\tilde{Q}:=\left\{L_{\boldsymbol{q}, j}(t)\right\}_{\boldsymbol{q} \in Q, j \in[\lambda], t \in \boldsymbol{F} \backslash\{0\}}$ and $\mathrm{st}_{Q}:=\left(Q,\left\{L_{\boldsymbol{q}, j}\right\}_{\boldsymbol{q} \in Q, j \in[\lambda]}\right)$.

Subroutine SelfCorr. $\operatorname{Rec}_{d}\left(\mathrm{st}_{Q}, X^{*}\right)$ :

1. Define $A: Q \rightarrow \boldsymbol{F} \cup\{\perp\}$ as follows. For each $\boldsymbol{q} \in Q$, check that there exists $c_{\boldsymbol{q}} \in \boldsymbol{F}$ such that

$$
\left|\left\{j \in[\lambda] \mid \operatorname{Recon}_{d}\left(\left\{X^{*}\left(L_{q, j}(t)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)=c_{q}\right\}\right| \geq 0.9 \lambda,
$$

let $A(\boldsymbol{q}):=c_{\boldsymbol{q}}$ if such $c_{\boldsymbol{q}}$ exists, and let $A(\boldsymbol{q}):=\perp$ otherwise.
2. Output $A$.

## Subroutine $\operatorname{Recon}_{d}\left(\left\{z_{t}\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)$ :

1. Obtain a degree- $d$ polynomial $P$ that satisfies $P(t)=z_{t}$ for $\forall t \in \boldsymbol{F} \backslash\{0\}$ through interpolation. (Output $\perp$ if the interpolation fails or $\perp \in\left\{z_{t}\right\}_{t \in \boldsymbol{F} \backslash\{0\}}$.
2. Output $P(0)$.

Claim D. 1 (From average correctness of input to average correctness of output). There exists a negligible function negl such that for every sufficiently large $\lambda \in \Lambda$ and every $x \in\{0,1\}^{n}$, it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\operatorname{Correct}(S) & S:=\left\{s_{i}\right\}_{i\left[\left[\log ^{2} \lambda\right]\right.}, \text { where } s_{i} \leftarrow I_{\text {in,LDE }} \\
\wedge \neg \operatorname{Correct}(T) & T:=\left\{t_{i}\right\}_{i \in\left[\log ^{2} \lambda\right]}, \text { where } t_{i} \leftarrow I_{\text {out,LDE }} \\
(\varphi, A) \leftarrow A \operatorname{Assign}(S \cup T)
\end{array}\right] \leq N \cdot \operatorname{neg|}(\lambda),
$$

where the events $\operatorname{Correct}(S)$ and $\operatorname{Correct}(T)$ are defined as follows. Given $\varphi$ and $f:=\operatorname{Aug}^{-1}(\varphi)$, let the correct assignment $A_{\text {corr }}($ to the variables in $\varphi$ ) be the assignment that is obtained from $x$ as above. Then, $\operatorname{Correct}(S)$ is the event that $A(s)=A_{\text {corr }}(s)$ holds for $\forall s \in S$ and $\operatorname{Correct}(T)$ is the event that $A(t)=A_{\text {corr }}(t)$ holds for $\forall t \in T$.
Claim D. 2 (From average correctness of output to worst-case correctness of output). There exists a negligible function negl such that for every sufficiently large $\lambda \in \Lambda$, every $x \in\{0,1\}^{n}$, and every $i^{*} \in \mathcal{I}_{\text {out }}$, it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\operatorname{Correct}(T) & T:=\left\{t_{i}\right\}_{i \in\left[\log ^{2} \lambda\right]}, \text { where } t_{i} \leftarrow I_{\text {out,LDE }} \\
\wedge A\left(i^{*}\right) \neq A_{\text {corr }}\left(i^{*}\right) & (\varphi, A) \leftarrow \operatorname{Assign}\left(T \cup\left\{i^{*}\right\}\right)
\end{array}\right] \leq \operatorname{negl}(\lambda),
$$

where $\operatorname{Correct}(T)$ and $A_{\text {corr }}$ are defined as in Claim D.1.
Claim D. 3 (From worst-case correctness of input to average correctness of input). There exist negligible functions negl, negl' such that for every sufficiently large $\lambda \in \Lambda$ and every $x \in\{0,1\}^{n}$, if it holds

$$
\operatorname{Pr}\left[A\left(i^{*}\right) \neq A_{\text {corr }}\left(i^{*}\right) \mid(\varphi, A) \leftarrow \operatorname{Assign}\left(\left\{i^{*}\right\}\right)\right] \leq \operatorname{negl}(\lambda)
$$

for every $i^{*} \in \mathcal{I}_{\text {in }}$, it also holds


Proof. See, e.g., [PR14, Section 4.4] and [PR17, Section 4.4]. (In their terminologies, Aug first transforms $f$ to an equivalent circuit $C$, then creates the "augmented version" of $C$, and then creates a 3 CNF formula that checks the computation of the augmented $C$.)

## D. 2 Our PCP system (PCP.P, PCP.V)

Let $f$ be an arbitrary functionality that is computable by $\Pi$.

## D.2.1 3CNF formula $\varphi_{f}^{\mu: v}$.

For each $\mu, v \in[M]$, we introduce 3 CNF formula $\varphi_{f}^{\mu: \nu}$, which takes views of the $\mu$-th and $\nu$-th parties $P^{\mu}, P^{v}$ of $\Pi$ as input and checks whether (1) these views are consistent and (2) both $P^{\mu}$ and $P^{v}$ output 1 in those views. Roughly speaking, we design $\varphi_{f}^{\mu: v}$ so that we can use Claim D. 1 in a round-by-round way. The details are given below.

Let $\operatorname{Next}_{f, \Pi}$ be the next message function of $\Pi$ and $N_{\text {round }}$ be the round complexity of $\Pi$, where the functionality to be computed by $\Pi$ is $f .{ }^{26}$ The input to $\mathrm{Next}_{f, \Pi}$ is of the form ( $\mathrm{st}_{\ell-1}^{i}$, i -msgs ${ }_{\ell}^{i}$ ) for some $i \in[M]$ and $\ell \in\left[N_{\text {round }}\right]$, where $\mathrm{st}_{\ell-1}^{i} \in$ $\{0,1\}^{n_{\text {st }}}$ is an internal state of party $P^{i}$ at the beginning of the $\ell$-th round and $\mathrm{i}-\mathrm{msgs}_{\ell}^{i}=\left(\mathrm{msg}_{\ell}^{i \leftarrow 1}, \ldots, \mathrm{msg}_{\ell}^{i \leftarrow M}\right) \in\{0,1\}^{M}$ is incoming messages that $P^{i}$ receives at the beginning of the $\ell$-th round. Given such input, $\operatorname{Next}_{f, \Pi}$ outputs $\left(\mathrm{st}_{\ell}^{i}, 0-\mathrm{msgs}_{\ell}^{i}\right)$, where $\mathrm{st}_{\ell}^{i} \in\{0,1\}^{n_{\mathrm{st}}}$ is the internal state at the end of the $\ell$-th round and o-msgs ${ }_{\ell}^{i}=\left(\mathrm{msg}_{\ell}^{i \rightarrow 1}, \ldots, \mathrm{msg}_{\ell}^{i \rightarrow M}\right) \in\{0,1\}^{M}$ is the outgoing messages that $P^{i}$ sends at the end of the $\ell$-th round. Let $N_{\mathrm{IO}}:=n_{\mathrm{st}}+M$ so that the input and the output of $\mathrm{Next}_{f, \Pi}$ are of length $N_{\text {Io }}$.

Let $\varphi_{f, \Pi}$ be the 3CNF Boolean formula that is obtained by $\varphi_{f, \Pi}:=\operatorname{Aug}\left(\operatorname{Next}_{f, \Pi}\right)$. Let $N_{\text {Aug }}$ be the number of the variables in $\varphi_{f, \Pi}$.

Now, for each $\mu, v \in[M]$, we define 3CNF Boolean formula $\varphi_{f}^{\mu: v}$ as follows, where the number of the variables is $N:=2 N_{\text {round }} N_{\text {Aug }}$ and they are denoted as $\left(\boldsymbol{w}_{1}^{\mu}, \cdots, \boldsymbol{w}_{N_{\text {round }}}^{\mu}, \boldsymbol{w}_{1}^{\nu}, \ldots, \boldsymbol{w}_{N_{\text {round }}}^{\nu}\right)$, where each $\boldsymbol{w}_{\ell}^{\xi}$ is a sequence of $N_{\text {Aug }}$ variables and thus can be viewed as the variables in $\varphi_{f, \Pi}$. (As explained below in Remark 3, the assignment to each $\boldsymbol{w}_{\ell}^{\xi}$ is supposed to be derived from the start state and incoming messages of round $\ell$ of party $P^{\xi}$ in the manner that is described in the "Satisfiability" paragraph in Lemma D.3.)

1. Let $\varphi_{1}$ be a 3CNF Boolean formula that checks $\varphi_{f, \Pi}\left(\boldsymbol{w}_{\ell}^{\xi}\right)=1$ for $\left.\forall \xi \in\{\mu, v\}, \ell \in\left[N_{\text {round }}\right]\right]^{27}$
2. For each $\boldsymbol{w}_{\ell}^{\xi}\left(\xi \in\{\mu, v\}, \ell \in\left[N_{\text {round }}\right]\right)$, let $\boldsymbol{w}_{\ell, \text { in }}^{\xi}=\left(\boldsymbol{w}_{\ell, \text { in }}^{\xi}(1), \ldots, \boldsymbol{w}_{\ell, \text { in }}^{\xi}\left(N_{\mathrm{IO}}\right)\right)$ and $\boldsymbol{w}_{\ell, \text { out }}^{\xi}=\left(\boldsymbol{w}_{\ell, \text { out }}^{\xi}(1), \ldots, \boldsymbol{w}_{\ell, \text { out }}^{\xi}\left(N_{\mathrm{IO}}\right)\right)$ be the sequences of $N_{\text {Io }}$ variables that correspond to $\boldsymbol{w}_{\text {in }}$ and $\boldsymbol{w}_{\text {out }}$ in Lemma D. 3 when $\boldsymbol{w}_{\ell}^{\xi}$ is viewed as the variables in $\varphi_{f, \Pi}=\operatorname{Aug}\left(\operatorname{Next}_{f, \Pi}\right)$. Then, let $\varphi_{2}$ be a 3CNF Boolean formula that checks each of the following. ${ }^{28}$

- $\boldsymbol{w}_{\ell, \text { out }}^{\xi}(i)=\boldsymbol{w}_{\ell+1, \text { in }}^{\xi}(i)$ holds for $\forall \xi \in\{\mu, v\}, \ell \in\left[N_{\text {round }}-1\right], i \in\left[n_{\text {st }}\right]$.
- $\boldsymbol{w}_{\ell, \text { out }}^{\mu}\left(n_{\text {st }}+v\right)=\boldsymbol{w}_{\ell+1, \text { in }}^{v}\left(n_{\mathrm{st}}+\mu\right)$ and $\boldsymbol{w}_{\ell, \text { out }}^{v}\left(n_{\mathrm{st}}+\mu\right)=\boldsymbol{w}_{\ell+1, \mathrm{in}}^{\mu}\left(n_{\mathrm{st}}+v\right)$ hold for $\forall \ell \in\left[N_{\text {round }}-1\right]$.
- $\boldsymbol{w}_{N_{\text {round }, \text { out }}}^{\xi}(1)=1$ holds for $\forall \xi \in\{\mu, \nu\}$.

3. Finally, let $\varphi_{f}^{\mu: v}$ be the 3CNF Boolean formula that checks that $\varphi_{1}$ and $\varphi_{2}$ are satisfied by $\left(\boldsymbol{w}_{1}^{\mu}, \cdots, \boldsymbol{w}_{N_{\text {round }}}^{\mu}, \boldsymbol{w}_{1}^{v}, \ldots, \boldsymbol{w}_{N_{\text {round }}}^{v}\right)$.
We think that each variable in $\varphi_{f}^{\mu: v}$ is indexed by an element in $[M] \times\left[N_{\text {round }}\right] \times\left[N_{\text {Aug }}\right]$ in the natural way (i.e., in the way that the $i$-th variable in $\boldsymbol{w}_{\ell}^{\xi}$ is indexed by $\left.(\xi, \ell, i)\right)$. For each $\xi \in\{\mu, v\}$ and $\ell \in\left[N_{\text {round }}\right]$, let $I_{\ell, \text { in }}^{\xi}=\left\{(\xi, \ell, 1), \ldots,\left(\xi, \ell, N_{\mathrm{ro}}\right)\right\}$ be the set of the indices of the variables in $\boldsymbol{w}_{\ell, \text { in }}^{\xi}, \mathcal{I}_{\ell, \text { out }}^{\xi}$ be the set of the indices of the variables in $\boldsymbol{w}_{\ell, \text { out }}^{\xi}, \mathcal{I}_{\ell, \text { in,LDE }}^{\xi}$ be the set of the indices of the variables in $\boldsymbol{w}_{\ell, \text { in,LDE }}^{\xi}$, and $\mathcal{I}_{\ell, \text { out,LDE }}^{\xi}$ be the set of the indices of the variables in $\boldsymbol{w}_{\ell, \text { out,LDE }}^{\xi}$, where $\boldsymbol{w}_{\ell, \text { in }, \text { LDE }}^{\xi}$ and $\boldsymbol{w}_{\ell, \text { Out,LDE }}^{\xi}$ are the sequences of variables that correspond to $\boldsymbol{w}_{\text {in,LDE }}$ and $\boldsymbol{w}_{\text {Out,LDE }}$ in Lemma D. 3 when $\boldsymbol{w}_{\ell}^{\xi}$ is viewed as the variables in $\varphi_{f, \Pi}=\operatorname{Aug}\left(\operatorname{Next}_{f, \Pi}\right)$. For each $\xi \in\{\mu, v\}$, let $I^{\xi}:=\left\{(\xi, 1,1), \ldots,\left(\xi, N_{\text {round }}, N_{\text {Aug }}\right)\right\}$ be the set of the indices of the variables in $\left(\boldsymbol{w}_{1}^{\xi}, \ldots, \boldsymbol{w}_{N_{\text {round }}}^{\xi}\right)$.
Remark 3. Given any views view ${ }^{\mu}$, view ${ }^{\nu}$ of $P^{\mu}, P^{\nu}$, we can obtain an assignment to the variables $\left(\boldsymbol{w}_{1}^{\mu}, \cdots, \boldsymbol{w}_{N_{\text {round }}}^{\mu}, \boldsymbol{w}_{1}^{v}, \ldots, \boldsymbol{w}_{N_{\text {round }}}^{v}\right)$ in $\varphi_{f}^{\mu: \nu}$ as follows.
4. Parse view ${ }^{\xi}$ as $\left(\mathrm{st}_{0}^{\xi}, i-\mathrm{msgs}_{1}^{\xi}, \ldots, i-\right.$ msgs $_{N_{\text {roumd }}}^{\xi}$ ) for $\forall \xi \in\{\mu, \nu\}$.

[^16]2. Repeat the following for $\xi \in\{\mu, \nu\}, \ell \in\left[N_{\text {round }}\right]$.
(a) Obtain an assignment to the variables $\boldsymbol{w}_{\ell}^{\xi}$ by using (st $t_{\ell-1}^{\xi}, \mathrm{i}-\mathrm{msgs}_{\ell}^{\xi}$ ), where $\boldsymbol{w}_{\ell}^{\xi}$ is viewed as the variables in $\varphi_{f, \Pi}=\operatorname{Aug}\left(\operatorname{Next}_{f, \Pi}\right)(\mathrm{cf}$. Lemma D.3).
(b) Compute $\left(\mathrm{st}_{\ell}^{\xi}\right.$, o-msgs $\left.{ }_{\ell}^{\xi}\right):=\operatorname{Next}_{f, \Pi}\left(\mathrm{st}_{\ell-1}^{\xi}\right.$, i-msgs $\left.{ }_{\ell}^{\xi}\right)$.

Note that when an assignment is computed in this way, the assignment to the variables in $\boldsymbol{w}_{1, \text { in }}^{\xi}$ (i.e., those that are indexed by $\mathcal{I}_{1, \mathrm{in}}^{\xi}$ ) is $\left(\mathrm{st}_{0}^{\xi}, \mathrm{i}-\mathrm{msgs}_{1}^{\xi}\right) \in\{0,1\}^{N_{\text {lo }}}$, and the assignment to the variables in $\boldsymbol{w}_{1}^{\xi}, \cdots, \boldsymbol{w}_{N_{\text {round }}}^{\xi}$ (i.e., those that are indexed by $I^{\xi}$ ) are computed from view ${ }^{\xi}$ alone for each $\xi \in\{\mu, \nu\}$. Also, note that if view ${ }^{\mu}$ and view ${ }^{\nu}$ are consistent and $P^{\mu}$ and $P^{\nu}$ output 1 in them, we obtain a satisfying assignment.

## D.2. 2 Prover PCP.P

The prover PCP.P of our PCP system is given in Algorithm 2.

```
Algorithm 2 Prover PCP.P of our PCP system (PCP.P, PCP.V)
Given input of the form \(\left(\mu, v, f\right.\), view \(^{\mu}\), view \(\left.{ }^{\nu}\right)\), PCP.P works as follows.
```

1. Obtain an assignment $x^{\mu: v}$ to the variables in $\varphi_{f}^{\mu: v}$ by using view ${ }^{\mu}$, view ${ }^{v}$ as described in Remark 3.
2. Output $\pi^{\mu: v}=\left(X^{\mu: v}, \ldots\right):=\operatorname{PCP} . \mathrm{P}_{\mathrm{KRR}}\left(\varphi_{f}^{\mu: v}, x^{\mu: v}\right)$ as the PCP proof.

Regarding PCP.P, we introduce several notations and without-loss-of-generality assumptions. (Roughly, we introduce them so that given $X^{\mu: \nu}$, which is an LDE of $x^{\mu: \nu}$, the verifier can evaluate LDEs of several substrings of $x^{\mu: \nu}$.)

Notations. Let $\boldsymbol{F}, \boldsymbol{H}$, and $m$ be the parameters of (PCP.P ${ }_{\text {KRR }}$, PCP. $V_{\text {KRR }}$ ) used in PCP.P. Let $m_{\text {round }}:=$ $\log N_{\text {round }} / \log |\boldsymbol{H}|, m_{\text {Aug }}:=\log N_{\text {Aug }} / \log |\boldsymbol{H}|$, and $m_{\text {IO }}:=\log N_{\text {io }} / \log |\boldsymbol{H}|$ so that $N_{\text {round }}=|\boldsymbol{H}|^{m_{\text {round }}}, N_{\text {Aug }}=|\boldsymbol{H}|^{m_{\text {Aug }}}$, and $N_{\mathrm{IO}}=|\boldsymbol{H}|^{m_{\mathrm{Io}}}$. (It is assumed that $m_{\text {round }}, m_{\text {Aug }}, m_{\mathrm{IO}}$ are integers.) Given assignment $x^{\mu: v}:[M] \times\left[N_{\text {round }}\right] \times\left[N_{\text {Aug }}\right] \rightarrow\{0,1\}$ to the variables in $\varphi_{f}^{\mu: v}$, let $x_{\ell, \text { in }}^{\xi}:\left[N_{\mathrm{Io}}\right] \rightarrow\{0,1\}$ be its assignment to the variables indexed by $\mathcal{I}_{\ell, \text { in }}^{\xi}$ (i.e., $x_{\ell, \text { in }}^{\xi}$ is defined so that $x_{\ell, \text { in }}^{\xi}(i)=x^{\mu: \nu}(\xi, \ell, i)$ for $\left.\forall i \in\left[N_{\mathrm{IO}}\right]\right)$, and $X_{\ell, \text { in }}^{\xi}$ be the LDE of $x_{\ell, \text { in }}^{\xi}$ w.r.t. $\boldsymbol{F}, \boldsymbol{H}, m_{\mathrm{IO}}$. Similarly, let $x^{\xi}:\left[N_{\text {round }}\right] \times\left[N_{\mathrm{Aug}}\right] \rightarrow\{0,1\}$ be its assignment to the variables indexed by $\mathcal{I}^{\xi}$ (i.e., $x^{\xi}$ is defined so that $x^{\xi}(\ell, i)=x^{\mu: \nu}(\xi, \ell, i)$ for $\forall \ell \in\left[N_{\text {round }}\right], i \in\left[N_{\text {Aug }}\right]$ ), and $X^{\xi}$ be the LDE of $x^{\xi}$ w.r.t. $\boldsymbol{F}, \boldsymbol{H}, m-1$. Let $D(X):=\boldsymbol{F}^{m}$ denote the domain of $X^{\mu: v}$.

WLOG assumptions. We assume that the parameters $\boldsymbol{F}, \boldsymbol{H}, m$ of (PCP. $\mathrm{P}_{\mathrm{KRR}}$, PCP. $\mathrm{V}_{\mathrm{KRR}}$ ) and the LDE $X^{\mu: v}$ of $x^{\mu: v}$ satisfy the following. ${ }^{29}$

- $\boldsymbol{H}:=\left\{0, \ldots, \log \left(N_{\text {round }} N_{\text {Aug }}\right)-1\right\}$ and $m:=m_{\text {round }}+m_{\text {Aug }}+1$.
- For each $\xi \in\{\mu, \nu\}$, there exists $\boldsymbol{z}_{\xi} \in \boldsymbol{H}$ such that $X^{\xi}$ can be evaluated on any point in $\boldsymbol{F}^{m-1}$ by evaluating $X^{\mu: v}$ on a point in $D\left(X^{\xi}\right):=\left\{\left(z_{\xi}, z\right) \mid \boldsymbol{z} \in \boldsymbol{F}^{m-1}\right\}$.
- For each $\xi \in\{\mu, \nu\}$ and $\ell \in\left[N_{\text {round }}\right]$, there exists $\boldsymbol{z}_{\ell} \in \boldsymbol{H}^{m_{\text {round }}}$ and $z_{\text {in }} \in \boldsymbol{H}^{m_{\text {Aug }}-m_{\text {Io }}}$ such that $X_{\ell, \text { in }}^{\xi}$ can be evaluated on any point in $\boldsymbol{F}^{m_{10}}$ by evaluating $X^{\mu, v}$ on a point in $D\left(X_{\ell, \text { in }}^{\xi}\right):=\left\{\left(\boldsymbol{z}_{\xi}, z_{\ell}, z_{\mathrm{in}}, z\right) \mid \boldsymbol{z} \in \boldsymbol{F}^{m_{\mathrm{lo}}}\right\}$.


## D.2.3 Verifier PCP.V = (PCP.Q, PCP.D).

The verifier PCP.V of our PCP system is given in Algorithm 3. (It becomes clear later in Appendix E why PCP.V additionally does various types of low-degree tests.)

It can be verified easily that PCP.V can be decomposed into a query algorithm PCP.Q and a decision algorithm PCP.D naturally, and that the query complexity of PCP.V is asymptotically the same as that of PCP.V ${ }_{\text {KRR }}$, i.e., is at most $O\left(m|\boldsymbol{F}|^{2}\right) \leq \operatorname{polylog}(N)$.

[^17]Algorithm 3 Verifier PCP.V of our PCP system (PCP.P, PCP.V)
Given input of the form $(\mu, v, f)$, the verifier $V$ does the following tests.

1. Do the same tests as $\operatorname{PCP} . \mathrm{V}_{\mathrm{KRR}}\left(\varphi_{f}^{\mu: \nu}\right)$.
2. $D\left(X_{1, \text { in }}^{\xi}\right)$-parallel Low-degree Test for $X^{\mu: v}$ : Choose random points $\boldsymbol{r}_{0} \in \boldsymbol{F}^{m}$ and $\boldsymbol{r}_{1} \in\left\{0^{1+m_{\mathrm{round}}+m_{\text {Aug }}-m_{\mathrm{lo}}}\right\} \times \boldsymbol{F}^{m_{\mathrm{lo}}}$, define a line $L: \boldsymbol{F} \rightarrow \boldsymbol{F}^{m}$ as $L(\alpha)=\boldsymbol{r}_{0}+\alpha \cdot \boldsymbol{r}_{1}$, and query $X^{\mu: v}$ on all the points $\{L(t)\}_{t \in \boldsymbol{F}}$. Check that the univariate polynomial $X^{\mu: v} \circ L: \boldsymbol{F} \rightarrow \boldsymbol{F}$ has degree at most $m_{\mathrm{IO}}|\boldsymbol{H}|$.
3. $D\left(X^{\xi}\right)$-parallel Low-degree Test for $X^{\mu: v}$ : Choose random points $\boldsymbol{r}_{0} \in \boldsymbol{F}^{m}$ and $\boldsymbol{r}_{1} \in\{0\} \times \boldsymbol{F}^{m-1}$, define a line $L: \boldsymbol{F} \rightarrow \boldsymbol{F}^{m}$ as $L(\alpha)=\boldsymbol{r}_{0}+\alpha \cdot \boldsymbol{r}_{1}$, and query $X^{\mu: v}$ on all the points $\{L(t)\}_{t \in \boldsymbol{F}}$. Check that the univariate polynomial $X^{\mu: v} \circ L: \boldsymbol{F} \rightarrow \boldsymbol{F}$ has degree at most $(m-1)|\boldsymbol{H}|$.
4. Low-degree Test for $X^{\mu \cdot \nu}$, conditioned on $L(0) \in D\left(X_{1, \mathrm{in}}^{\xi}\right)$ : For every $\xi \in\{\mu, v\}$, do the following: Choose a random line $L: \boldsymbol{F} \rightarrow \boldsymbol{F}^{m}$ such that $L(0) \in D\left(X_{1, \text { in }}^{\xi}\right)$, and query $X^{\mu: v}$ on all the points $\{L(t)\}_{t \in \boldsymbol{F}}$; then, check that the univariate polynomial $X^{\mu: v} \circ L: \boldsymbol{F} \rightarrow \boldsymbol{F}$ has degree at most $m|\boldsymbol{H}|$.
5. Low-degree Test for $X^{\mu: \nu}$, conditioned on $L(0) \in D\left(X^{\xi}\right)$ : For every $\xi \in\{\mu, \nu\}$, do the following: Choose a line $L: \boldsymbol{F} \rightarrow \boldsymbol{F}^{m}$ such that $L(0) \in D\left(X^{\xi}\right)$, and query $X^{\mu: v}$ on all the points $\{L(t)\}_{t \in \boldsymbol{F}}$; then, check that the univariate polynomial $X^{\mu: v} \circ L: \boldsymbol{F} \rightarrow \boldsymbol{F}$ has degree at most $m|\boldsymbol{H}|$.
6. Low-degree Test for $X_{1, \mathrm{i}}^{\xi}$ : For every $\xi \in\{\mu, \nu\}$, do the following: Choose a line $L: \boldsymbol{F} \rightarrow D\left(X_{1, \mathrm{in}}^{\xi}\right)$, and query $X^{\mu: v}$ on all the points $\{L(t)\}_{t \in \boldsymbol{F}}$; then, check that the univariate polynomial $X^{\mu: v} \circ L: \boldsymbol{F} \rightarrow \boldsymbol{F}$ has degree at most $m_{\text {⿺夂 }}|\boldsymbol{H}|$.
7. Low-degree Test for $X^{\xi}$ : For every $\xi \in\{\mu, v\}$, do the following: Choose a line $L: \boldsymbol{F} \rightarrow D\left(X^{\xi}\right)$, and query $X^{\mu: v}$ on all the points $\{L(t)\}_{t \in \boldsymbol{F}}$; then, check that the univariate polynomial $X^{\mu: v} \circ L: \boldsymbol{F} \rightarrow \boldsymbol{F}$ has degree at most $(m-1)|\boldsymbol{H}|$.
8. Low-degree Test for $X^{\mu: v}$ : Choose a line $L: \boldsymbol{F} \rightarrow D(X)$, and query $X^{\mu: v}$ on all the points $\{L(t)\}_{t \in \boldsymbol{F}}$. Check that the univariate polynomial $X^{\mu: v} \circ L: \boldsymbol{F} \rightarrow \boldsymbol{F}$ has degree at most $m|\boldsymbol{H}|$.

## D.2.4 Security.

(PCP.P, PCP.V) inherits security from (PCP.P ${ }_{\text {KRR }}$, PCP.V ${ }_{\text {KRR }}$ ). Specifically, since PCP.P just executes PCP. $P_{\text {KRR }}$ on a specific form of a 3SAT instance, and PCP.V does all the tests that PCP. $V_{\text {KRR }}$ does, we obtain the following lemma by combining Lemma D.1, Lemma D.2, and Lemma D.3.

Lemma D.4. The PCP system (PCP.P, PCP.V) in Algorithm 2 and Algorithm 3 satisfies the following soundness property. There exist polynomials $\kappa_{0}$ and $\kappa_{\max }$ such that for every negligible function $\epsilon$, every $\alpha, \beta \in[M]$, and every adaptive ( $\kappa_{0} \cdot \kappa_{\max }$ )CNS cheating prover PCP. $\mathrm{P}^{*}$, if it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left.\begin{array}{l}
(\text { st }, Q) \leftarrow \operatorname{PCP}^{2} \mathrm{Q}^{\otimes \lambda}(\alpha, \beta) ;(f, \pi) \leftarrow{\operatorname{PCP} . \mathrm{P}^{*}(Q) ;}^{b:=\operatorname{PCP} . \mathrm{D}^{\geq \lambda-\zeta}(\mathrm{st}, f, \pi)}
\end{array}\right] \geq 1-\epsilon(\lambda),
\end{array}\right]
$$

for $\zeta=\omega(\log \lambda)$ for infinitely many $\lambda \in \mathbb{N}($ let $\Lambda$ be the set of such $\lambda)$, then the procedure SelfCorr in Algorithm 1 satisfies the following four claims.

Claim D.4. SelfCorrr ${ }_{m|\boldsymbol{H}|, D(X)}^{\mathrm{PCP}} \mathrm{P}^{*}$ is an adaptive $\kappa_{\max }$-local assignment generator for every sufficiently large $\lambda \in \Lambda .{ }^{30}$ Moreover, the distribution of $f$ that is generated by $(f, A) \leftarrow \operatorname{SelfCorr}_{m|\boldsymbol{H}|, D(X)}^{\mathrm{PCP} \cdot \mathrm{P}^{*}}(W)$ for any $W$ is computationally indistinguishable from the distribution of $f$ that is generated by PCP.P*.

Claim D.5. There exists a negligible function negl such that for every sufficiently large $\lambda \in \Lambda$ and for $\forall\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]} \in\left(\{0,1\}^{N_{\text {to }}}\right)^{M}, \xi \in\{\alpha, \beta\}, \ell \in\left[N_{\text {round }}\right]$, it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\operatorname{Correct}(S) & \begin{array}{l}
S:=\left\{s_{i}\right\}_{i \in\left[\log ^{2} \lambda\right]}, \text { where } s_{i} \leftarrow \mathcal{I}_{\ell, \text { in,LDE }}^{\xi} \\
\wedge \neg \operatorname{Correct}(T)
\end{array} \\
T:=\left\{t_{i}\right\}_{i \in\left[\log ^{2} \lambda\right]}, \text { where } t_{i} \leftarrow \mathcal{I}_{\ell, \text { out,LDE }}^{\xi} \\
(f, A) \leftarrow \operatorname{SelfCorr}_{m|\boldsymbol{H}|, D(X)}^{\mathrm{PCP} \mathrm{P}^{*}}(S \cup T)
\end{array}\right] \leq N_{\mathrm{Aug}} \cdot \operatorname{negl}(\lambda),
$$

where the events $\operatorname{Correct}(S)$ and $\operatorname{Correct}(T)$ are defined as follows. Let the correct views $\left\{\operatorname{view}^{\mu}\right\}_{\mu \in[M]}$ be the views of the parties in the execution of $\Pi$ on $\left(f,\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}\right),{ }^{31}$ and let the correct assignment $A_{\text {corr }}\left(\right.$ to the variables in $\left.\varphi_{f}^{\alpha ; \beta}\right)$

[^18]be the assignment that is obtained from $\left(\mathrm{view}^{\alpha}, \mathrm{view}^{\beta}\right)$ as in Remark 3. Then, Correct $(S)$ is the event that $A(s)=A_{\text {corr }}(s)$ holds for $\forall s \in S$ and $\operatorname{Correct}(T)$ is the event that $A(t)=A_{\text {corr }}(t)$ holds for $\forall t \in T$.

Claim D.6. There exists a negligible function negl such that for every sufficiently large $\lambda \in \Lambda$ and for $\forall\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]} \in\left(\{0,1\}^{N_{\text {No }}}\right)^{M}, \xi \in\{\alpha, \beta\}, \ell \in\left[N_{\text {round }}\right]$, and $i^{*} \in I_{\ell, \text { out }}^{\xi}$, it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\operatorname{Correct}(T) & T:=\left\{t_{i}\right\}_{i \in\left[\log ^{2} \lambda\right]}, \text { where } t_{i} \leftarrow \mathcal{I}_{\ell, \text { out,LDE }}^{\xi} \\
\wedge A\left(i^{*}\right) \neq A_{\text {corr }}\left(i^{*}\right) & (f, A) \leftarrow \operatorname{SelfCorr}_{m|\boldsymbol{H}|, D(X)}\left(T \cup\left\{i^{*}\right\}\right)
\end{array}\right] \leq \operatorname{negl}(\lambda),
$$

where $\operatorname{Correct}(T)$ and $A_{\text {corr }}$ are defined as in Claim D.5.
Claim D.7. There exist negligible functions negl, negl' such that for every sufficiently large $\lambda \in \Lambda$ and for $\forall\left\{\left(\mathrm{st}_{0}^{\mu} \text {, } \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]} \in\left(\{0,1\}^{N_{\text {Io }}}\right)^{M}$ and $\xi \in\{\alpha, \beta\}$, if it holds

$$
\operatorname{Pr}\left[A\left(i^{*}\right) \neq A_{\text {corr }}\left(i^{*}\right) \mid(f, A) \leftarrow \operatorname{SelfCorr}_{m|\boldsymbol{H}|, D(X)}^{\mathrm{PCP} \cdot \mathrm{P}^{*}}\left(\left\{i^{*}\right\}\right)\right] \leq \operatorname{negl}(\lambda)
$$

for every $i^{*} \in I_{1, \mathrm{in}}^{\xi}$, it also holds

$$
\operatorname{Pr}\left[\neg \operatorname{Correct}(S) \left\lvert\, \begin{array}{l}
S:=\left\{s_{i}\right\}_{i \in\left[\log ^{2} \lambda\right], \text { where } s_{i}} \leftarrow I_{1, \mathrm{in}, \mathrm{LDE}}^{\xi} \\
(f, A) \leftarrow \operatorname{SelfCorr}_{m|\boldsymbol{H}|, D(X)}^{\mathrm{P}}(S)
\end{array}\right.\right] \leq \operatorname{negl}^{\prime}(\lambda),
$$

where $\operatorname{Correct}(S)$ and $A_{\text {corr }}$ are defined as in Claim D.5. Similarly, for $\ell \in\left\{2, \ldots, N_{\text {round }}\right\}$, if it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\operatorname{Correct}(T) & T:=\left\{t_{i}\right\}_{i \in\left[\log { }^{2} \lambda\right],} \text { where } t_{i} \leftarrow \mathcal{I}_{\ell-1, \text { out,LDE }}^{\xi} \\
\wedge A\left(i^{*}\right) \neq A_{\text {corr }}\left(i^{*}\right) & (f, A) \leftarrow \operatorname{SelfCorr}_{m|\boldsymbol{H}|, D(X)}^{\mathrm{PPP} \cdot \mathrm{P}^{*}}\left(T \cup\left\{i^{*}\right\}\right)
\end{array}\right] \leq \operatorname{negl}(\lambda)
$$

for every $i^{*} \in I_{\ell, \text { in }}^{\xi}$, it also holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\operatorname{Correct}(T) & T:=\left\{t_{i}\right\}_{i \in\left[\log ^{2} \lambda\right]}, \text { where } t_{i} \leftarrow I_{\ell-1, \text { out,LDE }}^{\xi} \\
\wedge \neg \operatorname{Correct}(S) & S:=\left\{s_{i}\right\}_{i \in\left[\log ^{2} \lambda\right], \text { where }} s_{i} \leftarrow I_{1, \text { n,LDE }}^{\xi} \\
(f, A) \leftarrow \operatorname{SelfCorr}_{m|\boldsymbol{|}|, D(X)}^{\mathrm{PPP} \cdot \mathrm{P}^{*}}(S \cup T)
\end{array}\right] \leq \operatorname{negl}^{\prime}(\lambda) .
$$

## E Step 1: Non-WI Scheme with (1 - negl)-Soundness against Well-behaving CNS Provers

As the first step to our commit-and-prove protocol, we give a non-WI commit-and-prove protocol $\left\langle C_{1}, R_{1}\right\rangle$ that is (1-negl)sound against "well-behaving" CNS provers.

## E. 1 Protocol Description

The formal description of $\left\langle C_{1}, R_{1}\right\rangle$ is given in Algorithm 4 and Algorithm 5, where $\zeta=\omega(\log \lambda)$ is a parameter for the use of Lemma D.4, and the subroutines SelfCorr and LD-Test are defined in Algorithm 1 and Algorithm 6 respectively. We remark that $\operatorname{Time}(f)$ is assumed to be known to the verifier in the commit phase so that the parameters $\boldsymbol{F}, \boldsymbol{H}, m_{\text {IO }}$ of (PCP.P, PCP.V) can be determined.

The communication complexity is polylogarithmic in $\operatorname{Time}(f)$ since the query complexity of (PCP.P, PCP.V) is polylogarithmic in the size of $\varphi_{f}^{\mu: \nu}$.
Remark 4 (On the open phase). At first sight, the open phase might seem to be unnecessarily too complex since the receiver does self-correction by using lines on $D\left(X_{1, \text { in }}^{\mu}\right) \subset \boldsymbol{F}^{m}$ rather than those on $\boldsymbol{F}^{m_{10}}$. Roughly speaking, the open phase is defined in this way since in the proof of soundness (where we construct an extractor that converts any successful prover into a successful decommitter), we consider an extractor that forwards the receiver's decommitment queries to the cheating prover by observing that queries in the open phase can be viewed as those in the prove phase. (Note that in the open phase, the receiver queries to $\tilde{X}^{\mu ; \mu}: \boldsymbol{F}^{m} \rightarrow \boldsymbol{F}$, which has the same domains as the LDEs that the receiver queries to in the prove phase.)

## E. 2 Proof of Binding

We prove the binding property against cheating adversaries that are no-signaling and well-behaving in the following sense.
Definition 17 (No-signaling committer-decommitter). A cheating committer-decommitter $C_{1}^{*}=\left(\mathrm{C}\right.$. Com $_{1}^{*}$, C.Dec $\left._{1}^{*}\right)$ against


## Algorithm 4 Commit Phase and Prove Phase of $\left\langle C_{1}, R_{1}\right\rangle$

Commit Phase:
Round 1: Given $x_{\text {сом }}$ as input, C.Com ${ }_{1}$ does the following.

1. Sample random $x_{\mathrm{MPC}}^{1}, \ldots, x_{\mathrm{MPC}}^{M} \in\{0,1\}^{n}$ such that $x_{\mathrm{MPC}}^{1} \oplus \cdots \oplus x_{\mathrm{MPC}}^{M}=x_{\text {Сом }}$
2. For each $\mu \in[M]$, define $X_{1, \mathrm{in}}^{\mu}: \boldsymbol{F}^{m_{10}} \rightarrow \boldsymbol{F}$ as follows.
(a) Define $x_{1, \text { in }}^{\mu} \in\{0,1\}^{N_{\text {lo }}}$ as follows: sample random $r_{\text {MPC }}^{\mu} \in\{0,1\}^{n_{\text {MPc }}}$ and let st $t_{0}^{\mu}:=x_{\text {MPC }}^{\mu} \| r_{\text {MPC }}^{\mu} \in\{0,1\}^{n_{\mathrm{st}}}$, i-msgs ${ }_{1}^{\mu}:=0^{M} \in\{0,1\}^{M}, x_{1, \text { in }}^{\mu}:=\mathrm{st}_{0}^{\mu} \| \mathrm{i}-\mathrm{msgs}_{1}^{\mu}$.
(b) Let $X_{1, \text { in }}^{\mu}$ be the low-degree extension of $x_{1, \text { in }}^{\mu}$ (w.r.t. $\boldsymbol{F}, \boldsymbol{H}, m_{10}$ ).
3. Output an empty string $\varepsilon$ as the commitment, and $\left\{X_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}$ as the internal state.

## Prove Phase:

Round 1: R.Prv. $\mathrm{Q}_{1}$ runs $\left(\mathrm{st}_{V}^{\mu, v}, Q^{\mu, v}\right) \leftarrow{\operatorname{PCP} . \mathrm{Q}^{\otimes \lambda}}(\mu, v)$ for every $\mu, v \in[M]$, and outputs $\left\{Q^{\mu, v}\right\}_{\mu, v \in[M]}$ as the query and $\left\{\mathrm{st}_{V}^{\mu, \nu}\right\}_{\mu, \nu \in[M]}$ as the internal state.
Round 2: Given (st ${ }_{C}, f,\left\{Q^{\mu, \nu}\right\}_{\mu, \nu \in[M]}$ ) as input, C. $\operatorname{Prv}_{1}$ does the following.

1. Run the MPC protocol $\Pi$ in the head on $\left(f^{\prime},\left\{\left(\mathrm{St}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}\right)$, where $f^{\prime}$ is defined as $f^{\prime}:\left(y^{1}, \ldots, y^{M}\right) \mapsto$ $f\left(y^{1} \oplus \cdots \oplus y^{M}\right)$. (Here, $\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}$ is recovered from st ${ }_{C}=\left\{X_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$.) Let $\left\{\text { view }^{\mu}\right\}_{\mu \in[M]}$ be the view of the parties in this execution.
2. Run $\pi^{\mu: v} \leftarrow \operatorname{PCP} . \mathrm{P}\left(\mu, v, f^{\prime}\right.$, view $^{\mu}$, view $\left.^{\nu}\right)$ for every $\mu, v \in[M]$.
3. Output $\left\{\left.\pi^{\mu: \nu}\right|_{Q^{\mu v \nu}}\right\}_{\mu, v \in[M]}$ as the proof.

Verification: Given ( $\left.\mathrm{st}_{R}, \operatorname{com}, f,\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right)$ (where $\mathrm{st}_{R}=\left\{\mathrm{st}_{V}^{\mu, v}\right\}_{\mu, v \in[M]}$ ), R.Prv. $\mathrm{D}_{1}$ does the following.

1. Run $b^{\mu: v}:=\operatorname{PCP} . \mathrm{D}^{\geq \lambda-\zeta}\left(\mathrm{St}_{V}^{\mu: \nu}, f^{\prime}, \pi^{* \mu: v}\right)$ for every $\mu, v \in[M]$.
2. Output 1 if and only if $b^{\mu: v}=1$ for every $\mu, v \in[M]$.

Definition 18 (Well-behaving committer-decommitter). A cheating committer-decommitter $C_{1}^{*}=\left(\mathrm{C} . \mathrm{Com}_{1}^{*}, \mathrm{C}_{1} \mathrm{Dec}_{1}^{*}\right)$ against $\left\langle C_{1}, R_{1}\right\rangle$ is well-behaving if the following holds.

- Consistency on $D\left(X_{1, \mathrm{in}}^{\mu}\right)$ : For $\forall \lambda \in \mathbb{N}$ and $\forall\left\{Q_{0}^{\mu}\right\}_{\mu \in[M]},\left\{Q_{1}^{\mu}\right\}_{\mu \in[M]}$ such that $Q_{0}^{\mu}, Q_{1}^{\mu} \subseteq D\left(X_{1, \mathrm{in}}^{\mu}\right)$, it holds

Note that well-behaving adversaries do not give different answers to the same query during the binding security experiment.
The formal statement of the binding property of $\left\langle C_{1}, R_{1}\right\rangle$ is given below. In the following, P-LD-Test is a parallel version of LD-Test; see Algorithm 7.

Lemma E.1. There exists a polynomial $\kappa_{\mathrm{dec}}$ such that the following holds. Let $\epsilon$ be any negligible function and $C_{1}^{*}=\left(\mathrm{C}_{1} \mathrm{Com}_{1}^{*}, \mathrm{C}^{2} \mathrm{Dec}_{1}^{*}\right)$ be any well-behaving $\kappa_{\mathrm{dec}}-$ CNS cheating committer-decommitter against $\left\langle C_{1}, R_{1}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following binding condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{1}^{*}\right.$, R.Com ${ }_{1}$ 〉.

- Binding Condition: If it holds
ford $:=m_{I O}|\boldsymbol{H}|$ and $D_{b}^{\mu}:=D\left(X_{1, \mathrm{in}}^{\mu}\right)$, then for every $i \in[n]$ it holds $\operatorname{Pr}\left[b_{B A D}=1\right] \leq \operatorname{neg} \mid(\lambda)$ in the following probabilistic experiment.

1. For each $\forall b \in\{0,1\}$, sample $\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}$ by $\left(\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}\right.$, st $\left._{b}\right) \leftarrow$ R.Dec. $Q_{1}(i)$.
2. Run $\left\{\tilde{X}^{*}{ }_{b}^{\mu \cdot \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \operatorname{C.} \operatorname{Dec}_{1}^{*}\left(\operatorname{st}_{C},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$.
3. Let $b_{B A D}:=1$ if and only if $x_{0}^{*} \neq \perp \wedge x_{1}^{*} \neq \perp \wedge x_{0}^{*} \neq x_{1}^{*}$ holds, where $x_{b}^{*}:=$ R.Dec. $\mathrm{D}_{1}\left(\mathrm{st}_{b}, \operatorname{com},\left\{\tilde{X}^{\mu}{ }_{b}^{\mu, \mu}\right\}_{\mu \in[M]}\right)$ for each $b \in\{0,1\}$.

## Algorithm 5 Open Phase of $\left\langle C_{1}, R_{1}\right\rangle$

## Open Phase:

Round 1: Given $i$ as input, R.Dec. Q $_{1}$ does the following.

1. Define $Q_{0}^{\mu}, Q_{1}^{\mu} \subset D\left(X_{1, \text { in }}^{\mu}\right)$ for every $\mu \in[M]$ as follows.

- (Low-degree Test on $\left.X_{1, \text { in }}^{\mu}\right)$ Run $\left(Q_{0}^{\mu}\right.$, st $\left._{0}^{\mu}\right) \leftarrow$ LD-Test. $Q_{D\left(X_{1, \mathrm{n}}^{\mu}\right)}$.
- (Self-correction) Run $\left(Q_{1}^{\mu}\right.$, $\left.\mathrm{st}_{1}^{\mu}\right) \leftarrow$ SelfCorr. $\mathrm{Q}_{D\left(X_{1, \mathrm{in}}^{\mu}\right)}(\{(\mu, 1, i)\})$, where $(\mu, 1, i) \in[M] \times\left[N_{\text {round }}\right] \times\left[N_{\text {Aug }}\right]$ is the index of a variable in the 3 CNF formula $\varphi_{f}^{\mu: \nu}$ (concretely, the variable that corresponds to the $i$-th bit of $P^{\mu}$ s internal state at the beginning of Round 1 , or equivalently the $i$-th bit of $P^{\mu}$ 's input).

Then, let $Q^{\mu}:=Q_{0}^{\mu} \cup Q_{1}^{\mu}$.
2. Output $\left\{Q^{\mu}\right\}_{\mu \in[M]}$ as the query, and $\left(i,\left\{\mathrm{st}_{0}^{\mu}, \mathrm{St}_{1}^{\mu}\right\}_{\mu \in[M]}\right)$ as the internal state.

Round 2: Given (st ${ }_{C},\left\{Q^{\mu}\right\}_{\mu \in[M]}$ ) as input (where st ${ }_{C}=\left\{X_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}$ ), C.Dec ${ }_{1}$ does the following.

1. For every $\mu \in[M]$, let $\tilde{X}^{\mu ; \mu}: D(X) \rightarrow \boldsymbol{F}$ be an arbitrary polynomial such that $\tilde{X}^{\mu, \mu}\left(z_{\mu}, z_{\ell}, z_{\text {in }}, z\right)=X_{1, \text { in }}^{\mu}(z)$ for every $\boldsymbol{z} \in \boldsymbol{F}^{m_{\mathrm{to}}}$ (recall that $\boldsymbol{z}_{\mu} \in \boldsymbol{H}, \boldsymbol{z}_{\ell} \in \boldsymbol{H}^{m_{\mathrm{round}}}$, and $z_{\text {in }} \in \boldsymbol{H}^{m_{\text {Aug }}-m_{\mathrm{lo}}}$ are the points such that $D\left(X_{\ell, \text { in }}^{\mu}\right)=\left\{\left(\boldsymbol{z}_{\mu}, \boldsymbol{z}_{\ell}, z_{\text {in }}, \boldsymbol{z}\right) \mid \boldsymbol{z} \in\right.$ $\left.\boldsymbol{F}^{m_{10}}\right\}$; see Section D.2.2).
2. Output $\left\{\left.\tilde{X}^{\mu: \mu}\right|_{Q^{\mu}}\right\}_{\mu \in[M]}$ as the decommitment.

Verification: Given $\left(\mathrm{st}_{R}, \operatorname{com},\left\{\tilde{X}^{*}{ }^{\mu \cdot \mu}\right\}_{\mu \in[M]}\right)$ as input (where $\mathrm{st}_{R}=\left(i,\left\{\mathrm{st}_{0}^{\mu}, \mathrm{st}_{1}^{\mu}\right\}_{\mu \in[M]}\right)$ ), R.Dec. $\mathrm{D}_{1}$ does the following.

1. (Low-degree Test) For $\forall \mu \in[M]$, check that LD-Test. $\mathrm{D}_{m_{\mathrm{o}}|\boldsymbol{H}|, 3 \zeta}\left(\mathrm{st}_{0}^{\mu}, \tilde{X}^{\mu}{ }^{\mu: \mu}\right)=1$.
2. (Self-correction) For $\forall \mu \in[M]$, run $A^{\mu}:=\operatorname{SelfCorr} \operatorname{Rec}_{m_{\mathrm{r} 0}|\boldsymbol{H}|}\left(\mathrm{st}_{1}^{\mu}, \tilde{X}^{\mu}{ }^{\mu / \mu}\right)$, and check that $\tilde{x}_{i}:=A^{\mu}(\mu, 1, i)$ is a binary value (i.e., $\tilde{x}_{i}=0$ or $\tilde{x}_{i}=1$ ).
3. Output $\tilde{x}_{i}:=\tilde{x}_{i}^{1} \oplus \cdots \oplus \tilde{x}_{i}^{M}$ as the decommitted value.

Intuition of the proof. At first sight, proving the binding property against well-behaving adversaries seems to be trivial. Specifically, since it is guaranteed that the adversary does not give different answers to the same query during the binding security experiment, it seems to be implied that the adversary cannot open a commitment to two different values.

A problem is that to open a commitment to two different values, the adversary does not need to give different answers to the same query. This is because the adversary only needs to let the receiver recover different values from SelfCorr, and SelfCorr can make different queries on the same input depending on the randomness.

We overcome this problem by relying on the other assumptions, namely that the adversary is CNS and that it passes the low-degree test (Equation (E.1)). First, we use the well-behaving assumption to argue that if a well-behaving CNS adversary can let the receiver recover two different values from SelfCorr in the real binding experiment (where two sets of queries for SelfCorr are included in two sets of decommitment queries), it can do so even in a hybrid binding experiment where two sets of queries for SelfCorr are included in a single set of decommitment queries. Next, we observe that from a result that is implicitly shown in [KRR14], it follows that if a CNS adversary passes the low-degree test, the adversary cannot let the receiver recover two different values from SelfCorr in the hybrid binding experiment.

Formal proof. We prove Lemma E. 1 by showing the following stronger lemma (which we will reuse later in the proof of soundness (Lemma E.3)).

Lemma E.2. There exists a polynomial $\kappa_{\mathrm{dec}}$ such that the following holds. Let $\epsilon$ be any negligible function and $C_{1}^{*}=\left(\mathrm{C} . \mathrm{Com}_{1}^{*}, \mathrm{C} . \mathrm{Dec}_{1}^{*}\right)$ be any well-behaving $\kappa_{\mathrm{dec}}-$ CNS cheating committer-decommitter against $\left\langle C_{1}, R_{1}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following binding condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow$ $\left\langle\right.$ C.Com $_{1}^{*}$, R.Com $\left.{ }_{1}\right\rangle$.

- Binding Condition: If it holds
for $d:=m_{I O}|\boldsymbol{H}|$ and $D_{b}^{\mu}:=D\left(X_{1, \mathrm{in}}^{\mu}\right)$, then for every $i \in\left[N_{I O}\right]$ it holds $\operatorname{Pr}\left[b_{B A D}=1\right] \leq \operatorname{neg}(\lambda)$ in the following probabilistic experiment.

1. For $\forall b \in\{0,1\}$, sample $\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}$ as follows.

## Algorithm 6 Low－degree Test Procedure LD－Test ${ }_{d, D, \zeta}^{\mathcal{Y}}$

1． $\operatorname{Run}(Q, \mathrm{st}) \leftarrow$ LD－Test． $\mathrm{Q}_{D}$ ．
2．Run $($ out,$A) \leftarrow \mathcal{A}(Q)$ ．
3．Output $b:=\mathrm{LD}$－Test． $\mathrm{D}_{d, \zeta}(\mathrm{st}, A)$ ．

## Subroutine LD－Test．$Q_{D}$ ：

1．Choose $\lambda$ random lines $L_{1}, \ldots, L_{\lambda}: \boldsymbol{F} \rightarrow D$ ．
2．Output $\left(Q\right.$ ，st），where $Q=\left\{L_{j}(t)\right\}_{j \in[\lambda], t \in F}$ and st $:=\left\{L_{j}\right\}_{j \in[\lambda]}$ ．
Subroutine LD－Test． $\mathrm{D}_{d, \zeta}(\mathrm{st}, A)$ ：
1．Output 1 if and only if

$$
\left|\left\{j \in[\lambda] \mid \operatorname{isLD}_{d}\left(\left\{A\left(L_{j}(t)\right)\right\}_{t \in \boldsymbol{F}}\right)=1\right\}\right| \geq \lambda-\zeta .
$$

## Subroutine isLD ${ }_{d}\left(\left\{z_{i}\right\}_{i \in \boldsymbol{F}}\right)$

1．Output 1 if the function $f: i \mapsto z_{i}$ can be expressed as a degree－$d$ polynomial，and output $\perp$ in any other cases．

```
Algorithm 7 Parallel Low-degree Test Procedure P-LD-Test \({ }_{d,\left\{D_{i}\right]_{i \in[\mid] \mid, \zeta}^{\mathcal{A}}}\)
```

1．Run $\left(Q_{i}, \mathrm{st}_{i}\right) \leftarrow$ LD－Test．$Q_{D_{i}}$ for every $i \in[K]$ ．
2．Run（out，$\left.\left\{A_{i}\right\}_{i \in[K]}\right) \leftarrow \mathcal{A}\left(\left\{Q_{i}\right\}_{i \in[K]}\right)$ ．
3．Output 1 if and only if LD－Test． $\mathrm{D}_{d, \zeta}\left(\mathrm{st}_{i}, A_{i}\right)=1$ for $\forall i \in[K]$ ．
（a）Run $\left(Q_{b, 0}^{\mu}, \mathrm{st}_{b, 0}^{\mu}\right) \leftarrow$ LD－Test． $\mathrm{Q}_{D\left(X_{1, \mathrm{i})}^{\mu}\right)}$ for $\forall \mu \in[M]$ ．
（b）Run $\left(Q_{b, 1}^{\mu}, \mathrm{st}_{b, 1}^{\mu}\right) \leftarrow$ SelfCorr． $\mathrm{Q}_{D\left(X_{1, \mathrm{i}}^{\mu}\right)}(\{(\mu, 1, i)\})$ for $\forall \mu \in[M]$ ．
（c）Let $Q_{b}^{\mu}:=Q_{b, 0}^{\mu} \cup Q_{b, 1}^{\mu}$ ．
2． $\operatorname{Run}\left\{\tilde{X}^{*}{ }_{b}^{\mu \cdot \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \operatorname{C.Dec}_{1}^{*}\left(\operatorname{st}_{C},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$ ．
3．Let $b_{B A D}:=1$ if and only if both of the following hold．
－LD－Test． $\mathrm{D}_{m_{⿰ ㇒ ⿻ 二 丨 冂 刂 ~}^{\prime \prime}} \boldsymbol{H} \mid, 3 \zeta\left(\mathrm{St}_{b, 0}^{\mu}, \tilde{X}^{*}{ }_{b}^{\mu \cdot \mu}\right)=1$ for $\forall b \in\{0,1\}, \mu \in[M]$ ．

To see that Lemma E． 2 indeed implies Lemma E．1，observe that if $b_{\text {BAD }}=1$ holds in the experiment in Lemma E．1， $b_{\text {BAD }}=1$ holds in the experiment in Lemma E． 2 due to the definitions of R．Dec．Q $Q_{1}$ and R．Dec．D．$D_{1}$ ．

Proof of Lemma E．2．We let $\kappa_{\mathrm{dec}}$ be the polynomial $\kappa_{1}$ that is given in Section K．1．Fix any $\epsilon$ and $C_{1}^{*}$ ，and assume for con－ tradiction that for infinitely many $\lambda$ ，with non－negligible probability over the choice of（st ${ }_{C}$ ，com）$\leftarrow\left\langle\right.$ C．Com $_{1}^{*}$, R．Com $\left.{ }_{1}\right\rangle$ ， Equation（E．2）holds but $\operatorname{Pr}\left[b_{\text {BAD }}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ ．Let $\Lambda \subseteq \mathbb{N}$ be the set of such $\lambda$ ．In what follows，we consider a sequence of claims to obtain a contradiction with Lemma K． 2 （Consistency of SelfCorr）in Section K．1，which roughly says that if a CNS adversary passes the low－degree test，it cannot let two different values recovered from SelfCorr when two sets of queries are queried together．

Claim E．1．Let $\mathcal{A}_{1}$ be the following algorithm．
－On input（aux，$\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}$ ）：
1． $\operatorname{Run}\left\{\tilde{X}^{*}{ }_{b}^{\mu \cdot \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \operatorname{C.Dec}{ }_{1}^{*}\left(\operatorname{aux},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$ ．
2．Output $\left\{\tilde{X}^{*}{ }_{b}^{\mu: \mu}\right\}_{b \in\{0,1\}, \mu \in[M]}$ ．
Then，for $\forall \lambda \in \Lambda$ ，there exists aux $\in\{0,1\}^{\mathrm{poly}(\lambda)}$ such that all of the following hold，where we let $d:=m_{I o}|\boldsymbol{H}|$ and $D^{\mu}:=D\left(X_{1, \mathrm{in}}^{\mu}\right)$ in the following．

1. For $\forall\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}$ such that $Q_{b}^{\mu} \subseteq D^{\mu}$, it holds

$$
\operatorname{Pr}\left[\left.\begin{array}{l|l}
\exists \mu \in[M], \boldsymbol{q} \in Q_{0}^{\mu} \cap Q_{1}^{\mu}  \tag{E.3}\\
\text { s.t. } A_{0}^{\mu}(\boldsymbol{q}) \neq \perp \\
& \wedge A_{1}^{\mu}(\boldsymbol{q}) \neq \perp \\
& \wedge A_{0}^{\mu}(\boldsymbol{q}) \neq A_{1}^{\mu}(\boldsymbol{q})
\end{array} \right\rvert\,\left\{A_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \mathcal{A}_{1}\left(\text { aux, }\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)\right] \leq \operatorname{negl}(\lambda) .
$$

2. It holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\forall b \in\{0,1\}, \mu \in[M], & \left(Q_{b}^{\mu}, \mathrm{st}_{b}^{\mu}\right) \leftarrow \text { LD-Test. } Q_{D^{\mu}} \text { for } \forall b \in\{0,1\}, \mu \in[M]  \tag{E.4}\\
\text { LD-Test.D }{ }_{d, 3 \zeta}\left(\mathrm{st}_{b}^{\mu}, A_{b}^{\mu}\right)=1 & \left\{A_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \mathcal{A}_{1}\left(\operatorname{aux},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)
\end{array}\right] \geq 1-\epsilon(\lambda) .
$$

3. There exist $z^{1} \in D^{1}, \ldots, z^{M} \in D^{M}$ such that it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\exists \mu \in[M] \text { s.t. } & \left(Q_{b}^{\mu}, \text { st }{ }_{b}^{\mu}\right) \leftarrow \text { SelfCorr. } Q_{D^{\mu}}\left(\left\{z^{\mu}\right\}\right) \text { for } \forall b \in\{0,1\}, \mu \in[M]  \tag{E.5}\\
\tilde{A}_{0}^{\mu}\left(z^{\mu}\right) \neq \tilde{A}_{1}^{\mu}\left(z^{\mu}\right) & \left\{\tilde{A}_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M] \leftarrow \mathcal{A}_{1}\left(\operatorname{aux}_{b}^{\mu},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)}:=\operatorname{SelfCorr}^{\left(\operatorname{Rec}_{d}\left(\mathrm{st}_{b}^{\mu}, A_{b}^{\mu}\right) \text { for } \forall b \in\{0,1\}, \mu \in[M]\right.}
\end{array}\right] \geq 1 / \operatorname{poly}(\lambda) .
$$

Proof. Fix any $\lambda \in \Lambda$. When aux $:=\mathrm{st}_{C}$ is chosen randomly as $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}\right.$. Com $_{1}^{*}$, R.Com $\left.{ }_{1}\right\rangle$, the above three equations hold with non-negligible probability since $C^{*}$ is well-behaving and CNS and it breaks the binding condition. Hence, from an average argument, there exists aux such that the above three equations hold.

Claim E.2. Let $\mathcal{A}_{1}$ be the algorithm that is defined in Claim E.1, and fix any $\lambda$, aux, and $z^{1}, \ldots, z^{M}$ on which Equations (E.3), (E.4) and (E.5) hold. Then, it holds

$$
\operatorname{Pr}\left[\begin{array}{c|l}
\exists \mu \in[M] \text { s.t. } & \left(Q_{b}^{\mu}, \text { st }_{b}^{\mu}\right) \leftarrow \text { SelfCorr. }_{D^{\mu}}\left(\left\{z^{\mu}\right\}\right) \text { for } \forall b \in\{0,1\}, \mu \in[M]  \tag{E.6}\\
\left\{A_{b}^{\mu}\right\}_{b \in[0,1\}, \mu \in[M]} \leftarrow \mathcal{A}_{1}\left(\operatorname{aux},\left\{S_{b}^{\mu}\right\}_{b \in\{0,1], \mu \in[M]}\right), \\
\text { where } S_{0}^{\mu}:=Q_{0}^{\mu} \cup Q_{1}^{\mu} \text { and } S_{1}^{\mu}:=Q_{1}^{\mu} \\
\tilde{A}_{0}^{\mu}\left(z^{\mu}\right) \neq \tilde{B}_{1}^{\mu}\left(z^{\mu}\right) & \tilde{A}_{b}^{\mu}:=\operatorname{SelfCorr.Rec}_{d}\left(\mathrm{st}_{b}^{\mu}, A_{b}^{\mu}\right) \text { for } \forall b \in\{0,1\}, \mu \in[M] \\
& \tilde{B}_{1}^{\mu}:=\operatorname{SelfCorrr}^{\mu} \operatorname{Rec}_{d}\left(\mathrm{st}_{1}^{\mu}, A_{0}^{\mu}\right) \text { for } \forall \mu \in[M]
\end{array}\right] \geq 1 / \text { poly }(\lambda) .
$$

Proof. Since Equation (E.5) holds, the CNS of $\mathcal{A}_{1}$ (which is inherited from the CNS of C.Dec ${ }_{1}^{*}$ ) implies that we can obtain Equation (E.6) by showing

Hence, we focus on showing Equation (E.7).
First, we remark that from the construction of SelfCorr, the two values $\tilde{A}_{1}^{\mu}\left(z^{\mu}\right)$ and $\tilde{B}_{1}^{\mu}\left(z^{\mu}\right)$ in Equation (E.7) are sampled in the following manner.

1. For $\forall b \in\{0,1\}, \mu \in[M]$, choose $\lambda$ random lines $L_{b, 1}^{\mu}, \ldots, L_{b, \lambda}^{\mu}: \boldsymbol{F} \rightarrow D^{\mu}$ such that each $L \in\left\{L_{b, 1}^{\mu}, \ldots, L_{b, \lambda}^{\mu}\right\}$ satisfies $L(0)=z^{\mu}$. Let $Q_{b}^{\mu}:=\left\{L_{b, j}^{\mu}(t)\right\}_{j \in[\lambda], t \in \boldsymbol{F} \backslash\{0\}}$.
2. $\operatorname{Run}\left\{A_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \mathcal{A}_{1}\left(\right.$ aux, $\left.\left\{S_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$, where $S_{0}^{\mu}:=Q_{0}^{\mu} \cup Q_{1}^{\mu}$ and $S_{1}^{\mu}:=Q_{1}^{\mu}$.
3. For $\forall \mu \in[M]$, check that there exists $c_{1}^{\mu} \in \boldsymbol{F}$ such that

$$
\left|\left\{j \in[\lambda] \mid \operatorname{Recon}_{d}\left(\left\{A_{1}^{\mu}\left(L_{1, j}^{\mu}(t)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)=c_{1}^{\mu}\right\}\right| \geq 0.9 \lambda .
$$

Let $\tilde{A}_{1}^{\mu}\left(z^{\mu}\right):=c_{1}^{\mu}$ if such $c_{1}^{\mu}$ exists, and let $\tilde{A}_{1}^{\mu}\left(z^{\mu}\right):=\perp$ otherwise.
4. For $\forall \mu \in[M]$, check that there exists $d_{1}^{\mu} \in \boldsymbol{F}$ such that

$$
\left|\left\{j \in[\lambda] \mid \operatorname{Recon}_{d}\left(\left\{A_{0}^{\mu}\left(L_{1, j}^{\mu}(t)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)=d_{1}^{\mu}\right\}\right| \geq 0.9 \lambda .
$$

Let $\tilde{B}_{1}^{\mu}\left(z^{\mu}\right):=d_{1}^{\mu}$ if such $d_{1}^{\mu}$ exists, and let $\tilde{B}_{1}^{\mu}\left(z^{\mu}\right):=\perp$ otherwise.

In what follows, we always consider probability over this sampling.
Note that due to Equation (E.4) and Lemma K. 1 (Correctness of SelfCorr) in Section K. 1 (which roughly says that SelfCorr outputs $\perp$ only with negligible probability for any adversary that passes the low-degree test), we have $\tilde{A}_{1}^{\mu}\left(z^{\mu}\right) \neq \perp$ and $\tilde{B}_{1}^{\mu}\left(z^{\mu}\right) \neq \perp$ for $\forall \mu \in[M]$ except with negligible probability. From the construction of Recon, this implies that except with negligible probability, we have

$$
\left|\left\{j \in[\lambda] \mid \perp \notin\left\{A_{1}^{\mu}\left(L_{1, j}^{\mu}(t)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right\}\right| \geq 0.9 \lambda
$$

and

$$
\left|\left\{j \in[\lambda] \mid \perp \notin\left\{A_{0}^{\mu}\left(L_{1, j}^{\mu}(t)\right)\right\}_{t \in F \backslash\{0\}}\right\}\right| \geq 0.9 \lambda
$$

for $\forall \mu \in[M]$. Combined with Equation (E.3), these two imply that except with negligible probability, we have

$$
\begin{equation*}
\mid\left\{j \in[\lambda] \mid A_{0}^{\mu}\left(L_{1, j}^{\mu}(t)\right)=A_{1}^{\mu}\left(L_{1, j}^{\mu}(t)\right) \neq \perp \text { for } \forall t \in \boldsymbol{F} \backslash\{0\}\right\} \mid \geq 0.8 \lambda \tag{E.8}
\end{equation*}
$$

for $\forall \mu \in[M]$. Thus, from a union bound, we have $\tilde{A}_{1}^{\mu}\left(z^{\mu}\right) \neq \perp, \tilde{B}_{1}^{\mu}\left(z^{\mu}\right) \neq \perp$, and Equation (E.8) for $\forall \mu \in[M]$ except with negligible probability. From the definitions of $\tilde{A}_{1}^{\mu}\left(z^{\mu}\right), \tilde{B}_{1}^{\mu}\left(z^{\mu}\right)$, this implies that we have $\tilde{A}_{1}^{\mu}\left(z^{\mu}\right)=\tilde{B}_{1}^{\mu}\left(z^{\mu}\right)$ for $\forall \mu \in[M]$ except with negligible probability, as desired.

Claim E.3. Let $\mathcal{A}_{2}$ be the following algorithm.

- On input (aux, $\left\{Q^{\mu}\right\}_{\mu \in[M]}$ ):

1. Run $\left\{A_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \mathcal{A}_{1}\left(\right.$ aux, $\left.\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$, where $Q_{0}^{\mu}:=Q^{\mu}$ and $Q_{1}^{\mu}:=\emptyset$.
2. Output $\left\{A_{0}^{\mu}\right\}_{\mu \in[M]}$.

Then, for $\forall \lambda \in \Lambda$, there exist $\operatorname{aux} \in\{0,1\}^{\text {poly( } \lambda)}$ and $z^{1} \in D^{1}, \ldots, z^{M} \in D^{M}$ such that

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\forall \mu \in[M], & \left(Q^{\mu}, \mathrm{st}^{\mu}\right) \leftarrow \mathrm{LD}^{2}-\text { Test. } Q_{D^{\mu}} \text { for } \mu \in[M] \\
\text { LD-Test.D } \mathrm{D}_{d, 3 \zeta}\left(\mathrm{st}^{\mu}, A^{\mu}\right)=1 & \left\{A^{\mu}\right\}_{\mu \in[M]} \leftarrow \mathcal{A}_{2}\left(\operatorname{aux},\left\{Q^{\mu}\right\}_{\mu \in[M]}\right)
\end{array}\right] \geq 1-\operatorname{negl}(\lambda)
$$

and

$$
\operatorname{Pr}\left[\begin{array}{c|l}
\exists \mu \in[M] \text { s.t. } & \left(Q_{b}^{\mu}, \mathrm{st}_{b}^{\mu}\right) \leftarrow{\text { SelfCorr. } Q_{D^{\mu}}\left(\left\{z^{\mu}\right\}\right) \text { for } \forall b \in\{0,1\}, \mu \in[M]}^{\tilde{A}_{0}^{\mu}\left(z^{\mu}\right) \neq \tilde{A}_{1}^{\mu}\left(z^{\mu}\right)} \\
\left\{\tilde{A}^{\mu}\right\}_{\mu \in[M]}^{\mu} \leftarrow \mathcal{A}_{2}\left(\operatorname{aux},\left\{Q_{0}^{\mu} \cup Q_{1}^{\mu}\right\}_{\mu \in[M]}\right) \\
\tilde{A}_{b}:=\operatorname{SelfCorr}^{\left(R e c_{d}\left(\mathrm{st}_{b}^{\mu}, A^{\mu}\right) \text { for } \forall b \in\{0,1\}, \mu \in[M]\right.}
\end{array}\right] \geq 1 / \operatorname{poly}(\lambda),
$$

where $d:=m_{I o}|\boldsymbol{H}|$ and $D^{\mu}:=D\left(X_{1, \mathrm{in}}^{\mu}\right)$.
Proof. This claim follows from Claim E.1, Claim E.2, and the assumption that $C_{1}^{*}$ is CNS (which implies that $\mathcal{A}_{1}$ is CNS), since the latter implies that the output of $\mathcal{A}_{2}\left(\right.$ aux, $\left.\left\{Q_{0}^{\mu} \cup Q_{1}^{\mu}\right\}_{\mu \in[M]}\right)$ is computationally indistinguishable from that of $\mathcal{A}_{1}\left(\right.$ aux, $\left.\left\{S_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$, where $S_{0}^{\mu}:=Q_{0}^{\mu} \cup Q_{1}^{\mu}$ and $S_{1}^{\mu}:=Q_{1}^{\mu}$.
Clearly, Claim E. 3 contradicts with (the straightforward parallel version of) Lemma K. 2 (Consistency of SelfCorr). This concludes the proof of Lemma E.2.

## E. 3 Proof of Soundness

We prove the soundness against cheating adversaries that are no-signaling and well-behaving in the following sense.
Definition 19 (No-signaling committer-prover). A cheating committer-prover $C_{1}^{*}=\left(\mathrm{C}_{1} \mathrm{Com}_{1}^{*}, \mathrm{C} . \operatorname{Prv} \mathrm{v}_{1}^{*}\right)$ against $\left\langle C_{1}, R_{1}\right\rangle$ is $\kappa_{\max }-$ CNS if $\mathrm{C}^{2} \mathrm{Prv}_{1}^{*}\left(\mathrm{st}_{C}, \cdot\right)$ is $\kappa_{\max }-C N S$ for $\forall\left(\mathrm{st}_{C}\right.$, com $) \leftarrow\left\langle\right.$ C.Com $_{1}^{*}$, R.Com $\left.{ }_{1}\right\rangle$.

Definition 20 (Well-behaving committer-prover). A cheating committer-prover $C_{1}^{*}=\left(\mathrm{C} . \mathrm{Com}_{1}^{*}, \mathrm{C} . \operatorname{Prv}{ }_{1}^{*}\right)$ against $\left\langle C_{1}, R_{1}\right\rangle$ is well-behaving if both of the following hold.

1. Consistency on $D\left(X_{1, \mathrm{i}}^{\mu}\right)$ : For $\forall \lambda \in \mathbb{N}, \forall\left\{Q_{0}^{\mu: \nu}\right\}_{\mu, v \in[M]},\left\{Q_{1}^{\mu: v}\right\}_{\mu, v \in[M]} \subseteq(D(X))^{M^{2}}, \forall \alpha, \beta, \gamma, \delta \in[M]$ such that $\exists \xi \in$ $\{\alpha, \beta\} \cap\{\gamma, \delta\}$, and $\forall \boldsymbol{q} \in Q_{0}^{\alpha ; \beta} \cap Q_{1}^{\gamma: \delta} \cap D\left(X_{1, \mathrm{in}}^{\xi}\right)$, it holds
2. Consistency on $D\left(X^{\mu}\right)$ : For $\forall \lambda \in \mathbb{N}, \forall\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]} \subseteq(D(X))^{M^{2}}, \forall \alpha, \beta, \gamma, \delta \in[M]$ such that $\exists \xi \in\{\alpha, \beta\} \cap\{\gamma, \delta\}$, and $\forall \boldsymbol{q} \in Q^{\alpha ; \beta} \cap Q^{\gamma: \delta} \cap D\left(X^{\xi}\right)$, it holds

Remark 5 (Intuition of well-behaving adversaries). Roughly speaking, the above definition of well-behaving adversaries guarantees the following. Recall that in $\left\langle C_{1}, R_{1}\right\rangle$, the ( $\mu, v$ )-th instance of (PCP.P, PCP.V) is used for proving consistency of a pair of views ( $\mathrm{view}^{\mu}$, view $^{\nu}$ ), and the receiver expects that it can query to an LDE $X^{\mu: \nu}$ of a satisfying assignment to the variables in $\varphi_{f}^{\mu: \nu}$. Also, recall that the receiver expects that through $X^{\mu: \nu}$, it can query to an LDE $X_{1, \text { in }}^{\mu}$ of an assignment to the variables indexed by $I_{1, \text { in }}^{\mu}$, and as remarked in Remark 3, the assignment to these variables should be the initial state of $P^{\mu}$ (which is committed in the commit phase). Also, recall that the receiver expects that through $X^{\mu: \nu}$, it can query to an LDE $X^{\mu}$ of an assignment to the variables indexed by $I^{\mu}$, and as remarked in Remark 3, the assignment to these variables should depend only on view ${ }^{\mu}$, meaning that the same values should be assigned to these variables in the ( $\mu, v$ )-th instance of (PCP.P, PCP.V) and in the ( $\mu, \xi$ )-th one. Now, the first condition of well-behaving adversaries guarantees that once the commit phase is completed, the adversary does not give different answers to the queries to $X_{1, \text { in }}^{\mu}$ in different invocations. Regarding the second condition, it guarantees that once the commit phase is completed, the adversary does not give different answers to the queries to $X^{\mu}$ in different instances of (PCP.P, PCP.V) in a single invocation.

The formal statement of the soundness of $\left\langle C_{1}, R_{1}\right\rangle$ is given below.
Lemma E.3. There exists a polynomial $\kappa_{\text {prv }}$ such that the following holds. Let $\epsilon_{\text {SND }}$ be any negligible function, $E_{1}$ be the extractor in Algorithm 8, and $C_{1}^{*}=\left(\mathrm{C} . \mathrm{Com}_{1}^{*}, \mathrm{C} . \mathrm{Prv}_{1}^{*}\right)$ be any well-behaving $\kappa_{\mathrm{prv}}-$ CNS cheating committer-prover against $\left\langle C_{1}, R_{1}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following soundness condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}\right.$. Com $_{1}^{*}$, R.Com $\left.{ }_{1}\right\rangle$.

- Soundness Condition: If it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{1} ;\left(f, \pi^{*}\right) \leftarrow \operatorname{C.Prv}_{1}^{*}\left(\mathrm{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot D_{1}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array} \tag{E.9}
\end{array}\right] \geq 1-\epsilon_{\text {SND }}(\lambda),
$$

then there exists $x_{\text {Cом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that

$$
\begin{equation*}
\forall i \in[n], \operatorname{Pr}\left[x_{i}=x_{i}^{*} \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{1}^{\mathrm{C} \cdot \operatorname{Prv}_{1}^{*}\left(\mathrm{st}_{C},\right)}(\operatorname{com}, i), \operatorname{R.Dec}_{1}(\operatorname{com}, i)\right\rangle\right] \geq 1-\operatorname{negl}(\lambda) \tag{E.10}
\end{equation*}
$$

and

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \mathrm{Q}_{1} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{1}^{*}\left(\mathrm{st}_{C}, Q\right)  \tag{E.11}\\
\wedge f\left(x_{\text {CoM }}^{*}\right)=0 & b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{1}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \leq \operatorname{negl}(\lambda)
$$

$$
\frac{\text { Algorithm } \left.8 \text { Extractor } E_{1} \text { (against }\left\langle C_{1}, R_{1}\right\rangle\right)}{\text { Input: com, } i \text {, and }\left\{Q^{\mu}\right\}_{\mu \in[M]} \subset D\left(X_{1, \text { in }}^{1}\right) \times \cdots \times D\left(X_{1, \text { in }}^{M}\right)}
$$

1. $\operatorname{Run}\left(f,\left\{\pi^{* \mu: \nu}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C.} \operatorname{Prv}_{1}^{*}\left(\operatorname{st}_{C},\left\{{\left.\left.Q^{\mu: v}\right\}_{\mu, v \in[M]}\right) \text {, where each } Q^{\mu: v} \text { is defined as } Q^{\mu: v}:=Q^{\mu} \text { if } \mu=v \text { and } Q^{\mu: v}:=\emptyset}_{\square}\right.\right.$ otherwise.
2. Output $\left\{\tilde{X}^{\mu}{ }^{\mu / \mu}\right\}_{\mu \in[M]}$ as the decommitment, where $\tilde{X}^{\mu}: \mu:=\pi^{* \mu ; \mu}$.

We directly go to the formal proof. An overview of the proof is given in Section 4.
Proof. We let $\kappa_{\mathrm{prv}}=\kappa_{0} \cdot \kappa_{\max }$, where $\kappa_{0}, \kappa_{\max }$ are the polynomials that are given in Lemma D.4. Fix any $\epsilon_{\mathrm{SND}}$ and $C_{1}^{*}=$ (C.Com ${ }_{1}^{*}$, C.Prví) such that for infinitely many $\lambda \in \mathbb{N}$, Equation (E.9) holds with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$ (if no such $\epsilon_{\text {sND }}$ and $C_{1}^{*}$ exist, the lemma holds trivially), and let $\Lambda$ be the set of such $\lambda$.

## Step 0: showing consistency properties on self-corrected C.Prv*.

First, we make preliminary observations on SelfCorr. Specifically, we observe that the consistency conditions of wellbehaving adversaries (Definition 20) hold even on the outputs of SelfCorr.
Claim E. 4 (Consistency on $D\left(X_{1, \text { in }}^{\mu}\right)$ ). For every $\lambda \in \Lambda$, the following holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$.

- If Equation (E.9) holds, then there exists $\left\{\left(x_{1}^{\mu}, \ldots, x_{N_{i o}}^{\mu}\right)\right\}_{\mu \in[M]} \in\left(\{0,1\}^{N_{i o}}\right)^{M}$ such that for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{1, \mathrm{n}}^{\xi}$,

Proof. Fix any $\lambda \in \Lambda$.
First, we observe that, by using the argument in the proof of Lemma E.2, we can show that with overwhelming probability over the choice of $\left(\right.$ st $_{C}$, com $) \leftarrow\left\langle\right.$ C.Com $_{1}^{*}$, R.Com $\left.{ }_{1}\right\rangle$, if Equation (E.9) holds, then there exists $\left\{\left(x_{1}^{\mu}, \ldots, x_{N_{\mathrm{Io}}}^{\mu}\right)\right\}_{\mu \in[M]} \in$ $\left(\boldsymbol{F}^{N_{\mathrm{oo}}}\right)^{M}$ such that for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in \mathcal{I}_{1, \text { in }}^{\xi}$, we have Equation (E.12). To see this, first observe that since the proof verification includes the low-degree test for $D\left(X_{1, \mathrm{i}}^{\xi}\right)$, Equation (E.9) implies $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$,

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b_{\mathrm{LD}}=1 & \begin{array}{l}
(Q, \mathrm{st}) \leftarrow \text { LD-Test. } Q_{D\left(X_{\mathrm{in}}^{\xi}\right)} \\
\left(f,\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \mathrm{C} . \operatorname{Prv}\left(\mathrm{st}_{C},\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right), \\
\text { where } Q^{\mu: v}:=\emptyset \text { for } \forall(\mu, v) \neq(\alpha, \beta) \text { and } Q^{\alpha: \beta}:=Q \\
b_{\mathrm{LD}}:=\text { LD-Test.D }
\end{array}
\end{array}\right] \geq 1-\operatorname{negl}(\lambda) .
$$

From this observation, it follows that given Equation (E.9), we can argue as in Lemma E. 2 to show that for $\forall \alpha, \beta \in[M]$, $\forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{1, \text { in }}^{\xi}$, the value of $A^{\alpha ; \beta}\left(i^{*}\right)$ is unique when computed as in Equation (E.12).

Second, since the prove phase of $\left\langle C_{1}, R_{1}\right\rangle$ is just parallel executions of (PCP.P, PCP.V), the everywhere local consistency of SelfCorr (Claim D.4) and Lemma K. 3 in Section K. 1 (which guarantees that the values recovered from
 (E.9) holds, then for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{1, \text { in }}^{\xi}$, we have $\operatorname{Pr}\left[A^{\alpha ; \beta}\left(i^{*}\right)=0 \vee A^{\alpha ; \beta}\left(i^{*}\right)=1\right] \geq 1-\operatorname{negl}(\lambda)$, where the probability is taken as in Equation (E.12). ${ }^{32}$

By combining the above two, we obtain the claim.
Claim E. 5 (Consistency on $D\left(X^{\mu}\right)$ ). For every $\lambda \in \Lambda$, the following holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$.

- If Equation (E.9) holds, then $\forall \alpha, \beta, \gamma, \delta \in[M]$ such that $\exists \xi \in\{\alpha, \beta\} \cap\{\gamma, \delta\}$, and $\forall i^{*} \in I^{\xi}$,

Proof. This claim can be proven similarly to Lemma E.2.
Next, we introduce a definition that we use in the rest of the proof.
Definition 21. For any $\lambda \in \Lambda$, we say that $\left(\mathrm{st}_{C}, \mathrm{com}\right)$ is good if under the condition that $\left(\mathrm{st}_{C}, \mathrm{com}\right)$ is output by $\left\langle\mathrm{C} . \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$, the following hold.

- Equation (E.9) holds.
- There exists $\left\{\left(x_{1}^{\mu}, \ldots, x_{N_{t o}}^{\mu}\right)\right\}_{\mu \in[M]} \in\left(\{0,1\}^{N_{\text {Io }}}\right)^{M}$ such that Equation (E.12) holds for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{1, \mathrm{n}}^{\xi}$.
- Equation (E.13) holds for $\forall \alpha, \beta, \gamma, \delta \in[M]$ such that $\exists \xi \in\{\alpha, \beta\} \cap\{\gamma, \delta\}$, and $\forall i^{*} \in I^{\xi}$.

Furthermore, for any $\lambda \in \Lambda$ and any good (st ${ }_{C}$, com), we say that $\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}=\left\{\left(x_{1}^{\mu}, \ldots, x_{N_{\text {Io }}}^{\mu}\right)\right\}_{\mu \in[M]} \in\left(\{0,1\}^{N_{i o}}\right)^{M}$ is the (unique) good MPC initial state if Equation (E.12) holds on $\left\{\left(x_{1}^{\mu}, \ldots, x_{N_{t o}}^{\mu}\right\}_{\mu \in[M]}\right.$ for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{1, \mathrm{n}}^{\xi}$.

From Claim E. 4 and Claim E.5, it follows that for proving Lemma E.3, it suffices to show that for any good (st ${ }_{C}$, com), there exists $x_{\text {Com }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that Equations (E.10) and (E.11) hold.

[^19]
## Step 1: showing that $E_{1}$ succeeds with overwhelming probability.

Claim E.6. For every $\lambda \in \Lambda$ and good (st ${ }_{C}$, com), Equation (E.10) holds for $x_{c o m}^{*}=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ that is defined as follows: let $\left\{\left(\mathrm{st}_{0}^{\mu} \text {, } \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}=\left\{\left(x_{1}^{\mu}, \ldots, x_{N_{t o}}^{\mu}\right)\right\}_{\mu \in[M]} \in\left(\{0,1\}^{N_{t o}}\right)^{M}$ be the good MPC initial state; then, let $x_{i}^{*}:=$ $x_{i}^{1} \oplus \cdots \oplus x_{i}^{M}$ for $\forall i \in[n]$.
Proof. Fix any $\lambda \in \Lambda$ and good (st ${ }_{C}, \mathrm{com}$ ), and let $x_{\text {сом }}^{*}$ be defined as in the claim statement. First, from the constructions of R.Dec ${ }_{1}$ and $E_{1}$, the CNS of C.Prv ${ }_{1}^{*}$, and Equation (E.12), it follows that we have Equation (E.10) if we have

Second, from the construction of R.Prv ${ }_{1}$ and the CNS of C.Prv ${ }_{1}^{*}$, it follows that we indeed have Equation (E.14) due to Equation (E.9) (which is guaranteed to hold since (st $C_{C}, \mathrm{com}$ ) is good).

## Step 2: showing that C.Prv ${ }_{1}^{*}$ fails to prove false statement.

Claim E.7. For every $\lambda \in \Lambda$ and good (st ${ }_{C}$, com), Equation (E.11) holds for $x_{\text {com }}^{*}$ that is defined as in Claim E.6.
To prove this claim, we show the following claim.
Claim E.8. For every $\lambda \in \Lambda$ and good (st $\left.{ }_{C}, \mathrm{com}\right)$, we have

$$
\operatorname{Pr}\left[f\left(x_{\text {com }}^{*}\right)=1 \left\lvert\, \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \operatorname{R.Prv} \cdot \mathrm{Q}_{1}  \tag{E.15}\\
\left(f, \pi^{*}\right) \leftarrow \operatorname{C.Prv}^{*}\left(\operatorname{st}_{C}, Q\right)
\end{array}\right.\right] \geq 1-\operatorname{negl}(\lambda),
$$

where $x_{\text {Сом }}^{*} \in\{0,1\}^{n}$ is defined as in Claim E.6.
Note that Claim E. 8 implies Claim E. 7 since Equations (E.15) implies Equation (E.11).
Proof of Claim E.8. We show three subclaims. Let P-SelfCorr be a parallel version of SelfCorr; see Algorithm 9.

```
Algorithm 9 Parallel Self-Correction Procedure P-SelfCorrr \({ }_{d, D}^{\text {C.Prv }}{ }^{*}\left(\right.\) st \(\left._{C}, *\right)\)
Input: \(\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\).
    1. Run \(\left(\tilde{Q}^{\mu: v}, \mathrm{st}_{Q}^{\mu: \nu}\right) \leftarrow \operatorname{SelfCorr} \mathrm{Q}_{D}\left(Q^{\mu: \nu}\right)\) for each \(\mu, v \in[M]\).
    2. \(\operatorname{Run}\left(f,\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C.} \operatorname{Prv}_{1}^{*}\left(\operatorname{st}_{C},\left\{\tilde{Q}^{\mu: v}\right\}_{\mu, v \in[M]}\right)\).
    3. Run \(A^{\mu: v}:=\operatorname{SelfCorrr}^{\operatorname{Rec}} \operatorname{Rec}_{d}\left(\mathrm{St}_{Q}^{\mu: v}, \pi^{* \mu: v}\right)\) for each \(\mu, v \in[M]\).
    4. Output \(\left(f,\left\{A^{\mu: v}\right\}_{\mu, v \in[M]}\right)\).
```

Sub-Claim E.1. There exists a negligible function $\epsilon_{1}$ such that for every $\lambda \in \Lambda$ and every good ( $s t_{C}, \mathrm{com}$ ), we have
where the event $\operatorname{Correct}\left(\left\{S^{\mu}\right\}_{\mu \in[M]}\right)$ is defined as follows. Let the correct view $\left\{\operatorname{view}^{\mu}\right\}_{\mu \in[M]}$ be the views of the parties in the execution of $\Pi$ on $\left(f,\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}\right)$, where $\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}$ is the good MPC initial states that is determined by ( $\mathrm{st}_{C}, \mathrm{com}$ ). For $\forall \mu, v \in[M]$, let the correct assignment $A_{\text {corr }}^{\mu: v}$ (to the variables in $\varphi_{f}^{\mu: v}$ ) be the assignment that is obtained from $\left(\operatorname{view}^{\mu}\right.$, view $\left.^{\nu}\right)$ as in Remark 3. Then, $\operatorname{Correct}\left(\left\{S^{\mu}\right\}_{\mu \in[M]}\right)$ is the event that $A^{\mu: v}(s)=A_{\mathrm{corr}}^{\mu \cdot v}(s)$ holds for $\forall \mu, v \in[M]$, $\forall \xi \in\{\mu, v\}, \forall s \in S^{\xi}$.

Proof. To show this subclaim, it suffices to show that there exists a negligible function $\epsilon$ such that for every $\lambda \in \Lambda$ and good (st ${ }_{C}$, com), the following holds.

- For $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}, \forall i^{*} \in \mathcal{I}_{1, \text { in }}^{\xi}$, we have
(Indeed, given Equation (E.17), we can obtain Equation (E.16) for a negligible function $\epsilon_{1}$ by using Claim D.7, the CNS of C.Prv ${ }_{1}^{*}$, and a union bound.) Now, we conclude the proof by noticing that for every $\lambda \in \Lambda$ and every good (st ${ }_{C}$, com), the above follows from Lemma K. 3 in Section K. 1 (which guarantees that the values recovered from SelfCorr $m_{m_{10}|\boldsymbol{H}|, D\left(X_{1, \mathrm{in}}^{\xi}\right)}$ and SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}$ are equal) since $\left\{\left(\mathrm{st}_{0}^{\mu}, \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right)\right\}_{\mu \in[M]}$ is the good MPC initial states.

Sub-Claim E.2. There exists a negligible function $\epsilon_{2}$ such that for every $\lambda \in \Lambda$, every good ( $\mathrm{st}_{C}$, com), and every $\ell \in$ [ $N_{\text {round }}$ ], we have
where $\operatorname{Correct}\left(\left\{S^{\mu}\right\}_{\mu \in[M]}\right)$ and $\operatorname{Correct}\left(\left\{T^{\mu}\right\}_{\mu \in[M]}\right)$ are defined as in Sub-Claim E.1.
Proof. Since the prove phase of $\left\langle C_{1}, R_{1}\right\rangle$ is just parallel executions of (PCP.P, PCP.V), this subclaim follows from Claim D.5.

Sub-Claim E.3. There exists a negligible function $\epsilon_{3}$ such that for every $\lambda \in \Lambda$, every good ( $\mathrm{st}_{C}, \mathrm{com}$ ), and every $\ell \in$ $\left\{2, \ldots, N_{\text {round }}\right\}$, we have
where $\operatorname{Correct}\left(\left\{S^{\mu}\right\}_{\mu \in[M]}\right)$ and $\operatorname{Correct}\left(\left\{T^{\mu}\right\}_{\mu \in[M]}\right)$ are defined as in Sub-Claim E.1.
Proof. As in the proof of Sub-Claim E.1, it suffices to show that there exists a negligible function $\epsilon$ such that for every $\lambda \in \Lambda$, every good (st ${ }_{C}, \mathrm{com}$ ), and every $\ell \in\left\{2, \ldots, N_{\text {round }}\right\}$, the following holds.

- For $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}, \forall i^{*} \in I_{\ell, \text { in }}^{\xi}$, we have

Fix any $\lambda$, (st ${ }_{C}$, com), and $\ell$ as above, and for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}, \forall i^{*} \in \mathcal{I}_{\ell-1, \text { out }}^{\xi} \cup I_{\ell, \text { in }}^{\xi}$, let $p^{\alpha ; \beta}\left(i^{*}\right)$ be the following probability.

In this notation, our goal is to show that we have $p^{\alpha: \beta}\left(i^{*}\right) \leq \epsilon(\lambda)$ for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}, \forall i^{*} \in \mathcal{I}_{\ell, \text { in }}^{\xi}$.
First, we observe that we have $p^{\alpha ; \beta}\left(i^{*}\right) \leq \operatorname{negl}(\lambda)$ for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{\ell-1, \text { out }}^{\xi}$. Indeed, since the prove phase of $\left\langle C_{1}, R_{1}\right\rangle$ is just parallel executions of (PCP.P, PCP.V), this bound follows from Claim D.6.

Next, we observe that we have $p^{\alpha \cdot \beta}\left(i^{*}\right) \leq \operatorname{negl}(\lambda)$ for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{\ell, \text { in }}^{\xi}$ such that

- either $i^{*}=(\xi, \ell, i)$ for some $i \in\left[n_{\mathrm{st}]}\right.$, i.e., $i^{*}$ is the index of a variable in $\boldsymbol{w}_{\ell, \text { in }}^{\xi}$, which corresponds to $P^{\xi}$, sinternal state at the beginning of Round $\ell$,
- or $i^{*} \in\left\{\left(\alpha, \ell, n_{\mathrm{st}}+\beta\right),\left(\beta, \ell, n_{\mathrm{st}}+\alpha\right)\right\}$, i.e., $i^{*}$ is the index of the variable $\boldsymbol{w}_{\ell, \mathrm{in}}^{\alpha}\left(n_{\mathrm{st}}+\beta\right)$ or $\boldsymbol{w}_{\ell, \text { in }}^{\beta}\left(n_{\mathrm{st}}+\alpha\right)$, where the former corresponds to $P^{\alpha}$ s incoming message from $P^{\beta}$ in Round $\ell$ and the latter corresponds to $P^{\beta}$, s incoming message from $P^{\alpha}$ in Round $\ell$.

Fix any $\alpha, \beta, \xi$ as above.

Case 1. When $i^{*}=(\xi, \ell, i)$ for some $i \in\left[n_{\mathrm{st}}\right]$, let $j^{*} \in \mathcal{I}_{\ell-1, \text { out }}^{\xi}$ be the index of the variable $\boldsymbol{w}_{\ell-1, \text { out }}^{\xi}(i)$, i.e., the variable that corresponds to the $i$-th bit of $P^{\xi}$,s internal state at the end of Round $\ell-1$. Since P -SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}^{\mathrm{C} . \mathrm{Prl}_{1}^{*}(\mathrm{st}, \cdot)}$ is an adaptive local assignment generator (Claim D.4), we have

$$
\operatorname{Pr}\left[A^{\alpha ; \beta}\left(i^{*}\right) \neq A^{\alpha: \beta}\left(j^{*}\right) \left\lvert\, \begin{array}{l}
\left(f,\left\{A^{\mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \mathrm{P}-\text { SelfCorr }_{m|\boldsymbol{H}|, D(X)}^{\mathrm{C} \cdot \operatorname{Prv}_{*}^{*}(\operatorname{stc}, \cdot)}\left(\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right),  \tag{E.18}\\
\text { where } Q^{\mu: v}:=\emptyset \text { for } \forall(\mu, v) \neq(\alpha, \beta) \text { and } Q^{\alpha ; \beta}:=\left\{i^{*}, j^{*}\right\}
\end{array}\right.\right] \leq \operatorname{negl}(\lambda) .
$$

Hence, we have

$$
\begin{aligned}
& p^{\alpha ; \beta}\left(i^{*}\right)
\end{aligned}
$$

$$
\begin{aligned}
& \leq p^{\alpha: \beta}\left(j^{*}\right)+\operatorname{neg|}(\lambda),
\end{aligned}
$$

where in the last inequality, we use the fact that the event

$$
\left(A^{\alpha ; \beta}\left(i^{*}\right) \neq A_{\mathrm{corr}}^{\alpha ; \beta}\left(i^{*}\right)\right) \wedge\left(A^{\alpha ; \beta}\left(j^{*}\right)=A_{\mathrm{corr}}^{\alpha ; \beta}\left(j^{*}\right)\right)
$$

implies $A^{\alpha ; \beta}\left(i^{*}\right) \neq A^{\alpha ; \beta}\left(j^{*}\right)$ since we have $A_{\text {corr }}^{\alpha ; \beta}\left(i^{*}\right)=A_{\text {corr }}^{\alpha ; \beta}\left(j^{*}\right)$ from the definition of $A_{\text {corr }}$. Now, since we have $p^{\alpha ; \beta}\left(j^{*}\right) \leq \operatorname{negl}(\lambda)$ from what is shown in the previous paragraph, we obtain $p^{\alpha ; \beta}\left(i^{*}\right) \leq \operatorname{negl}(\lambda)$ as desired.

Case 2. When $i^{*} \in\left\{\left(\alpha, \ell, n_{\mathrm{st}}+\beta\right),\left(\beta, \ell, n_{\mathrm{st}}+\alpha\right)\right\}$, let us focus on the case of $i^{*}=\left(\alpha, \ell, n_{\mathrm{st}}+\beta\right)$ for concreteness (the other case can be handled identically). Let $j^{*} \in \mathcal{I}_{\ell-1, \text { out }}^{\xi}$ be the index of the variable $\boldsymbol{w}_{\ell-1, \text { out }}^{\beta}\left(n_{\text {st }}+\alpha\right)$, i.e., the variable that corresponds to $P^{\beta}$, s outgoing message to $P^{\alpha}$ in Round $\ell-1$. Since $\operatorname{P}$-SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}^{C . P r v^{*}(\operatorname{stc},)}$ is an adaptive local assignment generator, we have Equation (E.18). Hence, as in Case 1, we obtain $p^{\alpha ; \beta}\left(i^{*}\right) \leq \operatorname{negl}(\lambda)$ as desired.
Finally, we observe that we have $p^{\alpha \cdot \beta}\left(i^{*}\right) \leq \operatorname{negl}(\lambda)$ for $\forall \alpha, \beta \in[M], \forall \xi \in\{\alpha, \beta\}$, and $\forall i^{*} \in I_{\ell, \text { in }}^{\xi}$. Fix any $\alpha, \beta, \xi$ as above, and let us focus for concreteness on the case of $\xi=\alpha$ (the case of $\xi=\beta$ can be handled identically). Given what is shown in the previous paragraph, it remains to consider the case that $i^{*}=\left(\alpha, \ell, n_{\mathrm{st}}+\gamma\right) \in I_{\ell, \text { in }}^{\alpha}$ for some $\gamma \in[M] \backslash\{\beta\}$, i.e., $i^{*}$ is the index of the variable $\boldsymbol{w}_{\ell, \text { in }}^{\alpha}\left(n_{\text {st }}+\gamma\right)$, which corresponds to $P^{\alpha}$ s incoming message from $P^{\gamma}$ in Round $\ell$. Now, for $\forall \gamma \in[M] \backslash\{\beta\}$, what is shown in the previous paragraph implies that $p^{\alpha \cdot \gamma}\left(i^{*}\right) \leq \operatorname{negl}(\lambda)$. Also, Claim E. 5 and Lemma K. 3 imply that

These two imply that

$$
\begin{aligned}
& p^{\alpha ; \beta}\left(i^{*}\right)
\end{aligned}
$$

$$
\begin{aligned}
& \text { where } Q^{\mu: v}:=T^{\mu} \cup T^{\nu} \text { for } \forall(\mu, v) \notin\{(\alpha, \beta),(\alpha, \gamma)\} \text { and } \\
& Q^{\mu: v}:=\left\{i^{*}\right\} \cup T^{\mu} \cup T^{v} \text { for } \forall(\mu, v) \in\{(\alpha, \beta),(\alpha, \gamma)\} \\
& \leq \operatorname{negl}(\lambda) \text {, }
\end{aligned}
$$

where in the last inequality, we use the fact that the event

$$
\left(A^{\alpha ; \beta}\left(i^{*}\right) \neq A_{\mathrm{corr}}^{\alpha ; \beta}\left(i^{*}\right)\right) \wedge\left(A^{\alpha ; \gamma}\left(i^{*}\right)=A_{\mathrm{corr}}^{\alpha ; \gamma}\left(i^{*}\right)\right)
$$

implies $A^{\alpha ; \beta}\left(i^{*}\right) \neq A^{\alpha ; \gamma}\left(i^{*}\right)$ since we have $A_{\text {corr }}^{\alpha ; \beta}\left(i^{*}\right)=A_{\text {corr }}^{\alpha ; \gamma}\left(i^{*}\right)$ from the definition of $A_{\text {corr }}$.
This completes the proof of Sub-Claim E.3.
Now, we are ready to prove Claim E.8. Fix any $\lambda \in \Lambda \operatorname{and} \operatorname{good}\left(\mathrm{st}_{C}, \operatorname{com}\right)$. For $\forall \ell \in\left[N_{\text {round }}\right]$, let $p_{\text {in }}(\ell)$ be the following probability.

$$
p_{\text {in }}(\ell):=\operatorname{Pr}\left[\begin{array}{c|c}
\operatorname{Correct}\left(\left\{S^{\mu}\right\}_{\mu \in[M]}\right) & \begin{array}{c}
S^{\mu}:=\left\{s_{i}^{\mu}\right\}_{i \in\left[\log ^{2} \lambda\right]} \text { for } \forall \mu \in[M], \text { where } s_{i}^{\mu} \leftarrow I_{\ell, \text { in }, \text { LDE }}^{\mu} \\
\left(f,\left\{A^{\mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{P-SelfCorr} \\
\text { c.Prvili(stc, } c,) \\
\text { where } Q^{\mu: v}:=S^{\mu} \cup S^{v} \text { for } \forall \mu, V(X) \in[M]
\end{array}
\end{array}\right]
$$

Similarly, for $\forall \ell \in\left[N_{\text {round }}\right]$, let $p_{\text {out }}(\ell)$ be the following probability.

From Sub-Claim E. 2 and Sub-Claim E.3, we have

$$
p_{\text {out }}(\ell) \geq p_{\text {in }}(\ell)-N_{\text {Aug }} \epsilon_{2}(\lambda)-\operatorname{negl}(\lambda)
$$

and

$$
p_{\text {in }}(\ell) \geq p_{\text {out }}(\ell-1)-\epsilon_{3}(\lambda)-\operatorname{negl}(\lambda)
$$

Hence, we have

$$
p_{\text {out }}\left(N_{\text {round }}\right) \geq p_{\text {in }}(1)-\operatorname{negl}(\lambda)
$$

Since we have $p_{\text {in }}(1) \geq 1-\operatorname{negl}(\lambda)$ from Sub-Claim E.1, we have

$$
p_{\text {out }}\left(N_{\text {round }}\right) \geq 1-\operatorname{negl}(\lambda)
$$

Combining this with Claim D.6, for $\forall \alpha, \beta \in[M], \xi \in\{\alpha, \beta\}, i^{*} \in I_{N_{\text {round }, \text { out }}}^{\xi}$, we have

$$
\operatorname{Pr}\left[A^{\alpha ; \beta}\left(i^{*}\right)=A_{\mathrm{corr}}^{\alpha: \beta}\left(i^{*}\right) \left\lvert\, \begin{array}{c}
\left(f,\left\{A^{\mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \mathrm{P}-\text { SelfCorr }_{m \mid \boldsymbol{H}, D, D(X)}^{\mathrm{C} \cdot \operatorname{Pr}_{1}^{*}(\operatorname{st}, \cdot)}\left(\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right), \\
\text { where } Q^{\mu: v}:=\emptyset \text { for }(\mu, v) \neq(\alpha, \beta) \text { and } Q^{\alpha ; \beta}:=\left\{i^{*}\right\}
\end{array}\right.\right] \geq 1-\operatorname{negl}(\lambda) .
$$

Fix any $\alpha, \beta \in[M], \xi \in\{\alpha, \beta\}$, and let $i^{*} \in I_{N_{\text {round }}, \text { out }}^{\xi}$ be the index of the variable $\boldsymbol{w}_{N_{\text {round }}, \text { out }}^{\xi}(1)$, which corresponds to the first bit of $P^{\xi}$, s internal state at the end of Round $N_{\text {round }}$ (or equivalently the output of $P^{\xi}$ ). Since P-SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}^{C} \mathrm{Prv}_{1}^{*}\left(\mathrm{st}_{C}\right.$, .) is an adaptive local assignment generator, we have

Hence, from a union bound, we have

$$
\operatorname{Pr}\left[\begin{array}{c|c}
A_{\text {corr }}^{\alpha ; \beta}\left(i^{*}\right)=1 & \left(f,\left\{A^{\mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \text { P-SelfCorr }_{m(H \mid, D(X)}^{\mathrm{C} \cdot \operatorname{Prv}_{1}^{*}\left(\mathrm{st}_{C},\right)}\left(\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right), \\
\text { where } Q^{\mu: v}:=\emptyset \text { for }(\mu, v) \neq(\alpha, \beta) \text { and } Q^{\alpha: \beta}:=\left\{i^{*}\right\}
\end{array}\right] \geq 1-\operatorname{negl}(\lambda) .
$$

From the definition of $A_{\text {corr }}^{\alpha ; \beta}$ and the correctness of $\Pi$, we have $A_{\text {corr }}^{\alpha ; \beta}\left(i^{*}\right)=f\left(x_{\text {сом }}^{*}\right)$. Hence, we have

Hence, from the "moreover" part of Claim D.4, we have

$$
\operatorname{Pr}\left[f\left(x_{\mathrm{coм}}^{*}\right)=1 \left\lvert\, \begin{array}{l|l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \operatorname{R.Prv.} \mathrm{Q}_{1} \\
\left(f, \pi^{*}\right) \leftarrow \operatorname{C.Prv}^{*}\left(\mathrm{st}_{C}, Q\right)
\end{array}\right.\right] \geq 1-\operatorname{negl}(\lambda)
$$

as desired. This completes the proof of Claim E.8.
This completes the proof of Lemma E.3.
Remark 6. By inspecting the proof of Claim E.6, one can easily see that Equation (E.10) holds even for a stronger version of R.Dec ${ }_{1}$ that uses LD-Test. $D_{m_{10}|\boldsymbol{H}|, \zeta}$ in the verification instead of LD-Test. $D_{m_{10}|\boldsymbol{H}|, 3 \zeta}$. (This observation is used later in the proof of Lemma G.2).

## F Step 2: Non-WI Scheme with (1-negl)-Soundness against CNS Provers

As the second step to our commit-and-prove protocol, we give a non-WI commit-and-prove protocol $\left\langle C_{2}, R_{2}\right\rangle$ that is (1-negl)-sound against (not necessarily well-behaving) CNS provers.

In this step, we use a collision-resistant hash function family $\mathcal{H}$. For any hf $\in \mathcal{H}$, we denote by $\mathrm{TreeHash}_{\text {hf }}$ an algorithm that computes the Merkle tree-hash of the input.

Algorithm 10 Commit Phase and Open Phase of $\left\langle C_{2}, R_{2}\right\rangle$

## Commit Phase

Round 1: R.Com 2 sends a hash function hf $\in \mathcal{H}$ to $\mathrm{C}^{\text {. }} \mathrm{Com}_{2}$.
Round 2: Given ( $x_{\text {сом }}$, hf) as input, C.Com 2 runs $\left\{X_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]} \leftarrow$ C.Com $_{1}\left(x_{\text {сом }}\right)$, and then outputs $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}:=\right.$ TreeHash $\left.{ }_{\mathrm{hf}}\left(X_{1, \text { in }}^{\mu}\right)\right\}_{\mu \in[M]}$ as the commitment and (hf, $\left.\left\{X_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}\right)$ as the internal state.

## Open Phase

Round 1: R.Dec. Q ${ }_{2}$ works identically with R.Dec.Q. That is, given $i$ as input, R.Dec. Q $_{2}$ runs $\left(\left\{Q^{\mu}\right\}_{\mu \in[M]}, \mathrm{st}_{R}\right) \leftarrow$ R.Dec. $Q_{1}(i)$, and then outputs $\left\{Q^{\mu}\right\}_{\mu \in[M]}$ as the query and $\mathrm{st}_{R}$ as the internal state.

Round 2: Given (st ${ }_{C},\left\{Q^{\mu}\right\}_{\mu \in[M]}$ ) as input (where st ${ }_{C}=\left(\mathrm{hf},\left\{X_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}\right)$ ), C. $^{\text {Dec }}{ }_{2}$ defines $\left\{\tilde{X}^{\mu \mu \mu}\right\}_{\mu \in[M]}$ just like C.Dec ${ }_{1}$ does, and outputs $\left\{\left.\tilde{Y}^{\mu ; \mu}\right|_{Q^{\mu}}\right\}_{\mu \in[M]}$ as the decommitment, where each $\tilde{Y}^{\mu: \mu}$ is defined as

$$
\tilde{Y}^{\mu ; \mu}(z):=\left\{\begin{array}{ll}
\left(\tilde{X}^{\mu \mu \mu}(z), \operatorname{cert}_{1, \mathrm{in}}^{\mu}(z)\right) & \text { if } z \in D\left(X_{1, \mathrm{in}}^{\mu}\right) \\
\left(\tilde{X}^{\mu ; \mu}(z), \perp\right) & \text { otherwise }
\end{array},\right.
$$

where for each $z=\left(z_{\xi}, z_{\ell}, z_{\text {in }}, i\right) \in D\left(X_{\ell_{\text {in }}}^{\xi}\right)$ (cf. Section D.2.2), $\operatorname{cert}_{1, \text { in }}^{\mu}(z)$ is the certificate of Merkle tree-hash for revealing the $i$-th bit of $X_{1, \text { in }}^{\mu}$.
Verification: Given (st ${ }_{R}$, com, $\left\{\tilde{Y}^{*}{ }^{\mu ; \mu}\right\}_{\mu \in[M]}$ ) as input, R.Dec.D ${ }_{2}$ outputs

$$
\tilde{x}_{i}:=\text { R.Dec.D.D }\left(\mathrm{st}_{R}, \operatorname{com}^{\prime},\left\{\operatorname{Filter}^{\mu}\left(\tilde{Y}^{*}{ }^{\mu: \mu}\right)\right\}_{\mu \in[M]}\right)
$$

as the decommitted value, where com $^{\prime}=\varepsilon$ is an empty string, and Filter ${ }^{\mu}$ is the following function: given input of the form $\left\{\left(x_{z}, \text { cert }_{z}\right)\right\}_{z \in Q}$, it outputs $\left\{\hat{x}_{z}\right\}_{z \in Q}$ such that $\hat{x}_{z}:=x_{z}$ if cert ${ }_{z}$ is a valid certificate w.r.t. (rt ${ }_{1, \text { in }}^{\mu}, z, x_{z}$ ) and $\hat{x}_{z}:=\perp$ otherwise.

## F. 1 Protocol Description

The formal description of $\left\langle C_{2}, R_{2}\right\rangle$ is given in Algorithm 10 and Algorithm 11.

## F. 2 Proof of Binding

Lemma F.1. Let $\epsilon$ be any negligible function, $\kappa_{\mathrm{dec}}$ be the polynomial that is given in Lemma E.1, and $C_{2}^{*}=$ (C.Com ${ }_{2}^{*}, \mathrm{C}^{2} \mathrm{Dec}_{2}^{*}$ ) be any $\kappa_{\mathrm{dec}}-$ CNS cheating committer-decommitter against $\left\langle C_{2}, R_{2}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following binding condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{2}^{*}, \mathrm{R} . \mathrm{Com}{ }_{2}\right\rangle$.

- Binding Condition: If it holds
then for every $i \in[n]$ it holds $\operatorname{Pr}\left[b_{B A D}=1\right] \leq \operatorname{negl}(\lambda)$ in the following probabilistic experiment $E X P_{2}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}_{2}^{*}, \mathrm{st}_{C}, \mathrm{com}, i\right)$.

1. For each $\forall b \in\{0,1\}$, sample $\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}$ by $\left(\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}\right.$, st $\left._{b}\right) \leftarrow$ R.Dec. $Q_{2}(i)$.
2. Run $\left\{\tilde{Y}^{*}{ }_{b}^{\mu \cdot \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \operatorname{C.Dec}_{2}^{*}\left(\mathrm{st}_{C},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$.
3. Let $b_{B A D}:=1$ if and only if $x_{0}^{*} \neq \perp \wedge x_{1}^{*} \neq \perp \wedge x_{0}^{*} \neq x_{1}^{*}$ holds, where $x_{b}^{*}:=\operatorname{R.Dec} . \mathrm{D}_{2}\left(\mathrm{st}_{b}, \operatorname{com},\left\{\tilde{Y}^{*}{ }_{b}^{\mu \cdot \mu}\right\}_{\mu \in[M]}\right)$ for each $b \in\{0,1\}$.

Proof. Fix any $\epsilon$ and $C_{2}^{*}=\left(\mathrm{C} . \mathrm{Com}_{2}^{*}, \mathrm{C}_{2} \mathrm{Dec}_{2}^{*}\right)$ as above, and assume for contradiction that for infinitely many $\lambda \in \mathbb{N}$, the binding condition does not hold with non-negligible probability over the choice of (st $\left.{ }_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2}\right.$ Com $_{2}^{*}$, R.Com $\left.{ }_{2}\right\rangle$.

To obtain a contradiction with the binding property of $\left\langle C_{1}, R_{1}\right\rangle$ (Lemma E.1), we consider the following cheating committer-decommitter $C_{1}^{*}=\left(\mathrm{C}_{1} \mathrm{Com}_{1}^{*}\right.$, $\left.\mathrm{C} . \mathrm{Dec}_{1}^{*}\right)$ against $\left\langle C_{1}, R_{1}\right\rangle$.

- Committer. C.Com ${ }_{1}^{*}$ runs $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{2}^{*}\right.$, R.Com $\left.{ }_{2}\right\rangle$ internally, sends an empty string to R.Com ${ }_{1}$ as the commitment, and stores $\left(\mathrm{com}, \mathrm{st}_{C}\right)$ as the internal state.
- Decommitter. Given (com, st ${ }_{C}$ ) and $\left\{Q^{\mu}\right\}_{\mu \in[M]}$ as input, C.Dec ${ }_{1}^{*}$ runs $\left\{\tilde{Y}^{\mu}{ }^{\mu / \mu}\right\}_{\mu \in[M]} \leftarrow$ C.Dec $_{2}^{*}\left(\operatorname{st}_{C},\left\{Q^{\mu}\right\}_{\mu \in[M]}\right)$ internally, and sends $\left\{\text { Filter }^{\mu}\left(\tilde{Y}^{*}{ }^{\mu / \mu}\right)\right\}_{\mu \in[M]}$ to R.Dec ${ }_{1}$ as the decommitment, where the function Filter ${ }^{\mu}\left(\tilde{Y}^{*}{ }^{\mu \cdot \mu}\right)$ is defined as in $\left\langle C_{2}, R_{2}\right\rangle$

```
Algorithm 11 Prove Phase of \(\left\langle C_{2}, R_{2}\right\rangle\)
Prove Phase
Round 1: R.Prv. \(Q_{2}\) works identically with R.Prv. \(Q_{1}\). That is, R.Prv. \(Q_{2}\) runs \(\left(\left\{Q^{\mu, \nu}\right\}_{\mu, v \in[M]},\left\{\mathrm{st}_{V}^{\mu, \nu}\right\}_{\mu, \nu \in[M]}\right) \leftarrow\) R.Prv. \(Q_{1}\),
and outputs \(\left\{Q^{\mu, v_{\mu}}\right\}_{\mu, v \in[M]}\) as the query and \(\left\{\mathrm{st}_{V}^{\mu, \nu}\right\}_{\mu, v \in[M]}\) as the internal state.
```

Round 2: Given (st ${ }_{C}, f,\left\{Q^{\mu, v}\right\}_{\mu, v \in[M]}$ ) as input, C. $\operatorname{Prv}_{2}$ does the following.

1. Obtain $\left\{\text { view }^{\mu}\right\}_{\mu \in[M]}$ just like C.Prv ${ }_{1}$ does.
2. Run $\left(\mathrm{rt}^{\mu}, \mathrm{rt}^{\nu}, \pi^{\mu: v}\right) \leftarrow \mathrm{PCP} . \mathrm{P}^{\prime}\left(\mu, v, f\right.$, view $^{\mu}$, view ${ }^{\nu}$, hf) for every $\mu, v \in[M]$, where PCP. $\mathrm{P}^{\prime}$ is identical with PCP.P except for the following.

- A hash function hf is given as an additional input.
- The hash of $X^{\mu}$ and $X^{\nu}$, denoted by $\mathrm{r}^{\mu}:=\operatorname{TreeHash}_{\mathrm{hf}}\left(X^{\mu}\right)$ and $\mathrm{rt}^{\nu}:=\operatorname{TreeHash}_{\mathrm{hf}}\left(X^{\nu}\right)$, are computed as additional outputs.
- The proof $\pi^{\mu: v}$ given by PCP.P ${ }^{\prime}$ is $\pi^{\mu: v}=\left(Y^{\mu: v}, \ldots\right)$ instead of $\pi^{\mu: \nu}=\left(X^{\mu: v}, \ldots\right)$, where $Y^{\mu: v}$ is defined as

$$
Y^{\mu: v}(z):= \begin{cases}\left(X^{\mu: v}(z), \operatorname{cert}_{1, \text { in }}^{\xi}(z)\right) & \text { if } \exists \xi \in\{\mu, v\} \text { s.t. } z \in D\left(X_{1, \text { in }}^{\xi}\right) \\ \left(X^{\mu: v}(z), \operatorname{cert}^{\xi}(z)\right) & \text { if } \exists \xi \in\{\mu, v\} \text { s.t. } z \in D\left(X^{\xi}\right) \backslash D\left(X_{1, \text { in }}^{\xi}\right), \\ \left(X^{\mu: v}(z), \perp\right) & \text { otherwise }\end{cases}
$$

where $\operatorname{cert}_{1, \text { in }}^{\xi}(z)$ is defined as in the open phase, and $\operatorname{cert}^{\xi}(z)$ is defined as follows: for each $z=\left(z_{\xi}, i\right) \in D\left(X^{\xi}\right)$ (cf. Section D.2.2), $\operatorname{cert}^{\xi}(z)$ is the certificate of Merkle tree-hash for revealing the $i$-th bit of $X^{\xi}$.
3. Output $\left(\left\{\mathrm{r}^{\mu}\right\}_{\mu \in[M]},\left\{\left.\pi^{\mu \nu v}\right|_{Q^{\mu v}}\right\}_{\mu, v \in[M]}\right)$ as the proof. (Note: the value of $\mathrm{r} \mathrm{t}^{\mu}$ that is computed by PCP.P $(\mu, v, \ldots)$ and the value of it that is computed by PCP. $\mathrm{P}^{\prime}(\mu, \xi, \ldots)$ are identical; see Remark 3.)

Verification: Given (st ${ }_{R}$, com, $f,\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\pi^{* \mu: \nu}\right\}_{\mu, v \in[M]}$ ) as input (where $\mathrm{st}_{R}=\left\{\mathrm{st}_{V}^{\mu, \nu}\right\}_{\mu, v \in[M]}$ and com $=\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$ ), R.Prv. $D_{2}$ does the following.

1. Run $b^{\mu: v} \leftarrow \operatorname{PCP} . \mathrm{D}^{\prime \geq \lambda-\xi}\left(\mathrm{st}_{V}^{\mu \nu \nu}, f^{\prime}, \mathrm{rt}_{1, \mathrm{in}}^{\mu}, \mathrm{rt}_{1, \mathrm{in}}^{v}, \mathrm{rt}^{\mu}, \mathrm{rt}^{\nu}, \pi^{* \mu: v}\right)$ for every $\mu, v \in[M]$, where PCP. $\mathrm{D}^{\prime}$ is identical with PCP.D except for the following.

- The TreeHash ${ }_{\mathrm{hf}}$ roots $\mathrm{rt}_{1, \mathrm{n}}^{\mu}, \mathrm{rt}_{1, \mathrm{n}}^{\nu}, \mathrm{rt}^{\mu}, \mathrm{rt}^{\nu}$ are given as additional inputs.
- Each test for $X^{\mu: v}$ is made in the following manner.
(a) Make the query to $Y^{\mu: v}$ instead of to $X^{\mu: v}$. Let $\left\{\left(x_{z}, \text { cert }_{z}\right)\right\}_{z \in Q}$ be the response from $Y^{\mu: \nu}$.
(b) Verify the test by considering Filter ${ }^{\mu: v}\left(\left\{\left(x_{z}, \operatorname{cert}_{z}\right)\right\}_{z \in Q}\right)$ to be the response from $X^{\mu: v}$, where Filter ${ }^{\mu: v}$ is the following function: given input $\left\{\left(x_{z}, \text { cert }_{z}\right)\right\}_{z \in Q}$, it outputs $\left\{\hat{x}_{z}^{\mu: v}\right\}_{z \in Q}$ such that

$$
\hat{x}_{z}^{\mu: \nu}:=\left\{\begin{array}{cc} 
& \text { if } \exists \xi \in\{\mu, v\} \text { s.t. } z \in D\left(X_{1, \text { in }}^{\xi}\right) \text { and cert } \\
& \text { w.r.t. }\left(\mathrm{rt}_{1, \text { in }}^{\xi}, z, x_{z}\right), \\
& \text { or } \exists \xi \in\{\mu, v\} \text { s.t. } z \in D\left(X^{\xi}\right) \backslash D\left(X_{1, \text { in }}^{\xi}\right) \text { and cert } z_{z} \text { is not a valid certificate } \\
& \text { certificate w.r.t. }\left(\mathrm{rt}^{\xi}, z, x_{z}\right)
\end{array}\right.
$$

2. Output 1 if and only if $b^{\mu: v}=1$ for every $\mu, v \in[M]$.

In what follows, we observe that (1) the binding condition of $\left\langle C_{1}, R_{1}\right\rangle$ does not hold with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$, and (2) $C_{1}^{*}$ is well-behaving and CNS.

First, the binding condition of $\left\langle C_{1}, R_{1}\right\rangle$ does not hold with non-negligible probability over the choice of (st $\left.{ }_{C}, \mathrm{com}\right) \leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$ since $C_{1}^{*}$ perfectly emulates (R.Com ${ }_{2}, \mathrm{R}^{2} \mathrm{Dec}_{2}$ ) for the internally emulated $C_{2}^{*}$.

Second, $C_{1}^{*}$ is a well-behaving CNS committer-decommitter since (1) the CNS property of C.Dec ${ }_{1}^{*}$ follows from that of C.Dec ${ }_{2}^{*}$, and (2) the consistency on $D\left(X_{1, \text { in }}^{\mu}\right)$ follows from the binding property of TreeHash ${ }_{\mathrm{hf}}$.

Hence, we obtain a contradiction.

## F. 3 Proof of Soundness

Lemma F.2. Let $\epsilon_{\text {SND }}$ be any negligible function, $\kappa_{\text {prv }}$ be the polynomial that is given in Lemma E.3, $E_{2}$ be the extractor in Algorithm 12, and $C_{2}^{*}=\left(\mathrm{C}_{2} \mathrm{Com}_{2}^{*}, \mathrm{C} . \mathrm{Prv}_{2}^{*}\right)$ be any $\kappa_{\mathrm{prv}}-\mathrm{CNS}$ cheating committer-prover against $\left\langle C_{2}, R_{2}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following soundness condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{2}^{*}\right.$, R.Com $\left.{ }_{2}\right\rangle$.

- Soundness Condition: If it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{2} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{2}^{*}\left(\mathrm{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{2}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array} \tag{F.1}
\end{array}\right] \geq 1-\epsilon_{\text {SND }}(\lambda),
$$

then there exists $x_{\text {сом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that

$$
\begin{equation*}
\forall i \in[n], \operatorname{Pr}\left[x_{i}=x_{i}^{*} \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{2}^{\mathrm{C} \cdot \operatorname{Prv}_{2}^{*}\left(\mathrm{st}_{C},\right)}(\operatorname{com}, i), \mathrm{R}^{2} \cdot \mathrm{Dec}_{2}(\operatorname{com}, i)\right\rangle\right] \geq 1-\operatorname{negl}(\lambda) \tag{F.2}
\end{equation*}
$$

and

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{2} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{2}^{*}\left(\mathrm{st}_{C}, Q\right)  \tag{F.3}\\
\wedge f\left(x_{\text {Cом }}^{*}\right)=0 & b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{2}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \leq \operatorname{negl}(\lambda)
$$

```
Algorithm 12 Extractor \(E_{2}\) (against \(\left\langle C_{2}, R_{2}\right\rangle\) )
Input: com, \(i\), and \(\left\{Q^{\mu}\right\}_{\mu \in[M]} \subset D\left(X_{1, \text { in }}^{1}\right) \times \cdots \times D\left(X_{1, \text { in }}^{M}\right)\)
```

1. Run $\left(f,\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C.Prv}{ }_{2}^{*}\left(\operatorname{st}_{C},\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right)$, where each $Q^{\mu: v}$ is defined as $Q^{\mu: v}:=Q^{\mu}$ if $\mu=v$ and $Q^{\mu: v}:=\emptyset$ otherwise.
2. Output $\left\{\tilde{Y}^{\mu ; \mu}\right\}_{\mu \in[M]}$ as the decommitment, where $\tilde{Y}^{*}{ }^{\mu ; \mu}:=\pi^{* \mu ; \mu}$.

Proof. Fix any $\epsilon_{\text {SND }}$ and $C_{2}^{*}=\left(\mathrm{C} . \mathrm{Com}_{2}^{*}, \mathrm{C} . \mathrm{Prv}_{2}^{*}\right)$ as in the lemma statement, and assume for contradiction that for infinitely many $\lambda \in \mathbb{N}$, the soundness condition does not hold with non-negligible probability over the choice of (st ${ }_{C}$, com) $\leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{2}^{*}\right.$, R.Com $\left.{ }_{2}\right\rangle$.

To obtain a contradiction with the soundness of $\left\langle C_{1}, R_{1}\right\rangle$ (Lemma E.3), we consider the following cheating committerprover $C_{1}^{*}=\left(\mathrm{C} . \mathrm{Com}_{1}^{*}, \mathrm{C}\right.$. Prv $\left._{1}^{*}\right)$ against $\left\langle C_{1}, R_{1}\right\rangle$.

- Committer. C.Com ${ }_{1}^{*}$ runs $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{\left(\mathrm{Com}_{2}^{*}, \text { R. } \mathrm{Com}_{2}\right\rangle \text { internally, sends an empty string to R.Com }}\right.$ as the commitment, and stores ( $\mathrm{com}, \mathrm{st}_{C}$ ) as the internal state.
- Prover. Given (com, st $C_{C}$ ) and $\left\{Q^{\mu: \nu}\right\}_{\mu, v \in[M]}$ as input, C.Prv ${ }_{1}^{*}$ first runs $\left(f,\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\pi^{* \mu: \nu}\right\}_{\mu, v \in[M]}\right) \leftarrow$ $\mathrm{C} . \operatorname{Prv}_{2}^{*}\left(\operatorname{st}_{C},\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}\right)$. For each $\mu, v \in[M]$, let ${\hat{\pi^{*}}}^{\mu: v}: Q^{\mu: v} \rightarrow \boldsymbol{F} \cup\{\perp\}$ be defined by

$$
\forall z \in Q^{\mu: v}: \hat{\pi}^{\mu / v}(z):=\left\{\begin{array}{ll}
\hat{x}_{z}^{\mu: \nu} & \text { if } z \in Q^{\mu: v} \cap D(X) \\
\pi^{* \mu: v}(z) & \text { otherwise }
\end{array},\right.
$$

where $\left\{\hat{x}_{z}^{\mu: v}\right\}_{z \in Q^{\mu: v} \cap D(X)}:=\operatorname{Filter}^{\mu: v}\left(\left.\pi^{* \mu: v}\right|_{Q^{\mu: v} \cap D(X)}\right)$ is defined as in $\left\langle C_{2}, R_{2}\right\rangle$. Then, C.Prv ${ }_{1}^{*}$ sends $\left(f,\left\{\hat{\pi}^{\mu / v}\right\}_{\mu, v \in[M]}\right)$ to R. $\operatorname{Prv}_{1}$ as the proof.

In what follows, we observe that (1) the soundness condition of $\left\langle C_{1}, R_{1}\right\rangle$ does not hold with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$, and (2) $C_{1}^{*}$ is well-behaving CNS.

First, the soundness condition of $\left\langle C_{1}, R_{1}\right\rangle$ does not hold with non-negligible probability over the choice of (st ${ }_{C}$, com) $\leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{1}^{*}\right.$, R.Com $\left.{ }_{1}\right\rangle$ since $C_{1}^{*}$ perfectly emulates (R.Com, R. $\mathrm{Dec}_{2}, \mathrm{R} . \mathrm{Prv}_{2}$ ) and $E_{2}$ for the internally emulated $C_{2}^{*}$.

Second, $C_{1}^{*}$ is a well-behaving committer-prover since (1) the CNS property of C.Prv ${ }_{1}^{*}$ follows from that of C.Prv ${ }_{2}^{*}$, and (2) the consistencies on $D\left(X_{1, \mathrm{in}}^{\mu}\right)$ and on $D\left(X^{\mu}\right)$ follow from the binding property of TreeHash ${ }_{\mathrm{hf}}$.

Hence, we obtain a contradiction.

Remark 7. Given the observation in Remark 6, one can easily see that Equation (F.2) holds even for a stronger version of R. Dec $_{2}$ that uses LD-Test. $D_{m_{10}|\boldsymbol{H}|, \zeta}$ in the verification instead of LD-Test. $D_{m_{\mathrm{to}}|\boldsymbol{H}|, 3 \zeta}$. (This observation is used later in the proof of Lemma G.2).

## G Step 3: Non-WI Scheme with negl-Soundness against CNS Provers

As the third step to our commit-and-prove protocol, we give a non-WI commit-and-prove protocol $\left\langle C_{3}, R_{3}\right\rangle$ that is neglsound against CNS provers.

In this step, we use a slightly extended version of the soundness amplification lemma of Brakerski et al. [BHK17], which is given as Lemma K. 4 in Section K.2.

## G. 1 Protocol Description

The formal description of $\left\langle C_{3}, R_{3}\right\rangle$ is given in Algorithm 13.

```
Algorithm 13 Commit Phase, Open Phase, and Prove Phase of \(\left\langle C_{3}, R_{3}\right\rangle\)
Commit Phase
The commit phase of \(\left\langle C_{3}, R_{3}\right\rangle\) is identical with that of \(\left\langle C_{2}, R_{2}\right\rangle\).
```


## Open Phase

```
The open phase of \(\left\langle C_{3}, R_{3}\right\rangle\) is identical with that of \(\left\langle C_{2}, R_{2}\right\rangle\) except that in the verification, R.Dec. \(D_{3}\) uses LD-Test. \(D_{m_{10}|\boldsymbol{H}|, \zeta}\) instead of LD-Test. \(\mathrm{D}_{m_{10}|\boldsymbol{H}|, 3 \zeta}\) for testing each \(\widetilde{X}^{\mu / \mu}\).
```


## Prove Phase

The prove phase of $\left\langle C_{3}, R_{3}\right\rangle$ is identical with that of $\left\langle C_{2}, R_{2}\right\rangle$ except that in the verification, R.Prv. $D_{3}$ uses PCP.D ${ }^{\prime \otimes \lambda}$ instead of PCP. $\mathrm{D}^{\prime \geq \lambda-\zeta}$ for verifying each proof $\pi^{\mu: \nu}$.

## G. 2 Proof of Binding

Lemma G.1. Let $\kappa_{V}$ be the query complexity of (PCP.P, PCP.V), $\kappa_{\text {dec }}$ be the polynomial that is given in Lemma E.1, and $C_{3}^{*}=\left(\mathrm{C}^{\mathrm{Com}}{ }_{3}^{*}, \mathrm{C} . \mathrm{Dec}_{3}^{*}\right)$ be any $\left(\kappa_{\mathrm{dec}}+\lambda \kappa_{v}\right)$-CNS cheating committer-decommitter against $\left\langle C_{3}, R_{3}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following binding condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{3}^{*}\right.$, R.Com $\left.{ }_{3}\right\rangle$.

- Binding Condition: For every $i \in[n]$, it holds $\operatorname{Pr}\left[b_{B A D}=1\right] \leq \operatorname{negl}(\lambda)$ in the following probabilistic experiment $E x P_{3}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}_{3}^{*}, \mathrm{st}_{C}, \mathrm{com}, i\right)$.

1. For each $\forall b \in\{0,1\}$, sample $\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}$ by $\left(\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}\right.$, st $\left._{b}\right) \leftarrow$ R.Dec. $Q_{3}(i)$.
2. Run $\left\{\tilde{Y}^{*}{ }_{b}^{\mu \cdot \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \operatorname{C.} \operatorname{Dec}_{3}^{*}\left(\mathrm{st}_{C},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$.
3. Let $b_{B A D}:=1$ if and only if $x_{0}^{*} \neq \perp \wedge x_{1}^{*} \neq \perp \wedge x_{0}^{*} \neq x_{1}^{*}$ holds, where $x_{b}^{*}:=$ R.Dec. $\mathrm{D}_{3}\left(\mathrm{st}_{b}, \operatorname{com},\left\{\tilde{Y}^{*}{ }_{b}^{\mu \cdot \mu}\right\}_{\mu \in[M]}\right)$ for each $b \in\{0,1\}$.

Proof. Fix any $C_{3}^{*}=\left(\mathrm{C}_{1} \mathrm{Com}_{3}^{*}, \mathrm{C}_{\mathrm{C}} \mathrm{Dec}_{3}^{*}\right)$ as above, and assume for contradiction that for infinitely many $\lambda \in \mathbb{N}$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{\prime} \mathrm{Com}_{3}^{*}\right.$, R.Com $\left.{ }_{3}\right\rangle$, there exists $i \in[n]$ such that we have $\operatorname{Pr}\left[b_{\text {BAD }}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in the experiment $\operatorname{Exp}_{3}^{\text {bind }}\left(\right.$ C. $^{\text {Dec }}{ }_{3}^{*}$, st $_{C}$, com, $\left.i\right)$.

To obtain a contradiction, we first define a cheating committer-decommitter $C_{2}^{*}=\left(\mathrm{C}_{2} \mathrm{Com}_{2}^{*}, \mathrm{C} . \operatorname{Dec}{ }_{2}^{*}\right)$ against $\left\langle C_{2}, R_{2}\right\rangle$ by using $C_{3}^{*}$, and then show that $C_{2}^{*}$ breaks the binding property of $\left\langle C_{2}, R_{2}\right\rangle$.

Let us first give some preliminaries. First, from the definitions of R.Dec.Q Q $_{3}$ and R.Dec. $D_{3}$, the probabilistic experiment $\operatorname{Exp}_{3}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}^{*}, \mathrm{st}_{C}\right.$, com, $\left.i\right)$ in the lemma statement can also be written as follows.

1. For $\forall b \in\{0,1\}$, sample $\left\{Q_{b}^{\mu}\right\}_{\mu \in[M]}$ as follows.
(a) $\operatorname{Run}\left(Q_{b, 0}^{\mu}, \mathrm{st}_{b, 0}^{\mu}\right) \leftarrow$ LD-Test. $\mathrm{Q}_{D\left(X_{1, \mathrm{i})}^{\mu}\right)}$ for $\forall \mu \in[M]$.
(b) Run $\left(Q_{b, 1}^{\mu}, \mathrm{st}_{b, 1}^{\mu}\right) \leftarrow$ SelfCorr. $\mathrm{Q}_{D\left(X_{1, \mathrm{n}}^{\mu}\right)}(\{(\mu, 1, i)\})$ for $\forall \mu \in[M]$.
(c) Let $Q_{b}^{\mu}:=Q_{b, 0}^{\mu} \cup Q_{b, 1}^{\mu}$.
2. Run $\left\{\tilde{Y}_{b}^{\mu}{ }_{b}^{\mu \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \operatorname{C.Dec}^{*}\left(\operatorname{st}_{C},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$.

For each $b \in\{0,1\}, \mu \in[M]$, let $\tilde{Y}_{b, 0}^{*}:=\left.\tilde{Y}^{*}{ }_{b}^{\mu \mu \mu}\right|_{Q_{b, 0}^{\mu}}$ and $\tilde{Y}_{b, 1}^{*}{ }_{b}^{\mu \cdot \mu}:=\left.\tilde{Y}_{b}^{*}{ }_{b}^{\mu \cdot \mu}\right|_{Q_{b, 1}^{\mu}}$
3. Let $b_{\text {BAD }}:=1$ if and only if both of the following events hold.

- Event ${ }_{0}$ : LD-Test. $D_{m_{10}|\boldsymbol{H}|, \zeta}\left(\mathrm{st}_{b, 0}^{\mu}, \operatorname{Filter}^{\mu}\left(\tilde{Y}_{b, 0}^{*}\right)\right)=1$ holds for $\forall b \in\{0,1\}, \mu \in[M]$.
- Event ${ }_{1}: x_{0}^{*} \neq \perp \wedge x_{1}^{*} \neq \perp \wedge x_{0}^{*} \neq x_{1}^{*}$ holds for $x_{0}^{*}, x_{1}^{*}$ that are defined as follows.
(a) Let $\tilde{x}_{b}^{\mu}:=A_{b}^{\mu}(\mu, 1, i)$, where $A_{b}^{\mu}:=\operatorname{SelfCorrr} \operatorname{Rec}_{m_{\text {to }}|\boldsymbol{H}|}\left(\operatorname{st}_{b, 1}^{\mu}, \operatorname{Filter}^{\mu}\left(\tilde{Y}^{*}{ }_{b, 1}^{\mu \cdot \mu}\right)\right)$.
(b) Let $x_{b}^{*}:=\perp$ if $\exists \mu \in[M]$ such that $\tilde{x}_{b}^{\mu} \notin\{0,1\}$, and let $x_{b}^{*}:=\tilde{x}_{b}^{1} \oplus \cdots \oplus \tilde{x}_{b}^{M}$ otherwise.
(Note that Event depends only on $\left\{\tilde{Y}^{*}{ }_{b, 0}^{\mu \cdot \mu}\right\}_{b \in\{0,1\}, \mu \in[M]}$ while Event ${ }_{1}$ depends only on $\left\{\tilde{Y}^{*}{ }_{b, 1}^{\mu \cdot \mu}\right\}_{b \in\{0,1\}, \mu \in[M]}$.)
Second, the probabilistic experiment Exp ${ }_{2}^{\text {bind }}\left(\mathrm{C}\right.$. Dec $^{*}$, $\mathrm{st}_{C}$, com, $i$ ) in Lemma F. 1 can be written similarly, where the only difference is that Event ${ }_{0}$ is replaced with the following event.
- Event ${ }_{0}^{\prime}$ : LD-Test. $\mathrm{D}_{m_{\mathrm{o}}|\boldsymbol{H}|, 3 \zeta}\left(\mathrm{St}_{b, 0}^{\mu}\right.$, Filter $\left.^{\mu}\left(\tilde{Y}^{*}{ }_{b, 0}^{\mu \mu}\right)\right)=1$ holds for $\forall b \in\{0,1\}, \mu \in[M]$.

Now, we define $C_{2}^{*}=\left(\mathrm{C} . \mathrm{Com}_{2}^{*}, \mathrm{C} . \mathrm{Dec}_{2}^{*}\right)$. Recall that we assume, for contradiction, that for infinitely many $\lambda$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{3}^{*}\right.$, R.Com $\left.{ }_{3}\right\rangle$, there exists $i \in[n]$ such that we have $\operatorname{Pr}\left[b_{\text {BAD }}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in $\operatorname{Exp}_{3}^{\text {bind }}\left(\mathrm{C}^{2} . \mathrm{Dec}_{3}^{*}, \mathrm{st}_{C}, \operatorname{com}, i\right)$. Let us call such $\lambda$, ( $\mathrm{st}_{C}, \operatorname{com}$ ), and $i$ be good. Then, the CNS of $\mathrm{C} . \mathrm{Dec}_{3}^{*}$ guarantees that there exists a constant $c \in \mathbb{N}$ such that for every good $\lambda$ and ( $\mathrm{st}_{C}, \mathrm{com}$ ), we have

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\text { Event }_{0} & \begin{array}{l}
\left(Q_{b, 0}^{\mu}, \mathrm{st}_{b, 0}^{\mu}\right) \leftarrow \text { LD-Test. } Q_{D\left(X_{1, \mathrm{n}}^{\mu}\right.} \\
\left\{\tilde{Y}_{b, 0}^{*}{ }_{b, 0}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \mathrm{C.Dec}_{3}^{*}\left(\mathrm{st}_{C},\left\{Q_{b, 0}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}^{\mu}\right)
\end{array}
\end{array}\right] \geq \frac{1}{\lambda^{c}}
$$

(where Event ${ }_{0}$ is defined as in Exp $_{3}^{\text {bind }}$ ). Hence, we can use Lemma K. 4 (Soundness Amplification Lemma) to obtain a Ppt oracle algorithm Amplify ${ }_{c}$ such that for every good $\lambda$ and (st ${ }_{C}$, com), we have
(where Event ${ }_{0}^{\prime}$ is defined as in Exp ${ }_{2}^{\text {bind }}$ ). Given Assign ${ }_{c}$, we define $C_{2}^{*}=\left(\mathrm{C} . \mathrm{Com}_{2}^{*}, \mathrm{C}_{\mathrm{D}} \mathrm{Dec}_{2}^{*}\right.$ ) as follows.

- Committer. C.Com ${ }_{2}^{*}$ works identically with C.Com ${ }_{3}^{*}$.
- Decommitter. Given st ${ }_{C}$ and $\left\{Q^{\mu}\right\}_{\mu \in[M]}$ as input, C.Dec ${ }_{2}^{*}$ works identically with Amplify ${ }_{c}{ }^{\text {C.Dec }}{ }_{3}^{*}\left(\right.$ st $\left._{C, \cdot}\right)\left(\left\{Q^{\mu}\right\}_{\mu \in[M]}\right)$.

Now, our goal is to show that $C_{2}^{*}$ breaks the binding property of $\left\langle C_{2}, R_{2}\right\rangle$. Since the CNS of $C_{2}^{*}$ follows from Lemma K. $4,{ }^{33}$ we focus on showing that for infinitely many $\lambda$, the binding condition of $\left\langle C_{2}, R_{2}\right\rangle$ in Lemma F. 1 does not hold with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\right.$ C.Com $_{2}^{*}$, R.Com $\left.{ }_{2}\right\rangle$. Toward this end, it suffices to show that for every good $\lambda$ and (st ${ }_{C}$, com), we have both

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\text { Event }_{0}^{\prime} & \begin{array}{l}
\left(Q_{b}^{\mu}, \mathrm{St}_{b}^{\mu}\right) \leftarrow \text { LD-Test. } \mathrm{Q}_{D\left(X_{1, \mathrm{n}}^{\mu}\right.} \text { for } \forall b \in\{0,1\}, \mu \in[M] \\
\left\{\tilde{Y}_{b}^{*}{ }_{b}^{\mu \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \mathrm{C.Dec}_{2}^{*}\left(\mathrm{St}_{C},\left\{Q_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)
\end{array} \tag{G.2}
\end{array}\right] \geq 1-\epsilon(\lambda)
$$

 inition, Equation (G.2) follows from Equation (G.1). Second, we obtain $\operatorname{Pr}\left[b_{\mathrm{BAD}}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in $\operatorname{Exp}_{2}^{\text {bind }}$ (C.Dec ${ }_{2}^{*}$, , $\mathrm{st}_{C}$, com, $i$ ) as follows.

- On the one hand, since we have

$$
\operatorname{Pr}\left[\text { Event }_{1} \mid \text { Event }_{0}\right] \geq \operatorname{Pr}\left[\text { Event }_{1} \wedge \text { Event }_{0}\right]=\operatorname{Pr}\left[b_{\mathrm{BAD}}=1\right] \geq \frac{1}{\operatorname{poly}(\lambda)}
$$

in $\operatorname{Exp}_{3}^{\text {bind }}\left(\mathrm{C}^{2} . \mathrm{Dec}_{3}^{*}, \mathrm{st}_{C}\right.$, com, $i$ ), the "furthermore" part of Lemma K. 4 guarantees that we have $\operatorname{Pr}\left[\right.$ Event $\left._{1}\right] \geq$ 1/poly $(\lambda)$ in $\operatorname{Exp}_{2}^{\text {bind }}\left(\right.$ C.Dec $_{2}^{*}$, st $_{C}$, com,$\left.i\right)$.

- On the other hand, Equation (G.2) and the CNS of C.Dec $2_{2}^{*}$ guarantees that we have $\operatorname{Pr}\left[\right.$ Event $\left._{0}^{\prime}\right] \geq 1-\operatorname{negl}(\lambda)$ in $\operatorname{Exp}_{2}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}_{2}^{*}, \mathrm{st}_{C}, \mathrm{com}, i\right)$.
- Hence, from a union bound, we have

$$
\operatorname{Pr}\left[b_{\mathrm{BAD}}=1\right]=\operatorname{Pr}\left[\text { Event }_{0}^{\prime} \wedge \text { Event }_{1}\right] \geq \frac{1}{\operatorname{poly}(\lambda)}-\operatorname{negl}(\lambda) \geq \frac{1}{\operatorname{poly}(\lambda)}
$$

in $\operatorname{Exp}_{2}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}_{2}^{*}\right.$, $\mathrm{st}_{C}$, com, $\left.i\right)$ as desired.
This completes the proof of Lemma G.1.

[^20]
## G. 3 Proof of Soundness

We first remark that, due to Lemma K. 4 and the definition of $\mathrm{C} . \mathrm{Prv}_{3}^{*}$, for every constant $c \in \mathbb{N}$, there is a ppt oracle algorithm Amplify ${ }_{c}$ such that for every $\kappa_{\max }-\mathrm{CNS}$ cheating prover $C_{3}^{*}=\left(\mathrm{C} . \mathrm{Com}_{3}^{*}, \mathrm{C} . \operatorname{Prv} \mathrm{v}_{3}^{*}\right)$ and every $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{3}^{*}\right.$, R.Com $\left._{3}\right\rangle$, if it holds

$$
\operatorname{Pr}\left[b=1 \left\lvert\, \begin{array}{l|l}
\left.b, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{3} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C.Prv}_{3}^{*}\left(\operatorname{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{3}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right.\right] \geq \frac{1}{\lambda^{c}}
$$

for infinitely many $\lambda$ (let $\Lambda$ be the set of such $\lambda$ ), then Amplify ${ }^{C \cdot P r v}{ }_{3}^{*}\left(\operatorname{st}_{C}, \cdot\right)$ is an adaptive ( $\kappa_{\max }-\lambda \kappa_{\mathrm{v}}$ )-CNS cheating prover such that there is a negligible function negl such that for every $\lambda \in \Lambda$,

$$
\operatorname{Pr}\left[b=1 \left\lvert\, \begin{array}{l|l}
b & \left(Q, \mathrm{st}_{R}\right) \leftarrow \operatorname{R.Prv} \cdot \mathrm{Q}_{2} ;\left(f, \pi^{*}\right) \leftarrow \operatorname{Amplify}_{c}^{\mathrm{C} \cdot \operatorname{Prv}_{3}^{*}\left(\operatorname{sta}_{C}, \cdot\right)}(Q) \\
b \leftarrow \operatorname{R.Prv} \cdot \mathrm{D}_{2}\left(\operatorname{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right.\right] \geq 1-\operatorname{negl}(\lambda) .
$$

Now, as the soundness of $\left\langle C_{3}, R_{3}\right\rangle$, we prove the following lemma.
Lemma G.2. Let $c \in \mathbb{N}$ be any constant, $\kappa_{v}$ be the query complexity of (PCP.P, PCP.V), $\kappa_{\text {prv }}$ be the polynomial that is given in Lemma E.3, and let $E_{3}$ be the extractor in Algorithm 14. Then, for any ( $\kappa_{\mathrm{prv}}+\lambda \kappa_{v}$ )-CNS cheating committerprover $C_{3}^{*}=\left(\mathrm{C}_{2} \mathrm{Com}_{3}^{*}, \mathrm{C} . \mathrm{Prv}_{3}^{*}\right)$ against $\left\langle C_{3}, R_{3}\right\rangle$, the following holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} \mathrm{Com}_{3}^{*}\right.$, R.Com $\left.{ }_{3}\right\rangle$.

- Soundness Condition: If it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{3} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{3}^{*}\left(\mathrm{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{3}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array} \tag{G.3}
\end{array}\right] \geq \frac{1}{\lambda^{c}},
$$

then there exists $x_{\text {сом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that

$$
\begin{equation*}
\forall i \in[n], \operatorname{Pr}\left[x_{i}=x_{i}^{*} \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{3}^{\mathrm{C} \cdot \operatorname{Prv}_{3}^{*}\left(\mathrm{st}_{c}, \cdot\right)}(\operatorname{com}, i), \mathrm{R}^{2} \cdot \operatorname{Dec}_{3}(\operatorname{com}, i)\right\rangle\right] \geq 1-\operatorname{negl}(\lambda) \tag{G.4}
\end{equation*}
$$

and

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{3} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{3}^{*}\left(\operatorname{st}_{C}, Q\right)  \tag{G.5}\\
\wedge f\left(x_{c о м}^{*}\right)=0 & b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{3}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \leq \operatorname{negl}(\lambda) .
$$

```
Algorithm 14 Extractor \(E_{3}\) (against \(\left\langle C_{3}, R_{3}\right\rangle\) )
Input: com, \(i\), and \(\left\{Q^{\mu}\right\}_{\mu \in[M]} \subset D\left(X_{1, \text { in }}^{1}\right) \times \cdots \times D\left(X_{1, \text { in }}^{M}\right)\)
    1. Run \(\left\{\tilde{Y}^{\mu}\right\}_{\mu \in[M]} \leftarrow E_{2}^{\mathrm{C} \cdot \operatorname{Prv}_{2}^{*}\left(\operatorname{sta}_{C}, *\right)}\left(\operatorname{com}, i,\left\{Q^{\mu}\right\}_{\mu \in[M]}\right)\), where C.Prv\({ }_{2}^{*}\left(\operatorname{st}_{C}, \cdot\right):=\operatorname{Amplify}_{c}^{\left.\mathrm{C} \cdot \operatorname{Prv}_{3}^{*} \operatorname{stc}_{C}, \cdot\right)}(\cdot)\).
    2. Output \(\left\{\tilde{Y}^{\mu}\right\}_{\mu \in[M]}\) as the decommitment.
```

Proof. Fix any $c$ and $C_{3}^{*}=\left(\mathrm{C} . \mathrm{Com}_{3}^{*}\right.$, C.Prv $\left.{ }_{3}^{*}\right)$ as in the lemma statement, and assume for contradiction that for infinitely many $\lambda \in \mathbb{N}$, the soundness condition does not hold with non-negligible probability over the choice of (st ${ }_{C}$, com) $\leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{3}^{*}, \mathrm{R} . \mathrm{Com}_{3}\right\rangle$. Let us say that any $\lambda$ and ( $\mathrm{st}_{C}, \mathrm{com}$ ) are $\operatorname{good}$ (for $C_{3}^{*}$ ) if the soundness condition does not hold on $\lambda$ and ( $\mathrm{st}_{C}, \mathrm{com}$ ). Then, from the binding property of $\left\langle C_{3}, R_{3}\right\rangle$, it follows that for every good $\lambda$ and (st ${ }_{C}$, com), we have Equation (G.3) and also have

- either there exists $i \in[n]$ such that

$$
\begin{equation*}
\operatorname{Pr}\left[x_{i}=\perp \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{3}^{\mathrm{C}_{3} \cdot \operatorname{Prv}_{3}^{*}\left(\operatorname{st} t_{C},\right)}(\operatorname{com}, i), \mathrm{R}^{2} \operatorname{Dec}_{3}(\operatorname{com}, i)\right\rangle\right] \geq \frac{1}{\operatorname{poly}(\lambda)}, \tag{G.6}
\end{equation*}
$$

- or there exists $x_{\text {сом }}^{*}=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that we have Equation (G.4), but we also have

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{3} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{3}^{*}\left(\mathrm{st}_{C}, Q\right)  \tag{G.7}\\
\wedge f\left(x_{\text {сом }}^{*}\right)=0 & b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{3}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \geq \frac{1}{\operatorname{poly}(\lambda)} .
$$

To obtain a contradiction, we consider the following cheating committer-prover $C_{2}^{*}=\left(\right.$ C.Com ${ }_{2}^{*}$, C.Prv ${ }_{2}^{*}$ ) against $\left\langle C_{2}, R_{2}\right\rangle$.

- C.Com ${ }_{2}^{*}$ is identical with C.Com ${ }_{3}^{*}$.
- C.Prv ${ }_{2}^{*}\left(\operatorname{st}_{C}, \cdot\right)$ is identical with Amplify ${ }_{c}^{\text {C.Prv }}{ }^{*}\left(\mathrm{st}_{C}, \cdot\right)(\cdot)$.

Note that, due to Lemma K. $4, C_{2}^{*}$ is $\kappa_{\text {prv }}-$ CNS.
We now show that $C^{*}$ breaks the soundness property of $\left\langle C_{2}, R_{2}\right\rangle$. Fix any good $\lambda$ and (st ${ }_{C}, \mathrm{com}$ ). First, since we have Equation (G.3), we have

$$
\begin{align*}
& \operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{2} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C}^{2} \cdot \operatorname{Prv}_{2}^{*}\left(\operatorname{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} . \operatorname{Prv} . \mathrm{D}_{2}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}
\end{array}\right] \\
& =\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left.\begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{2} ;\left(f, \pi^{*}\right) \leftarrow \operatorname{Amplify}_{c}^{\mathrm{C} \cdot \operatorname{Prv}_{3}^{*}\left(\mathrm{st}_{c,},\right)}(Q) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{2}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right]
\end{array}\right. \\
& \geq 1-\operatorname{negl}(\lambda) \quad(\text { from Lemma K.4). } \tag{G.8}
\end{align*}
$$

Next, consider the following two cases.

- Case 1. Assume that there exists $i \in[n]$ such that we have Equation (G.6). Then, from the constructions of R. $\mathrm{Dec}_{3}, \mathrm{C} . \operatorname{Prv}_{2}^{*}, E_{2}$, and $E_{3}$, we have

$$
\begin{aligned}
& \operatorname{Pr}\left[x_{i}=\perp \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{2}^{\mathrm{C} \cdot \operatorname{Prv}_{2}^{*}\left(\operatorname{stc}_{c},\right)}(\operatorname{com}, i), \mathrm{R}^{2} \operatorname{Dec}_{3}(\operatorname{com}, i)\right\rangle\right] \\
& =\operatorname{Pr}\left[x_{i}=\perp \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{3}^{\mathrm{C}_{3} \cdot \operatorname{Prv}_{3}^{*}\left(\operatorname{stc}_{C},\right)}(\mathrm{com}, i), \operatorname{R.Dec}_{3}(\operatorname{com}, i)\right\rangle\right] \\
& \geq \frac{1}{\operatorname{poly}(\lambda)}
\end{aligned}
$$

This contradicts with the observation that is made in Remark 7.

- Case 2. Assume that there exists $x_{\text {сом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that we have Equation (G.4), but we also have Equation (G.7). Then, from the furthermore part of Lemma K.4, we have

$$
\begin{aligned}
& \operatorname{Pr}\left[f\left(x_{\text {сом }}^{*}\right)=0 \mid \quad\left(Q, \mathrm{st}_{R}\right) \leftarrow \text { R.Prv. } \mathrm{Q}_{2} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{2}^{*}\left(\mathrm{st}_{C}, Q\right)\right] \\
& =\operatorname{Pr}\left[f\left(x_{\mathrm{CoM}}^{*}\right)=0 \mid \quad\left(Q, \mathrm{st}_{R}\right) \leftarrow \operatorname{R.Prv} \cdot \mathrm{Q}_{2} ;\left(f, \pi^{*}\right) \leftarrow \operatorname{Amplify}_{c}^{\mathrm{CPPr}_{3}^{*}\left(\mathrm{st}_{C}, \cdot\right)}(Q)\right] \\
& =\operatorname{Pr}\left[f\left(x_{\text {com }}^{*}\right)=0 \mid \quad\left(Q, \mathrm{st}_{R}\right) \leftarrow \operatorname{R.Prv} . Q_{3} ;\left(f, \pi^{*}\right) \leftarrow \operatorname{Amplify}_{c}^{\mathrm{CPPrv}_{3}^{*}\left(\mathrm{st}_{C},\right)}(Q)\right] \\
& \geq \operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv}^{2} \mathrm{Q}_{3} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{3}^{*}\left(\mathrm{st}_{C}, Q\right) \\
\wedge f\left(x_{\mathrm{coM}}^{*}\right)=0 & b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{3}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right]-\operatorname{negl}(\lambda) \\
& \geq \frac{1}{\operatorname{poly}(\lambda)} \quad \text { (from Equation (G.7)). }
\end{aligned}
$$

From a union bound with Equation (G.8), we obtain

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 \wedge f\left(x_{\mathrm{com}}^{*}\right)=0 & \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{2} \\
\left(f, \pi^{*}\right) \leftarrow \operatorname{CPrv}_{2}^{*}\left(\mathrm{st}_{C}, Q\right) \\
b \leftarrow \operatorname{R.Prv} \cdot \mathrm{D}_{2}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}
\end{array}\right] \geq \frac{1}{\operatorname{poly}(\lambda)}
$$

and thus $C_{2}^{*}$ breaks soundness of $\left\langle C_{2}, R_{2}\right\rangle$.
By combining what are shown above, we conclude that $C_{2}^{*}$ breaks the soundness of $\left\langle C_{2}, R_{2}\right\rangle$.
By combining all the above, we obtain a contradiction. This concludes the proof of Lemma G.2.

## H Step 4: Non-WI Scheme with Standard negl-Soundness

As the fourth step to our commit-and-prove protocol, we give a non-WI commit-and-prove protocol $\left\langle C_{4}, R_{4}\right\rangle$ that is neglsound against (not necessarily CNS) provers. Additionally, we also show that $\left\langle C_{4}, R_{4}\right\rangle$ satisfies "2-privacy" in a similar sense to the MPC protocol $\Pi$. In this step, we use a 2-round PIR protocol PIR $=$ (PIR.Enc, PIR.Res, PIR.Dec).

## H. 1 Preliminaries

In $\left\langle C_{4}, R_{4}\right\rangle$, we use PIR for the receiver to encrypt its PCP queries as in Kalai et al. [KRR14]. To simplify the exposition of $\left\langle C_{4}, R_{4}\right\rangle$, we introduce a triple of algorithms, (PIR.EncSet, PIR.ResSet, PIR.DecSet), that use PIR for this purpose. Let $\kappa$ be a polynomial and $Q \subseteq[N]$ be a subset of the indices of a database $D B \in \Sigma^{N}$.

- $\left(\mathbb{Q}, \mathrm{st}_{\text {PIR }}\right) \leftarrow \operatorname{PIR.EncSet}_{k(\lambda)}\left(1^{\lambda}, Q, N\right)$ :

1. Choose a random injective function $\tau: Q \rightarrow[\kappa(\lambda)]$.
2. For each $i \in[\kappa(\lambda)]$, run

$$
\left(\mathbb{G}_{i}, \mathrm{St}_{\mathrm{PIR}}^{i}\right) \leftarrow \begin{cases}\operatorname{PIR} \cdot \operatorname{Enc}\left(1^{\lambda}, q, N\right) & \text { if } \exists q \in Q \text { s.t. } \tau(q)=i \\ \operatorname{PIR} \cdot \operatorname{Enc}\left(1^{\lambda}, 1, N\right) & \text { otherwise }\end{cases}
$$

3. Output $\mathbb{Q}:=\left\{\mathrm{q}_{i}\right\}_{i \in[\kappa(\lambda)]}$ and $s t_{\mathrm{PIR}}:=\left\{\mathrm{s}_{\mathrm{PIR}}^{i}\right\}_{i \in[\kappa(\lambda)]}$.

- $\mathcal{K} \leftarrow \operatorname{PIR} . \operatorname{ResSet}\left(1^{\lambda}, \mathbb{Q}, D B\right)$

1. For each $i \in[\kappa(\lambda)]$, run $x_{i} \leftarrow \operatorname{PIR} \cdot \operatorname{Res}\left(1^{\lambda}, \mathbb{q}_{i}, D B\right)$
2. Output $\mathcal{K}:=\left\{x_{i}\right\}_{i \in[\kappa(\lambda)]}$.

- $X \leftarrow$ PIR.DecSet( $\left.\mathrm{st}_{\mathrm{PIR}}, \mathcal{X}\right)$

1. For each $q \in Q$, run $x_{\tau(q)} \leftarrow \operatorname{PIR} . \operatorname{Dec}\left(\operatorname{st}_{\mathrm{PIR}}^{\tau(q)}, x_{\tau(q)}\right)$.
2. Output $X:=\left\{x_{\tau(q)}\right\}_{q \in Q}$.

## H. 2 Protocol Description

The formal description of $\left\langle C_{4}, R_{4}\right\rangle$ is given in Algorithm 15, where polynomials $\kappa_{\mathrm{dec}}^{\prime}, \kappa_{\mathrm{prv}}^{\prime}$ are defined as $\kappa_{\mathrm{dec}}^{\prime}=\kappa_{\mathrm{dec}}+\lambda \kappa_{\mathrm{v}}$ and $\kappa_{\mathrm{prv}}^{\prime}=\kappa_{\mathrm{prv}}+\lambda \kappa_{\mathrm{v}}$, where $\kappa_{\mathrm{dec}}$ is the polynomial that is given in Lemma E.1, $\kappa_{\mathrm{prv}}$ is the polynomial that is given in Lemma E.3, and $\kappa_{\mathrm{v}}$ is the query complexity of (PCP.P, PCP.V).

The communication complexity of $\left\langle C_{4}, R_{4}\right\rangle$ can be bounded by a polynomial in $\kappa_{\text {dec }}, \kappa_{\mathrm{prv}}$, and $\kappa_{\mathrm{v}}$, all of which can be bounded by a polynomial in $\lambda$ and $\log (\operatorname{Time}(f))$ (cf. Footnote 22, Footnote 24, Lemma D.2).

## H. 3 Proof of Binding

Lemma H.1. Let $C_{4}^{*}=\left(\mathrm{C}_{4} \mathrm{Com}_{4}^{*}, \mathrm{C} . \mathrm{Dec}_{4}^{*}\right)$ be any PPT cheating committer-decommitter against $\left\langle C_{4}, R_{4}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following binding condition holds with overwhelming probability over the choice of ( $\left.\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow$ $\left\langle\mathrm{C} . \mathrm{Com}_{4}^{*}\right.$, R.Com $\left._{4}\right\rangle$.

- Binding Condition: For every $i \in[n]$, it holds $\operatorname{Pr}\left[b_{B A D}=1\right] \leq \operatorname{negl}(\lambda)$ in the following probabilistic experiment $E x P_{4}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}_{4}^{*}, \mathrm{st}_{C}, \mathrm{com}, i\right)$.

1. For each $\forall b \in\{0,1\}$, sample $\left\{\mathbb{Q}_{b}^{\mu}\right\}_{\mu \in[M]}$ by $\left(\left\{\mathbb{Q}_{b}^{\mu}\right\}_{\mu \in[M]}, \mathrm{st}_{b}\right) \leftarrow$ R.Dec. $\mathbb{Q}_{4}(i)$.
2. Run $\left\{\tilde{\mathbb{Y}}^{*}{ }_{b}^{\mu: \mu}\right\}_{b \in\{0,1\}, \mu \in[M]} \leftarrow \operatorname{C.Dec}_{4}^{*}\left(\mathrm{St}_{C},\left\{\mathbb{Q}_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$.
3. Let $b_{B A D}:=1$ if and only if $x_{0}^{*} \neq \perp \wedge x_{1}^{*} \neq \perp \wedge x_{0}^{*} \neq x_{1}^{*}$ holds, where $x_{b}^{*}:=$ R.Dec. $\mathrm{D}_{4}\left(\mathrm{st}_{b}, \operatorname{com},\left\{\tilde{\Psi}^{*}{ }_{b}^{\mu \mu}\right\}_{\mu \in[M]}\right)$ for each $b \in\{0,1\}$.

Proof. Fix any $C_{4}^{*}=\left(\mathrm{C} . \mathrm{Com}_{4}^{*}, \mathrm{C}_{\mathrm{C}} \mathrm{Dec}_{4}^{*}\right)$ as above, and assume for contradiction that for infinitely many $\lambda \in \mathbb{N}$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} . \mathrm{Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$, there exists $i \in[n]$ such that we have $\operatorname{Pr}\left[b_{\text {BAD }}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in the experiment $\operatorname{ExP}_{4}^{\text {bind }}\left(\right.$ C. $_{\text {Dec }}^{4}$, , $_{C}$, com, $\left.i\right)$.

To obtain a contradiction, we define a cheating committer-decommitter $C_{3}^{*}=\left(\right.$ C.Com $_{3}^{*}$, C.Dec $\left._{3}^{*}\right)$ against $\left\langle C_{3}, R_{3}\right\rangle$ by using $C_{4}^{*}$, and show that $C_{3}^{*}$ breaks the binding property of $\left\langle C_{3}, R_{3}\right\rangle$. Specifically, consider the following $C_{3}^{*}=$ (C.Com ${ }_{3}^{*}$, C.Dec ${ }_{3}^{*}$ ).

- Committer: C.Com ${ }_{3}^{*}$ is identical with C.Com ${ }_{4}^{*}$.
- Decommitter: Given $\mathrm{st}_{C}$ and $\left\{Q^{\mu}\right\}_{\mu \in[M]}$ as input, C. $^{\text {Dec }}{ }_{3}^{*}$ does the following.

1. Run $\left(\mathbb{Q}^{\mu}\right.$, st $\left._{\text {PIR }}^{\mu}\right) \leftarrow$ PIR.EncSet $_{\kappa_{\text {dec }}^{\prime}}\left(Q^{\mu},\left|\pi^{\mu: v}\right|\right)$ for each $\mu \in[M]$.
2. Run ${\widetilde{Y^{*}}}^{\mu / \mu} \leftarrow \mathrm{C} . \operatorname{Dec}_{4}^{*}\left(\operatorname{St}_{C},\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}\right)$.
3. Run $\tilde{Y}^{*} \mu^{\mu} \leftarrow$ PIR.DecSet(St $\left.{ }_{\text {PIR }}^{\mu}, \tilde{Y}^{\mu^{\prime}}\right)$ for each $\mu \in[M]$.
4. Output $\left\{\tilde{Y}^{*}{ }^{\mu \cdot \mu}\right\}_{\mu \in[M]}$ as the decommitment.

It is straightforward to see that for every $\lambda$, (st $C_{C}, \mathrm{com}$ ), and $i$ such that we have $\operatorname{Pr}\left[b_{\mathrm{BAD}}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in the experiment $\operatorname{Exp}_{4}^{\text {bind }}\left(\mathrm{C}_{\mathrm{D}} \operatorname{Dec}_{4}^{*}, \mathrm{st}_{C}\right)$, we have $\operatorname{Pr}\left[b_{\text {BAD }}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in the experiment $\operatorname{Exp}_{3}^{\text {bind }}\left(\mathrm{C} . \operatorname{Dec}_{3}^{*}, \mathrm{st}_{C}\right)$. (To see this, observe that C.Dec ${ }_{3}^{*}$ perfectly emulates $R . \mathrm{Dec}_{4}$ for the internally emulated $\mathrm{C} . \mathrm{Dec}_{4}^{*}$ ). Furthermore, it is relatively easy to see that the security of PIR implies the CNS of C.Dec.*. (The proof of this fact is virtually the same as the proof of, e.g., [KRR13, Theorem 12] and [BHK16, Claim 7].) Hence, $C_{3}^{*}$ breaks the binding property of $\left\langle C_{3}, R_{3}\right\rangle$, and we obtain a contradiction.

```
Algorithm 15 Commit Phase, Open Phase, and Prove Phase of \(\left\langle C_{4}, R_{4}\right\rangle\)
Commit Phase
The commit phase of \(\left\langle C_{4}, R_{4}\right\rangle\) is identical with that of \(\left\langle C_{3}, R_{3}\right\rangle\).
```


## Open Phase

Round 1: Given $i$ as input, R.Dec. Q $_{4}$ does the following.

1. $\operatorname{Run}\left(\left\{Q^{\mu}\right\}_{\mu \in[M]}, \mathrm{st}_{R}^{\prime}\right) \leftarrow$ R.Dec. $\mathrm{Q}_{3}(i)$.
2. Run $\left(\mathbb{Q}^{\mu}, \operatorname{st}_{\mathrm{PIR}}^{\mu}\right) \leftarrow$ PIR.EncSet $_{\kappa_{\text {dec }}^{\prime}}\left(Q^{\mu},\left|\pi^{\mu: v}\right|\right)$ for each $\mu \in[M]$. (Here, $\left|\pi^{\mu: v}\right|$ denotes the length of the PCP proofs that are computed by the prover in the prove phase).
3. Output $\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}$ as the query and $\left(\mathrm{st}_{R}^{\prime},\left\{\mathrm{St}_{\mathrm{PIR}}^{\mu}\right\}_{\mu \in[M]}\right)$ as the internal state.

Round 2: Given (st $C_{C},\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}$ ) as input, C.Dec ${ }_{4}$ does the following.

1. Define $\tilde{Y}^{\mu ; \mu}$ just like C. Dec $_{3}$ does.
2. Run $\tilde{\mathbb{Y}}^{\mu: \mu} \leftarrow \mathrm{PIR}$.ResSet $\left(\mathbb{Q}^{\mu}, \tilde{Y}^{\mu ; \mu}\right)$ for each $\mu \in[M]$.
3. Output $\left\{\tilde{\mathbb{Y}}^{\mu \mu \mu}\right\}_{\mu \in[M]}$ as the decommitment.

Verification: Given ( $\mathrm{st}_{R}$, com, $\left.\left\{\tilde{\mathbb{Y}}^{\mu \mu \mu}\right\}_{\mu \in[M]}\right)$ as input (where $\mathrm{st}_{R}=\left(\mathrm{st}_{R}^{\prime},\left\{\mathrm{st}_{\mathrm{PIR}}^{\mu}\right\}_{\mu \in[M]}\right)$, R.Dec ${ }_{4}$ does the following.

1. Run $\tilde{Y}^{\mu: \mu} \leftarrow$ PIR.DecSet( $\left.\mathrm{St}_{\mathrm{PIR}}^{\mu}, \tilde{\mathbb{Y}}^{\mu: \mu}\right)$ for each $\mu \in[M]$.
2. Output $\tilde{x}_{i}:=$ R.Dec. $\mathrm{D}_{3}\left(\mathrm{st}_{R}^{\prime}, \operatorname{com},\left\{\tilde{Y}^{\mu \mu \mu}\right\}_{\mu \in[M]}\right)$ as the decommitted value.

## Prove Phase

Round 1: R. Prv $_{4}$ does the following.

1. Run $\left(\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}, \mathrm{st}_{R}^{\prime}\right) \leftarrow$ R.Prv. $\mathrm{Q}_{3}$.
2. Run $\left(\mathbb{Q}^{\mu: \nu}, \mathrm{St}_{\mathrm{PIR}}^{\mu: \nu}\right) \leftarrow \operatorname{PIR}^{2}$ EncSet $_{\mathrm{Kprv}^{\prime}+\kappa_{\mathrm{dec}}^{\prime}}\left(Q^{\mu: v},\left|\pi^{\mu: v}\right|\right)$ for each $\mu, v \in[M]$.
3. Output $\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}$ as the query and $\left(\mathrm{st}_{R}^{\prime},\left\{\mathrm{S}_{\mathrm{PIR}}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$ as the internal state.

Round 2: Given (st ${ }_{C}, f,\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}$ ) as input, C.Prv ${ }_{4}$ does the following.

1. Obtain $\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]}$ and $\left\{\pi^{\mu: v}\right\}_{\mu, v \in[M]}$ just like C. Prv $_{3}$ does.
2. Run $\mathbb{\pi}^{\mu: \nu} \leftarrow \operatorname{PIR}$.ResSet $\left(\mathbb{Q}^{\mu: \nu}, \pi^{\mu: v}\right)$ for each $\mu, v \in[M]$.
3. Output $\left(\left\{\mathrm{r}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{T}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$ as the proof.

Verification: Given (st, com, $\left.f,\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{T}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$ as input (where $\mathrm{st}_{R}=\left(\mathrm{st}_{R}^{\prime},\left\{\mathrm{St}_{\mathrm{PIR}}^{\mu, v}\right\}_{\mu, v \in[M]}\right)$ ), R. $\operatorname{Prv}_{4}$ does the following.

1. $\operatorname{Run} \pi^{* \mu: v} \leftarrow$ PIR. $\operatorname{DecSet}\left(\mathrm{St}_{\mathrm{PIR}}^{\mu: \nu}, \pi^{* \mu: v}\right)$ for each $\mu, v \in[M]$.
2. Run $b \leftarrow$ R.Prv. $\mathrm{D}_{3}\left(\mathrm{st}_{R}^{\prime}\right.$, com, $\left.f,\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\pi^{* \mu: v}\right\}_{\mu, v \in[M]}\right)$. That is, do the following.
(a) Run $b^{\mu: v} \leftarrow \operatorname{PCP} . \mathrm{D}^{\otimes \lambda}\left(\mathrm{st}_{V}^{\mu: \nu}, f^{\prime}, \mathrm{rt}_{1, \mathrm{n}}^{\mu}, \mathrm{rt}_{1, \mathrm{n}}^{v}, \mathrm{rt}^{\mu}, \mathrm{rt}^{\nu}, \pi^{* \mu: v}\right)$ for every $\mu, v \in[M]$
(b) Let $b:=1$ if and only if $b^{\mu: \nu}=1$ for every $\mu, v \in[M]$.
3. Output $b$.

## H. 4 Proof of 2-Privacy

Lemma H.2. For any PPT cheating receiver $R^{*}$, there exists a PPT simulator $\mathcal{S}_{4}$ such that for $\forall \lambda \in \mathbb{N}, \forall x_{\text {com }} \in\{0,1\}^{n}$ (where $n=\operatorname{poly}(\lambda)), \forall \alpha, \beta \in[M]$, and $\forall z \in\{0,1\}^{*}$, the outputs of the following two experiments are identically distributed.

## Real Experiment:

1. $\left(\mathrm{hf}, \mathrm{st}_{R}\right) \leftarrow R^{*}\left(x_{\text {Сом }}, z\right)$.
2. $\left(\left\{\mathrm{rr}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}, \mathrm{st}_{C}\right) \leftarrow \mathrm{C.Com}_{4}\left(x_{\text {CoM }}, \mathrm{hf}\right)$.
3. $\left(f,\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, v \in[M]}, \mathrm{st}_{R}^{\prime}\right) \leftarrow R^{*}\left(\mathrm{st}_{R}\right)$. If $f\left(x_{\text {com }}\right) \neq 1$, the experiment aborts with output $\perp$.
4. $\left(\left\{\mathrm{rr}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{T r}^{\mu v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C.} \operatorname{Prv}_{4}\left(\mathrm{st}_{C}, f,\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$.
5. Output $\left(\mathrm{st}_{R}^{\prime}, \mathrm{rt}_{1, \mathrm{i}}^{\alpha}, \mathrm{rt}_{1, \mathrm{in}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \mathrm{w}^{\alpha ; \beta}\right)$.

## Ideal Experiment:

1. $\left(\mathrm{hf}, \mathrm{st}_{R}\right) \leftarrow R^{*}\left(x_{\text {COM }}, z\right)$.
2. $\left(f,\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}, \mathrm{st}_{R}^{\prime}\right) \leftarrow R^{*}\left(\mathrm{st}_{R}\right)$. If $f\left(x_{\text {CoM }}\right) \neq 1$, the experiment aborts with output $\perp$.
3. $\left(\mathrm{rt}_{1, \mathrm{i}}^{\alpha}, \mathrm{rt}_{1, \mathrm{i}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \mathrm{m}^{\alpha ; \beta}\right) \leftarrow \mathcal{S}_{4}\left(\alpha, \beta, f, \mathrm{hf}, \mathbb{Q}^{\alpha ; \beta}\right)$.
4. Output $\left(\mathrm{st}_{R}^{\prime}, \mathrm{rt}_{1, \mathrm{in}}^{\alpha}, \mathrm{rt}_{1, \mathrm{in}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \mathrm{w}^{\alpha ; \beta}\right)$.

Proof. First, we remark that from the constructions of C.Com 4 and C.Prv 4 , Step 2 and Step 4 of the real experiment can be written as follows.
2. Compute $\left(\left\{\mathrm{rt}_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}\right.$, st $\left._{C}\right)$, where $\mathrm{st}_{C}=\left(\mathrm{hf}_{\mathrm{f}}\left\{X_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}\right)$, as follows.
(a) Sample random $x_{\mathrm{MPC}}^{1}, \ldots, x_{\mathrm{MPC}}^{M} \in\{0,1\}^{n}$ such that $x_{\mathrm{MPC}}^{1} \oplus \cdots \oplus x_{\mathrm{MPC}}^{M}=x_{\text {Сом }}$.
(b) Compute $X_{1, \text { in }}^{\mu}$ and $\mathrm{rt}_{1, \text { in }}^{\mu}$ for $\forall \mu \in[M]$ as follows: sample random $r_{\text {MPC }}^{\mu} \in\{0,1\}^{n_{\text {MPC }}}$, let st ${ }_{0}^{\mu}:=x_{\text {MPC }}^{\mu} \| r_{\text {MPC }}^{\mu}$, let i-msgs ${ }_{1}^{\mu}:=0^{M}$, let $x_{1, \text { in }}^{\mu}:=\operatorname{st}_{0}^{\mu} \|$ i-msgs ${ }_{1}^{\mu}$, let $X_{1, \text { in }}^{\mu}$ be the low-degree extension of $x_{1, \mathrm{in}}^{\mu}$, and let $\mathrm{rt}_{1, \mathrm{in}}^{\mu}:=$ $\operatorname{TreeHash}_{\text {hf }}\left(X_{1, \text { in }}^{\mu}\right)$.
4. Compute $\left(\left\{\mathrm{r}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{\pi}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$ as follows.
(a) Run the MPC protocol $\Pi$ on $\left(f^{\prime},\left\{\mathrm{st}_{0}^{\mu} \text {, } \mathrm{i}-\mathrm{msgs}_{1}^{\mu}\right\}_{\mu \in[M]}\right)$, and let $\left\{\text { view }^{\mu}\right\}_{\mu \in[M]}$ be the view of the parties in this execution.
(b) Run $\left(\mathrm{rt}^{\mu}, \mathrm{rt}^{\nu}, \pi^{\mu: v}\right) \leftarrow \operatorname{PCP} \cdot \mathrm{P}^{\prime}\left(\mu, v, f^{\prime}, \mathrm{view}^{\mu}\right.$, view ${ }^{\nu}$, hf) for every $\mu, v \in[M]$.
(c) Run $\mathbb{\pi}^{\mu: v} \leftarrow \operatorname{PIR} . \operatorname{Res} \operatorname{Set}\left(\mathbb{Q}^{\mu: \nu}, \pi^{\mu: v}\right)$ for every $\mu, v \in[M]$.

Given the above description of the real experiment in mind, we consider the simulator $\mathcal{S}_{4}$ in Algorithm 16 (where we use the simulator $\mathcal{S}_{\mathrm{MPC}}$ of $\Pi$ that is guaranteed to exist due to its 2-privacy).

```
Algorithm 16 Simulator \(\mathcal{S}_{4}\) (against \(\left\langle C_{4}, R_{4}\right\rangle\) )
    1. Run \(\left\{\operatorname{view}^{\xi}\right\}_{\xi \in\{\alpha, \beta\}} \leftarrow \mathcal{S}_{\mathrm{MPC}}\left(\{\alpha, \beta\},\left\{x_{\mathrm{MPC}}^{\xi}\right\}_{\xi \in\{\alpha, \beta\}}, 1\right)\) for random \(x_{\mathrm{MPC}}^{\alpha}, x_{\mathrm{MPC}}^{\beta} \in\{0,1\}^{n}\). Let \(r_{\mathrm{MPC}}^{\xi}\), i-msgs \({ }_{0}^{\xi}\) be the randomness
        and the dummy incoming messages of the first round that are recorded in view \({ }^{\xi}\) for \(\forall \xi \in\{\alpha, \beta\}\).
    2. Compute \(\mathrm{rt}_{1, \mathrm{in}}^{\xi}\) for \(\xi \in\{\alpha, \beta\}\) as follows: let \(x_{1, \mathrm{in}}^{\xi}:=x_{\mathrm{MPC}}^{\xi}\left\|r_{\mathrm{MPC}}^{\xi}\right\| \mathrm{i}-\mathrm{msgs}_{1}^{\xi}\), let \(X_{1, \mathrm{in}}^{\xi}\) be the low-degree extension of \(x_{1, \mathrm{in}}^{\xi}\),
        and let \(\mathrm{rt}_{1, \mathrm{in}}^{\xi}:=\operatorname{TreeHash}_{\mathrm{hf}}\left(X_{1, \mathrm{in}}^{\xi}\right)\).
    3. Run \(\left(\mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \pi^{\alpha ; \beta}\right) \leftarrow \operatorname{PCP}^{\prime} \mathrm{P}^{\prime}\left(\alpha, \beta, f^{\prime}\right.\), view \(^{\alpha}\), view \({ }^{\beta}\), hf).
    4. Run \(\pi^{\alpha ; \beta} \leftarrow \operatorname{PIR}\).ResSet \(\left(\mathbb{Q}^{\alpha ; \beta}, \pi^{\alpha ; \beta}\right)\).
    5. Output \(\left(\mathrm{rt}_{1, \mathrm{i}}^{\alpha}, \mathrm{rt}_{1, \mathrm{i}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \mathrm{m}^{\alpha ; \beta}\right)\).
```


## H. 5 Proof of Soundness

Lemma H.3. Fix any constant $c \in \mathbb{N}$, and let $E_{4}$ be the extractor in Algorithm 17. Then, for any ppt cheating committerprover $C_{4}^{*}=\left(\mathrm{C} . \mathrm{Com}_{4}^{*}, \mathrm{C} . \operatorname{Prv}_{4}^{*}\right)$ against $\left\langle C_{4}, R_{4}\right\rangle$, the following condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C.Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$.

- Soundness Condition: If it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left.\begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{4} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{4}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \geq \frac{1}{\lambda^{c}}, ~ \tag{H.1}
\end{array}\right.
$$

then there exists $x_{\text {cом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that

$$
\begin{equation*}
\forall i \in[n], \operatorname{Pr}\left[x_{i}=x_{i}^{*} \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{4}^{\mathrm{C} \cdot \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}, \cdot\right)}(\operatorname{com}, i), \mathrm{R}^{2} \mathrm{Dec}_{4}(\operatorname{com}, i)\right\rangle\right] \geq 1-\operatorname{negl}(\lambda) \tag{H.2}
\end{equation*}
$$

and

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \mathrm{Q}_{4} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}, Q\right)  \tag{H.3}\\
\wedge f\left(x_{\text {com }}^{*}\right)=0 & b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{4}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \leq \operatorname{negl}(\lambda) .
$$

## Algorithm 17 Extractor $E_{4}$ (against $\left\langle C_{4}, R_{4}\right\rangle$ )

Input: com, $i$, and $\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}$.

1. Define $\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, v \in[M]}$ as follows: let $\mathbb{Q}^{\mu ; \mu}:=\mathbb{Q}^{\mu}$ for $\forall \mu \in[M]$, and let $\mathbb{Q}^{\mu: \nu}$ be sampled by ( $\left.\mathbb{Q}^{\mu: \nu}, \mathrm{St}_{\mathrm{PIR}}^{\mu: \nu}\right) \leftarrow$ PIR.EncSet ${ }_{\kappa_{\operatorname{dec}}^{\prime}}\left(\emptyset,\left|\pi^{\mu: v}\right|\right)$ for $\forall \mu, v \in[M]$ such that $\mu \neq v$.
2. For each $i \in\left[\lambda^{c+1}\right]$, do the following.
(a) Run $\left(\left\{\mathbb{Q}_{i}^{\mu: \nu}\right\}_{\mu, v \in[M]}\right.$, st $\left._{i}\right) \leftarrow$ R.Prv. $Q_{4}^{\prime}$, where R.Prv. $Q_{4}^{\prime}$ is identical with R.Prv. $Q_{4}$ except that PIR.EncSet ${ }_{\kappa_{\mathrm{prv}}}$ is used instead of PIR.EncSet $\kappa_{\kappa_{\mathrm{prv}}^{\prime}+\kappa_{\text {dec }}^{\prime}}$.
(b) For each $\mu, v \in[M]$, choose a random permutation $\tau:\left[\kappa_{\mathrm{prv}}^{\prime}+\kappa_{\mathrm{dec}}^{\prime}\right] \rightarrow\left[\kappa_{\mathrm{prv}}^{\prime}+\kappa_{\mathrm{dec}}^{\prime}\right]$ and define $\mathbb{S}_{i}^{\mu: v}=\left\{s_{j}\right\}_{j \in\left[\kappa_{\mathrm{prv}}^{\prime}+\kappa_{\mathrm{dec}}^{\prime}\right]}$ as follows: parse $\mathbb{Q}^{\mu: \nu}$ as $\left\{\mathbb{q}_{j}^{\mu: \nu}\right\}_{j \in\left[\kappa_{\text {dec }}^{\prime}\right]}$ and parse $\mathbb{Q}_{i}^{\mu: \nu}$ as $\left\{\mathbb{Q}_{i, j}^{\mu: \nu}\right\}_{j \in\left[K_{\text {prv }}^{\prime}\right]}$; then, each $s_{j}$ is defined as

$$
s_{j}:= \begin{cases}\mathbb{q}_{i, k}^{\mu: v} & \text { if } k:=\tau^{-1}(j) \text { satisfies } k \in\left\{1, \ldots, \kappa_{\text {prv }}^{\prime}\right\} \\ \mathbb{q}_{k}^{\mu: v} & \text { if } k:=\tau^{-1}(j) \text { satisfies } k \in\left\{\kappa_{\text {prv }}^{\prime}+1, \ldots, \kappa_{\text {prv }}^{\prime}+\kappa_{\text {dec }}^{\prime}\right\} .\end{cases}
$$

(c) $\operatorname{Run}\left(f_{i},\left\{\mathrm{rt}_{i}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{P}_{i}^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C.Prv} \mathrm{P}_{4}^{*}\left(\operatorname{st}_{C},\left\{\mathbb{S}_{i}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$, and parse each $\mathbb{P}_{i}^{* \mu: v}$ as $\left\{\mathbb{P}_{i, j}^{* \mu: v}\right\}_{j \in\left[K_{\max }\right]}$.

3. Find the first $i^{*} \in\left[\lambda^{c+1}\right]$ such that

$$
\text { R.Prv.D }{ }_{4}\left(\mathrm{st}_{i^{*}}, \operatorname{com}, f_{i^{*}},\left\{\mathrm{rr}_{i^{*}}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{\pi}_{i^{*}}^{* \mu \cdot v}\right\}_{\mu, v \in[M]}\right)=1 .
$$

If such $i^{*}$ does not exist, output $\perp$. Otherwise, send $\left\{\mathbb{Y}_{i^{*}}^{\mu ; \mu}\right\}_{\mu \in[M]}$ to R.Dec ${ }_{4}$.
Remark 8 (Intuition of $E_{4}$ ). Essentially, $E_{4}$ emulates an execution of $E_{3}$ while (1) inlining executions of Amplify ${ }_{c}$ (Algorithm 22 in Section K.2) and $E_{2}$, and (2) encrypting the queries to the cheating prover by PIR.

Proof . Fix any $c$ and $C_{4}^{*}=\left(\mathrm{C}_{\mathrm{Com}}^{4}\right.$, C.Prv $\left.{ }_{4}^{*}\right)$ as above, and assume for contradiction that for infinitely many $\lambda \in \mathbb{N}$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$, the soundness condition does not hold.

To obtain a contradiction, we define a cheating committer-prover $\left.C_{3}^{*}=\left(\mathrm{C} . \mathrm{Com}_{3}^{*}, \mathrm{C} . \operatorname{Prv}\right)_{3}^{*}\right)$ against $\left\langle C_{3}, R_{3}\right\rangle$ by using $C_{4}^{*}$, and show that $C_{3}^{*}$ breaks the soundness of $\left\langle C_{3}, R_{3}\right\rangle$. Specifically, consider the following $C_{3}^{*}=\left(\mathrm{C} . \mathrm{Com}_{3}^{*}, \mathrm{C} \cdot \mathrm{Prv}_{3}^{*}\right)$.

- Committer: C.Com ${ }_{3}^{*}$ is identical with C.Com ${ }_{4}^{*}$.
- Prover: Given $s t_{C}$ and $\left\{Q^{\mu: v}\right\}_{\mu, v \in[M]}$ as input, C.Prv ${ }_{3}^{*}$ does the following.

1. Run $\left(\mathbb{Q}^{\mu: v}, \mathrm{St}_{\mathrm{PIR}}^{\mu: \nu}\right) \leftarrow \operatorname{PIR}^{2}$ EncSet $_{\kappa_{\text {prv }}^{\prime}+\kappa_{\text {dec }}^{\prime}}\left(Q^{\mu: v},\left|\pi^{\mu: v}\right|\right)$ for each $\mu, v \in[M]$.
2. $\operatorname{Run}\left(f,\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{\pi}^{* \mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \operatorname{C} \cdot \operatorname{Prv}_{4}^{*}\left(\operatorname{st}_{C},\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$.
3. Run $\pi^{* \mu: v} \leftarrow$ PIR.DecSet $\left(\mathrm{St}_{\mathrm{PIR}}^{\mu: v}, \pi^{* \mu: v}\right)$ for each $\mu, v \in[M]$.
4. Output $\left(f,\left\{\mathrm{r}^{\mu}\right\}_{\mu \in[M]},\left\{\pi^{* \mu: \nu}\right\}_{\mu, v \in[M]}\right)$ as the proof.

By carefully comparing $E_{4}^{\mathrm{C} . \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}, \cdot\right)}$ and $E_{3}^{\mathrm{C} . \mathrm{Prv}_{3}^{*}\left(\mathrm{st}_{C}, \cdot\right)}$, one can see that for every $\lambda$ and ( $\mathrm{st}_{C}$, com) for which the soundness condition of $\left\langle C_{4}, R_{4}\right\rangle$ does not hold against $C_{4}^{*}$, the soundness condition of $\left\langle C_{3}, R_{3}\right\rangle$ does not hold against $C_{3}^{*}$ either. (To see this, observe that the execution of $E_{4}^{\mathrm{CPPrv}_{4}^{*}\left(\mathrm{st}_{C}, \cdot\right)}$ is perfectly emulated during the execution of $E_{3}^{\mathrm{C.Prv}}{ }^{*}\left(\mathrm{st}_{C}, \cdot\right)$. Furthermore, it is relatively easy to see that the security of PIR implies the CNS of C.Prv**. (The proof of this fact is virtually the same as the proof of, e.g., [KRR13, Theorem 12] and [BHK16, Claim 7].) Hence, $C_{3}^{*}$ breaks the soundness of $\left\langle C_{3}, R_{3}\right\rangle$, and we obtain a contradiction.

## I Step 5: WI Scheme with Standard $O$ (1)-Soundness

Finally, we give our WI commit-and-prove protocol $\left\langle C_{5}, R_{5}\right\rangle$. In this section, we use the following additional building blocks.

- Non-interactive statistically binding commitment scheme SBCom. (A non-interactive one is used for simplicity, and actually a 2-round one is sufficient. It is known that a 2-round statistically binding commitment scheme can be obtained from a one-way function, which in turn can be obtained from a collision-resistant hash function.)
- 2-round statistically hiding commitment scheme SHCom such that the first-round message from the receiver is a hash function. (Such a statistically hiding commitment scheme can be obtained from a collision-resistant hash function.)
- 2-round 1-out-of- $M^{2}$ OT protocol OT.

Also, we assume that all the building blocks (the above ones and the ones that are used in $\left\langle C_{4}, R_{4}\right\rangle$ ) are sub-exponentially secure, so that we can assume that (1) the committed value of a SBCom commitment can be extracted by brute force in a quasi-polynomial time $T_{\mathrm{SB}}$, and (2) the security of SHCom, OT , and $\left\langle C_{4}, R_{4}\right\rangle$ holds against poly $\left(T_{\mathrm{SB}}\right)$-time adversaries.

## I. 1 Protocol Description

The formal description of $\left\langle C_{5}, R_{5}\right\rangle$ is given in Algorithm 18. It is easy to see that the communication complexity of $\left\langle C_{5}, R_{5}\right\rangle$ is bounded by a polynomial in $\lambda$ and the communication complexity of $\left\langle C_{4}, R_{4}\right\rangle$, and hence is bounded by a polynomial in $\lambda$ and $\log (\operatorname{Time}(f))$.

## I. 2 Proof of Binding

Lemma I.1. Let $C_{5}^{*}=\left(\mathrm{C}_{5} \mathrm{Com}_{5}^{*}, \mathrm{C} . \mathrm{Dec}_{5}^{*}\right)$ be any poly $\left(T_{s B}\right)$-time cheating committer-decommitter against $\left\langle C_{5}, R_{5}\right\rangle$. Then, for every $\lambda \in \mathbb{N}$, the following binding condition holds with overwhelming probability over the choice of (st $\left.{ }_{C}, \mathrm{com}\right) \leftarrow$ $\left\langle\mathrm{C}^{-C_{0}}{ }_{5}^{*}\right.$, R.Com $\left.{ }_{5}\right\rangle$.

- Binding Condition: For every $i \in[n]$, it holds $\operatorname{Pr}\left[b_{B A D}=1\right] \leq \operatorname{negl}(\lambda)$ in the following probabilistic experiment $E x P_{5}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}_{5}^{*}, \mathrm{st}_{C}, \mathrm{com}, i\right)$.

1. For each $\forall b \in\{0,1\}$, sample $\left\{\mathbb{Q}_{b}^{\mu}\right\}_{\mu \in[M]}$ by $\left(\left\{\mathbb{Q}_{b}^{\mu}\right\}_{\mu \in[M]}\right.$, st $\left._{b}\right) \leftarrow$ R.Dec. $\mathbb{Q}_{5}(i)$.
2. Run $\left\{\operatorname{dec}_{b}\right\}_{b \in\{0,1\}} \leftarrow \operatorname{C.Dec}{ }_{5}^{*}\left(\operatorname{st}_{C},\left\{\mathbb{Q}_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$.
3. Let $b_{B A D}:=1$ if and only if $x_{0}^{*} \neq \perp \wedge x_{1}^{*} \neq \perp \wedge x_{0}^{*} \neq x_{1}^{*}$ holds, where $x_{b}^{*}:=$ R.Dec. $\mathrm{D}_{5}\left(\mathrm{st}_{b}, \mathrm{com}^{2}, \mathrm{dec}_{b}\right)$ for each $b \in\{0,1\}$.
 non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{\prime} \mathrm{Com}_{5}^{*}\right.$, R.Com $\left.{ }_{5}\right\rangle$, there exists $i \in[n]$ such that we have $\operatorname{Pr}\left[b_{\text {BAD }}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in the experiment $\operatorname{Exp}_{5}^{\text {bind }}\left(\right.$ C. $^{\text {Dec }}{ }_{5}^{*}$, st $_{C}$, com, $\left.i\right)$.

To obtain a contradiction, we define a cheating committer-decommitter $C_{4}^{*}=\left(\mathrm{C} . \mathrm{Com}_{4}^{*}, \mathrm{C} . \mathrm{Dec}_{4}^{*}\right)$ against $\left\langle C_{4}, R_{4}\right\rangle$ by using $C_{5}^{*}$, and show that $C_{4}^{*}$ breaks the binding property of $\left\langle C_{4}, R_{4}\right\rangle$.

Concretely, we define $C_{4}^{*}=\left(\right.$ C.Com $\left.{ }_{4}^{*}, \mathrm{C} . \mathrm{Dec}_{4}^{*}\right)$ as follows.

- Committer: Given hf as input, C.Com ${ }_{4}^{*}$ does the following.

1. Run $\left(\operatorname{com}^{\prime}, \mathrm{st}_{C}^{\prime}\right) \leftarrow \mathrm{C} . \mathrm{Com}_{5}^{*}(\mathrm{hf})$, and parse $\mathrm{com}^{\prime}$ as $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$.
2. Define com and $\mathrm{st}_{C}$ as follows.
(a) Sample $\left\{\mathbb{Q}_{b}^{\mu}\right\}_{\mu \in[M]}$ by $\left(\left\{\mathbb{Q}_{b}^{\mu}\right\}_{\mu \in[M]}\right.$, st $\left._{b}\right) \leftarrow$ R.Dec. $\mathbb{Q}_{5}(i)$ for $\forall b \in\{0,1\}$.
(b) Run $\left\{\operatorname{dec}_{b}\right\}_{b \in\{0,1\}} \leftarrow \operatorname{C.} . \operatorname{Dec}_{5}^{*}\left(\operatorname{st}_{C},\left\{\mathbb{Q}_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$, and parse $\operatorname{dec}_{0}$ as $\left(\left\{\tilde{\mathbb{Y}^{*}}{ }^{\mu: \mu}\right\}_{\mu \in[M]},\left\{\mathrm{rt}_{1, \mathrm{n}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$.
(c) Let com := $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$ and $\mathrm{st}_{C}:=\left(\mathrm{com}, \mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}\right)$ if $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ is a valid decommitment for opening $\mathrm{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\mu}$ for $\forall \mu \in[M]$, and let com $:=\perp$ and $\mathrm{st}_{C}:=\perp$ otherwise.
```
Algorithm 18 Commit Phase, Open Phase, and Prove Phase of \(\left\langle C_{5}, R_{5}\right\rangle\)
Commit Phase
```



```
Round 2: Given ( \(x_{\text {сом }}, \mathrm{hf}\) ) as input, C.Com 5 does the following.
```

1. Run $\left(\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}, \mathrm{st}_{C}^{\prime}\right) \leftarrow \operatorname{C.Com}_{4}\left(x_{\text {com }}, \mathrm{hf}\right)$.
2. Run $\operatorname{com}_{\mathrm{sH}}^{\mu} \leftarrow \operatorname{SHCom}_{\mathrm{hf}}\left(\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right)$ for every $\mu \in[M]$. Let $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ be the decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{r}_{1, \mathrm{in}}^{\mu}$.
3. Output $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ as the commitment and $\left(\mathrm{st}_{C}^{\prime},\left\{\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$ as the internal state.

## Open Phase

Round 1: Given $i$ as input, R.Dec. $Q_{5}$ runs $\left(\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}, \mathrm{st}_{R}\right) \leftarrow$ R.Dec. $\mathrm{Q}_{4}(i)$, and then outputs $\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}$ as the query and $\mathrm{St}_{R}$ as the internal state.
Round 2: Given (st $C_{C},\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}$ ) as input (where st ${ }_{C}=\left(\operatorname{st}_{C}^{\prime},\left\{\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$ ), C.Dec Cuns $_{5}\left\{\tilde{\mathbb{Y}}^{\mu / \mu}\right\}_{\mu \in[M]} \leftarrow$ C. $\operatorname{Dec}_{4}\left(\mathrm{st}_{C}^{\prime},\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}\right)$ and then outputs $\left(\left\{\tilde{\mathbb{Y}}^{\mu: \mu}\right\}_{\mu \in[M]},\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$ as the decommitment.

Verification: Given (st ${ }_{R}$, com, $\left\{\tilde{Y}^{*} \mu^{\mu \mu}\right\}_{\mu \in[M]},\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ ) as input (where com $=\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ ), R.Dec.D $\mathrm{D}_{5}$ does the following.

1. Check that each $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\mu}$.
2. Output $\tilde{x}_{i}:=$ R.Dec. $\mathrm{D}_{4}\left(\mathrm{st}_{R}, \operatorname{com}^{\prime},\left\{{\tilde{\mathbb{Y}^{*}}}^{\mu \cdot \mu}\right\}_{\mu \in[M]}\right)$ as the decommitted value, where com $:=\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$.

## Prove Phase

Round 1: R.Prv. $Q_{5}$ does the following.

1. Run $\left(\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, v \in[M]}, \mathrm{st}_{R}^{\prime}\right) \leftarrow$ R.Prv. $\mathrm{Q}_{4}$.
2. Choose random $\alpha, \beta \in[M]$, and run $\left(\mathrm{ot}_{1}, \mathrm{st}_{\mathrm{or}}\right) \leftarrow \mathrm{OT}_{1}\left(1^{\lambda},(\alpha, \beta)\right)$.
3. Output $\left(\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}\right)$ as the query and $\left(\mathrm{st}_{R}^{\prime}, \alpha, \beta, \mathrm{st}_{\mathrm{ot}}\right)$ as the internal state.

Round 2: Given ( $\left.\mathrm{st}_{C}, f,\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}\right)$ as input (where $\mathrm{st}_{C}=\left(\mathrm{st}_{C}^{\prime},\left\{\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$ ), $\mathrm{C} . \operatorname{Prv}_{5}$ does the following.

1. $\operatorname{Run}\left(\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{T}^{\mu: v}\right\}_{\mu, v \in[M]}\right) \leftarrow \mathrm{C}^{\operatorname{Prv}} \operatorname{Pr}_{4}\left(\mathrm{st}_{C}^{\prime}, f,\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$.
2. Run $\operatorname{com}_{\mathrm{sB}}^{\mu} \leftarrow \operatorname{SBCom}\left(\mathrm{rt}_{1, \mathrm{in}}^{\mu}\left\|\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\| \mathrm{r}^{\mu}\right)$ for each $\mu \in[M]$. Let dec $\mathrm{S}_{\mathrm{sB}}^{\mu}$ be the decommitment for opening $\operatorname{com}_{\mathrm{sB}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{i}}^{\mu}\left\|\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\| \mathrm{rt}^{\mu}$.
3. Run $\operatorname{com}_{\mathrm{SB}}^{\mu: \nu} \leftarrow \operatorname{SBCom}\left(\pi^{\mu: \nu}\right)$ for each $\mu, v \in[M]$. Let $\operatorname{deC}_{\mathrm{SB}}^{\mu: \nu}$ be the decommitment for opening $\operatorname{com}_{\mathrm{sB}}^{\mu: \nu}$ to $\pi^{\mu: \nu}$.
4. Run ot ${ }_{2} \leftarrow \mathrm{OT}_{2}\left(\mathrm{ot}_{1},\left\{\mathrm{msg}^{\mu: \nu}\right\}_{\mu, v \in[M]}\right)$, where $\mathrm{msg}^{\mu: v}:=\left(\mathrm{rt}_{1, \mathrm{i}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}, \mathrm{rt}^{\mu}, \operatorname{dec}_{\mathrm{sB}}^{\mu}, \mathrm{rt}_{1, \mathrm{in}}^{\nu}, \operatorname{dec}_{\mathrm{sH}}^{v}, \mathrm{rt}^{\nu}, \operatorname{dec}_{\mathrm{sB}}^{\nu}, \mathrm{T}^{\mu: \nu}, \operatorname{dec}_{\mathrm{sB}}^{\mu: v}\right)$.
5. Output $\left(\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: \nu}\right\}_{\mu, v \in[M]}\right.$, ot $\left.{ }_{2}\right)$ as the proof.

Verification Given (st $\left., \operatorname{com}, f,\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: \nu}\right\}_{\mu, \nu \in[M]}, \mathrm{ot}_{2}\right)$ as input (where $\mathrm{st}_{R}=\left(\mathrm{st}_{R}^{\prime}, \alpha, \beta, \mathrm{st}_{\mathrm{ot}}\right)$ and com $=$ $\left.\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$, R.Prv. $\mathrm{D}_{5}$ does the following.

1. Run $\mathrm{msg}^{\alpha ; \beta}:=\mathrm{OT}_{3}\left(\mathrm{st}_{\mathrm{ot}}, \mathrm{ot}_{2}\right)$, where $\mathrm{msg}^{\alpha ; \beta}:=\left(\mathrm{rt}_{1, \mathrm{in}}^{\alpha}, \mathrm{dec}_{\mathrm{sH}}^{\alpha}, \mathrm{rt}^{\alpha}, \operatorname{dec}_{\mathrm{sB}}^{\alpha}, \mathrm{rt}_{1, \mathrm{in}}^{\beta}, \operatorname{dec}_{\mathrm{sH}}^{\beta}, \mathrm{rt}^{\beta}, \mathrm{dec}_{\mathrm{sB}}^{\alpha}, \mathrm{w}^{\alpha ; \beta}, \operatorname{dec}_{\mathrm{sB}}^{\alpha ; \beta}\right)$.
2. Check that

- $\operatorname{dec}_{\mathrm{sH}}^{\xi}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\xi}$ to $\mathrm{r}_{1, \mathrm{in}}^{\xi}$ for $\forall \xi \in\{\alpha, \beta\}$,
- $\operatorname{dec}_{\mathrm{sB}}^{\xi}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sB}}^{\xi}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\xi}\left\|\operatorname{dec}_{\mathrm{sH}}^{\xi}\right\| \mathrm{r} \mathrm{t}^{\xi}$ for $\forall \xi \in\{\alpha, \beta\}$, and
- $\operatorname{dec}_{\mathrm{SB}}^{\alpha ; \beta}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sB}}^{\alpha ; \beta}$ to $\mathbb{\pi}^{\alpha ; \beta}$.

3. Run $\pi^{\alpha ; \beta} \leftarrow$ PIR.DecSet(St $\left.{ }_{\text {PIR }}^{\alpha ; \beta}, \pi^{\alpha ; \beta}\right)$.
4. Output $b^{\alpha ; \beta} \leftarrow$ PCP. $\mathrm{D}^{\prime \otimes \lambda}\left(\mathrm{st}_{V}^{\alpha ; \beta}, f^{\prime}, \mathrm{rt}_{1, \mathrm{in}}^{\alpha}, \mathrm{rt}_{1, \mathrm{in}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \pi^{* \alpha ; \beta}\right)$.
5. Output com as the commitment, and $\mathrm{st}_{C}$ as the internal state.

- Decommitter: Given st ${ }_{C}$ and $\left\{\mathbb{Q}_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}$ as input, C.Dec ${ }_{4}^{*}$ aborts if $\mathrm{st}_{C}=\perp$, and does the following otherwise.

1. Parse $\mathrm{st}_{C}$ as (com, $\mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}$ ), and parse com' as $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$.
2. Run $\left\{\operatorname{dec}_{b}^{\prime}\right\}_{b \in\{0,1\}} \leftarrow \operatorname{C.} \operatorname{Dec}_{5}^{*}\left(\mathrm{st}_{C}^{\prime},\left\{\mathbb{Q}_{b}^{\mu}\right\}_{b \in\{0,1\}, \mu \in[M]}\right)$.
3. For each $b \in\{0,1\}$, define $\operatorname{dec}_{b}$ as follows.
(a) Parse $\operatorname{dec}_{b}^{\prime}$ as $\left(\left\{\tilde{\mathbb{Y}}^{\mu / \mu}\right\}_{\mu \in[M]},\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$.
(b) Let $\operatorname{dec}_{b}:=\perp$ if (1) $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\mu}$ for $\forall \mu \in[M]$ but (2) $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]} \neq$ com. Let $\operatorname{dec}_{b}:=\left\{{\tilde{\Psi^{*}}}^{\mu: \mu}\right\}_{\mu \in[M]}$ otherwise.
4. Output $\left\{\operatorname{dec}_{b}\right\}_{b \in\{0,1\}}$ as the decommitments.

Now, we show that $C_{4}^{*}$ breaks the binding property of $\left\langle C_{4}, R_{4}\right\rangle$. First, note that the binding property of SHCom implies that for $\forall \lambda \in \mathbb{N}$, with overwhelming probability over the choice of (st $\left.{ }_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{\prime} \mathrm{Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$, we have that for $\forall i \in[n]$,

$$
\operatorname{Pr}\left[\operatorname{st}_{C} \neq \perp \wedge\left(\exists b \in\{0,1\} \text { s.t. } \operatorname{dec}_{b}=\perp\right)\right] \leq \operatorname{neg} \mid(\lambda)
$$

in $\operatorname{Exp}_{4}^{\text {bind }}\left(\mathrm{C} . \mathrm{Dec}_{4}^{*}, \mathrm{St}_{C}\right.$, com, $\left.i\right)$. Given this, one can easily see that from the assumption that $C_{5}^{*}$ breaks the binding property of $\left\langle C_{5}, R_{5}\right\rangle$, it follows that for $\forall \lambda \in \mathbb{N}$, with non-negligible probability over the choice of (st ${ }_{C}$, com) $\leftarrow\left\langle\right.$ C.Com $_{4}^{*}$, R.Com $\left.{ }_{4}\right\rangle$, there exists $i \in[n]$ such that we have $\operatorname{Pr}\left[b_{\text {bAD }}=1\right] \geq 1 / \operatorname{poly}(\lambda)$ in the experiment $\operatorname{Exp}_{4}^{\text {bind }}\left(\mathrm{C}^{2} . \mathrm{Dec}_{4}^{*}\right.$, st ${ }_{C}$, com, $\left.i\right)$. Hence, $C_{4}^{*}$ breaks the binding property of $\left\langle C_{4}, R_{4}\right\rangle$, and we obtain a contradiction.

## I. 3 Proof of Witness Indistinguishability

Lemma I.2. $\left\langle C_{5}, R_{5}\right\rangle$ is witness indistinguishable.
Proof. We show the witness indistinguishability by showing that there exists a super-polynomial-time simulator that can simulate the receiver's view in Experiment $b$ (Definition 4) for $\forall b \in\{0,1\}$ without knowing the value of $b$. Fix any cheating receiver $R^{*}$, and consider the simulator $\mathcal{S}_{5}$ in Algorithm 19 (where we use the simulator $\mathcal{S}_{4}$ of $\left\langle C_{4}, R_{4}\right\rangle$ that is guaranteed to exist due to its 2-privacy).

```
Algorithm 19 Simulator \(\mathcal{S}_{5}\)
Commit Phase. Given hf from \(R^{*}\), do the following.
```

1. Simulate $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ by committing to all-zero strings by using SHCom.
2. Send $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ to $R^{*}$.

Prove Phase. Given $\left(f,\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}\right)$ from $R^{*}$, does the following.

1. Extract the receiver's choice $(\alpha, \beta)$ by $(\alpha, \beta):=\operatorname{Ext}_{\mathrm{OT}}\left(\mathrm{ot}_{1}\right)$. (Note: This step requires super-polynomial time; cf. Definition 13.)
2. Simulate $\operatorname{com}_{\mathrm{sB}}^{\alpha}, \operatorname{com}_{\mathrm{sB}}^{\beta}$, and $\operatorname{com}_{\mathrm{sB}}^{\alpha ; \beta}$ as follows.
(a) $\operatorname{Run}\left(\mathrm{rt}_{1, \mathrm{n}}^{\alpha}, \mathrm{rt}_{1, \mathrm{n}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \mathrm{w}^{\alpha ; \beta}\right) \leftarrow \mathcal{S}_{4}\left(\alpha, \beta, f, \mathrm{hf}, \mathbb{Q}^{\alpha: \beta}\right)$.
(b) For each $\xi \in\{\alpha, \beta\}$, compute (by brute force) a decommitment dec ${ }_{\mathrm{sH}}^{\xi}$ for opening $\operatorname{com}_{\mathrm{sH}}^{\xi}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\xi}$.
(c) For each $\xi \in\{\alpha, \beta\}$, obtain $\operatorname{com}_{\mathrm{sB}}^{\xi}$ by $\operatorname{com}_{\mathrm{sB}}^{\xi} \leftarrow \operatorname{SBCom}\left(\mathrm{rt}_{1, \mathrm{in}}^{\xi}\left\|\mathrm{dec}_{\mathrm{SH}}^{\xi}\right\| \mathrm{rt}^{\xi}\right)$.
(d) Obtain $\operatorname{com}_{\mathrm{sB}}^{\alpha ; \beta}$ by $\operatorname{com}_{\mathrm{sB}}^{\alpha ; \beta} \leftarrow \mathrm{SBCom}\left(\mathbb{T}^{\alpha ; \beta}\right)$.
3. Simulate $\operatorname{com}_{\mathrm{sB}}^{\mu}, \operatorname{com}_{\mathrm{sB}}^{\nu}$, and $\operatorname{com}_{\mathrm{SB}}^{\mu: v}$ for $\forall(\mu, v) \neq(\alpha, \beta)$ by committing to all-zero strings by using SBCom.
4. Simulate ot ${ }_{2}$ by ot ${ }_{2} \leftarrow \mathrm{OT}_{2}\left(\mathrm{ot}_{1},\left\{\mathrm{msg}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$, where

$$
\mathrm{msg}^{\alpha ; \beta}:=\left(\mathrm{rt}_{1, \mathrm{in}}^{\alpha}, \operatorname{dec}_{\mathrm{sH}}^{\alpha}, \mathrm{rt}^{\alpha}, \operatorname{dec}_{\mathrm{sB}}^{\alpha}, \mathrm{rt}_{1, \mathrm{i}}^{\beta}, \operatorname{dec}_{\mathrm{sH}}^{\beta}, \mathrm{rt}^{\beta}, \operatorname{dec}_{\mathrm{sB}}^{\beta}, \mathrm{ur}^{\alpha ; \beta}, \operatorname{dec}_{\mathrm{sB}}^{\alpha ; \beta}\right),
$$

and $\mathrm{msg}^{\mu: v}$ for $\forall(\mu, v) \neq(\alpha, \beta)$ is an all-zero string.
5. Output $\left(f,\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: \nu}\right\}_{\mu, v \in[M]}\right.$, ot $\left._{2}\right)$ as the proof.

Now, we consider a sequence of hybrid experiments.

- Hybrid $H_{0}$ is identical with the real experiment.
- Hybrid $H_{1}$ is identical with $H_{0}$ except that in the prove phase, (1) the receiver's choice $(\alpha, \beta)$ is extracted by $(\alpha, \beta):=$ Extot $\left(\mathrm{ot}_{1}\right)$, and (2) $\mathrm{msg}^{\mu: v}$ for $\forall(\mu, v) \neq(\alpha, \beta)$ is an all-zero string.
- Hybrid $H_{2}$ is identical with $H_{1}$ except that in the prove phase, $\operatorname{com}_{\mathrm{sB}}^{\mu}$, $\operatorname{com}_{\mathrm{sB}}^{\nu}$, and $\operatorname{com}_{\mathrm{sB}}^{\mu \cdot v}$ for $\forall(\mu, v) \neq(\alpha, \beta)$ are generated by committing to all-zero strings by using SBCom.
- Hybrid $H_{3}$ is identical with $H_{2}$ except that (1) in the commit phase, $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ is generate by committing to all-zero strings by using SHCom, and (2) in the prove phase, $\mathrm{com}_{\mathrm{sB}}^{\alpha}$ and $\mathrm{com}_{\mathrm{sB}}^{\beta}$ are generated as follows.

1. For each $\xi \in\{\alpha, \beta\}$, compute (by brute force) a decommitment $\mathrm{dec}_{\mathrm{sH}}^{\xi}$ for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\xi}$.
2. For each $\xi \in\{\alpha, \beta\}$, obtain $\operatorname{com}_{\mathrm{sB}}^{\xi}$ by $\operatorname{com}_{\mathrm{sB}}^{\xi} \leftarrow \mathrm{SBCom}\left(\mathrm{rt}_{1, \mathrm{in}}^{\xi}\left\|\mathrm{dec}_{\mathrm{sH}}^{\xi}\right\| \mathrm{r}^{\xi}\right)$.

- Hybrid $H_{4}$ is identical with $H_{3}$ except that $\mathrm{rt}_{1, \mathrm{i} \mathrm{n}}^{\alpha}, \mathrm{rt}_{1, \mathrm{in}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}$, and $\mathbb{\pi}^{\alpha ; \beta}$ are generated by $\left(\mathrm{rt}_{1, \mathrm{in}}^{\alpha}, \mathrm{rt}_{1, \mathrm{in}}^{\beta}, \mathrm{rt}^{\alpha}, \mathrm{rt}^{\beta}, \mathbb{\pi}^{\alpha ; \beta}\right) \leftarrow$ $\mathcal{S}_{4}\left(\alpha, \beta, f, \mathrm{hf}, \mathbb{Q}^{\alpha: \beta}\right)$.
- Hybrid $H_{5}$ is identical with the ideal execution.

From a hybrid argument, it suffices to show that the output of each hybrid is indistinguishable from that of the preceding one.

Claim I.1. The output of $H_{1}$ is computationally indistinguishable from that of $H_{0}$.
Proof. Assume for contradiction that for infinitely many $\lambda$, the output of $H_{0}$ and that of $H_{1}$ are distinguishable. Fix any such $\lambda$. From an average argument, it follows that the output of $H_{0}$ and that of $H_{1}$ are distinguishable even when the transcript of their executions are fixed up until Round 1 of the prove phase. Since the transcript up until Round 1 of the prove phase determines the value of $(\alpha, \beta)$, and Round 2 of the prove phase of $H_{1}$ can be emulated in polynomial time given $(\alpha, \beta)$, we can now break the sender security of OT, and hence obtain a contradiction.

Claim I.2. The output of $H_{2}$ is computationally indistinguishable from that of $H_{1}$.
Proof. This claim can be proven similarly to Claim I.1; the difference is that we rely on the hiding property of SBCom instead of the sender security of OT.

Claim I.3. The output of $\mathrm{H}_{3}$ is statistically indistinguishable from that of $\mathrm{H}_{2}$.
Proof. This claim follows from the statistical hiding property of SHCom.
Claim I.4. The output of $H_{4}$ is identically distributed with that of $H_{3}$.
Proof. This claim follows from the 2-privacy of $\left\langle C_{4}, R_{4}\right\rangle$.
Claim I.5. The output of $H_{5}$ is identically distributed with that of $H_{4}$.
Proof. This claim can be proven trivially since, by inspection, one can see that the messages that are sent to $R^{*}$ in $H_{4}$ are generated identically with those in the ideal execution.

This concludes the proof of Lemma I.2.

## I. 4 Proof of Soundness

Lemma I.3. Fix any constant $c \in \mathbb{N}$, and let $E_{5}$ be the extractor in Algorithm 20. Then, for any poly $\left(T_{s B}\right)$-time cheating committer-prover $C_{5}^{*}=\left(\mathrm{C}_{\mathrm{Com}}^{5}\right.$, $\left.\mathrm{C} . \operatorname{Prv} \mathrm{v}_{5}^{*}\right)$ against $\left\langle C_{5}, R_{5}\right\rangle$, the following condition holds with overwhelming probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} \mathrm{Com}_{5}^{*}\right.$, R.Com $\left.{ }_{5}\right\rangle$.

- Soundness Condition: If it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left.\begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{5} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C}^{2} \cdot \operatorname{Prv}_{5}^{*}\left(\mathrm{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{5}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \geq 1-\frac{1}{M^{2}}+\frac{1}{\lambda^{c}}, ~ \tag{I.1}
\end{array}\right.
$$

then there exists $x_{\text {com }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that

$$
\begin{equation*}
\forall i \in[n], \operatorname{Pr}\left[x_{i}=x_{i}^{*} \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{5}^{\mathrm{C} \cdot \operatorname{Prv}_{5}^{*}\left(\mathrm{st}_{C}, \cdot\right)}(\operatorname{com}, i), \mathrm{R}^{2} \cdot \operatorname{Dec}_{5}(\operatorname{com}, i)\right\rangle\right] \geq 1-\operatorname{negl}(\lambda) \tag{I.2}
\end{equation*}
$$

and

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \mathrm{Q}_{5} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} . \operatorname{Prv}_{5}^{*}\left(\mathrm{st}_{C}, Q\right) \\
\wedge f\left(x_{\text {Cом }}^{*}\right)=0 & b \leftarrow \mathrm{R} . \operatorname{Prv} . \mathrm{D}_{5}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \leq 1-\frac{1}{M^{2}}+\operatorname{negl}(\lambda) .
$$

```
Algorithm 20 Extractor \(E_{5}\) (against \(\left\langle C_{5}, R_{5}\right\rangle\) )
Input: com, \(i\), and \(\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}\), where com \(=\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\).
```

Step 1. Repeat the following at most $T_{\mathrm{SB}}$ times to obtain $\left\{\mathrm{rl}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ such that each $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\mu}$.

1. Run $\left(\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}, \mathrm{st}_{R}\right) \leftarrow$ R.Prv. $\mathrm{Q}_{5}$.
2. $\operatorname{Run}\left(f,\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: v}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{2}\right) \leftarrow \operatorname{C} \cdot \operatorname{Prv}_{5}^{*}\left(\mathrm{St}_{C},\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}\right)$.
3. Extract the committed values of $\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]}$ by brute force. Let $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\left\|\mathrm{dec}_{\mathrm{sH}}^{\mu}\right\| \mathrm{rt}^{\mu}\right\}_{\mu \in[M]}$ be the extracted values.

Abort if such $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ does not obtained after repeating the above for $T_{\mathrm{sB}}$ times.
Step 2. Run $\left\{\mathbb{Y}^{* \mu: \mu}\right\}_{\mu \in[M]} \leftarrow E_{4}^{\mathcal{A}}\left(\operatorname{com}^{\prime}, i,\left\{\mathbb{Q}^{\mu}\right\}_{\mu \in[M]}\right)$, where com $^{\prime}$ is defined as com ${ }^{\prime}:=\left\{\mathrm{rt}_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}$, and $\mathcal{A}$ is the cheating prover that works as follows on input any $\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, \nu \in[M]}$ :

1. Run $\left(f,\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: \nu}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{2}\right) \leftarrow \mathrm{C} . \operatorname{Prv}_{5}^{*}\left(\mathrm{St}_{C},\left\{\mathbb{Q}^{\mu: \nu}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}\right)$, where ot ${ }_{1}$ is sampled by $\left(\mathrm{ot}_{1}, \mathrm{st}_{\text {от }}\right) \leftarrow \mathrm{OT}_{1}\left(1^{\lambda},(\alpha, \beta)\right)$ for random $\alpha, \beta \in[M]$.
2. Define proof as follows.
(a) Extract the committed values of $\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: v}\right\}_{\mu, v \in[M]}$ by brute force. Let $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\left\|\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\|\right.$ $\left.\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{T}^{\mu \cdot v}\right\}_{\mu, v \in[M]}$ be the extracted values.
(b) Let proof $:=\perp$ if (1) $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\mu}$ for $\forall \mu \in[M]$ but (2) $\left\{\mathrm{rt}_{1, \text { in }}^{\mu}\right\}_{\mu \in[M]}$ is not equal to the one that is obtained in Step 1. Let proof $:=\left(\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\mathrm{Tr}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$ otherwise.
3. Output ( $f$, proof) as the proof.

Step 3. Send $\left(\left\{\mathbb{Y}^{* \mu: \mu}\right\}_{\mu \in[M]},\left\{\text { rt }_{1, \mathrm{i}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}\right)$ to R.Dec ${ }_{5}$.

Proof. Fix any $c$ and $C_{5}^{*}=\left(\mathrm{C}_{5} \mathrm{Com}_{5}^{*}\right.$, C.Prv ${ }_{5}^{*}$ ) as above, and assume for contradiction that for infinitely many $\lambda$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{\mathrm{Com}}{ }_{5}^{*}\right.$, R.Com $\left.{ }_{5}\right\rangle$, the soundness condition does not hold.

To obtain a contradiction, we define a cheating committer-prover $C_{4}^{*}=\left(\mathrm{C} . \mathrm{Com}_{4}^{*}, \mathrm{C} . \operatorname{Prv} 4\right)$ against $\left\langle C_{4}, R_{4}\right\rangle$ by using $C_{5}^{*}$, and show that $C_{4}^{*}$ breaks the soundness of $\left\langle C_{4}, R_{4}\right\rangle$.

Concretely, we consider the following $C_{4}^{*}=\left(\mathrm{C} . \mathrm{Com}_{4}^{*}, \mathrm{C} . \mathrm{Prv}_{4}^{*}\right)$.

- Committer: Given hf as input, C.Com ${ }_{4}^{*}$ does the following.

1. Run $\left(\operatorname{com}^{\prime}, \mathrm{st}_{C}^{\prime}\right) \leftarrow \mathrm{C} . \operatorname{Com}_{5}^{*}(\mathrm{hf})$, and parse $\mathrm{com}^{\prime}$ as $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$.
2. Repeat the following at most $T_{\mathrm{sB}}$ times to obtain $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ such that each $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\mu}$.
(a) $\operatorname{Run}\left(\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}\right.$, ot $\left._{1}, \mathrm{st}_{R}\right) \leftarrow$ R.Prv. $\mathrm{Q}_{5}$.
(b) $\operatorname{Run}\left(f,\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu \cdot v}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{2}\right) \leftarrow \operatorname{C.Prv}_{5}^{*}\left(\mathrm{st}_{C},\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}\right)$.
(c) Extract the committed values of $\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]}$ by brute force. Let $\left\{\mathrm{rt}_{1, \mathrm{i}}^{\mu}\left\|\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\| \mathrm{rt}^{\mu}\right\}_{\mu \in[M]}$ be the extracted values.

Let $b_{\text {BAD }}:=1$ if such $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{SH}}^{\mu}\right\}_{\mu \in[M]}$ does not obtained after repeating the above $T_{\mathrm{SB}}$ times, and let $b_{\text {BAD }}:=0$ otherwise.
3. If $b_{\text {BAD }}=0$, output com $:=\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]}$ as the commitment and $\mathrm{st}_{C}:=\left(\mathrm{com}, \mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}\right)$ as the internal state. If $b_{\text {BAD }}=1$, output com $:=\perp$ as the commitment and $\mathrm{st}_{C}:=\left(\perp, \mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}\right)$ as the internal state.

- Prover: Given $\left(\mathrm{st}_{C},\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$ as input, $\mathrm{C} . \operatorname{Prv}_{4}^{*}$ does the following.

1. Parse $\mathrm{st}_{C}$ as (com, $\mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}$ ), and parse com' as $\left\{\operatorname{com}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$. Abort if com $=\perp$.
2. Run $\left(f,\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: v}\right\}_{\mu \nu v[M]}, \mathrm{ot}_{2}\right) \leftarrow \operatorname{C.Prv}_{5}^{*}\left(\mathrm{St}_{C}^{\prime},\left\{\mathbb{Q}^{\mu: v}\right\}_{\mu, v \in[M]}, \mathrm{ot}_{1}\right)$, where ot ${ }_{1}$ is sampled by $\left(\mathrm{ot}_{1}, \mathrm{st}_{\text {от }}\right) \leftarrow \mathrm{OT}_{1}\left(1^{\lambda},(\alpha, \beta)\right)$ for random $\alpha, \beta \in[M]$.
3. Define proof as follows.
(a) Extract the committed values of $\left\{\operatorname{com}_{\mathrm{sB}}^{\mu}\right\}_{\mu \in[M]},\left\{\operatorname{com}_{\mathrm{sB}}^{\mu: \nu}\right\}_{\mu, \nu \in[M]}$ by brute force. Let $\left\{\mathrm{rr}_{1, \mathrm{in}}^{\mu}\left\|\operatorname{dec}_{\mathrm{sH}}^{\mu}\right\|\right.$ $\left.\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\mathbb{T}^{\mu: \nu}\right\}_{\mu, \nu \in[M]}$ be the extracted values.
(b) Let proof $:=\perp$ if (1) $\operatorname{dec}_{\mathrm{sH}}^{\mu}$ is a valid decommitment for opening $\operatorname{com}_{\mathrm{sH}}^{\mu}$ to $\mathrm{rt}_{1, \mathrm{in}}^{\mu}$ for $\forall \mu \in[M]$ but (2) $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}\right\}_{\mu \in[M]} \neq$ com. Let proof $:=\left(\left\{\mathrm{rt}^{\mu}\right\}_{\mu \in[M]},\left\{\mathrm{T}^{\mu: v}\right\}_{\mu, v \in[M]}\right)$ otherwise.
4. Output ( $f$, proof) as the proof.

Now, we show that $C_{4}^{*}$ breaks the soundness of $\left\langle C_{4}, R_{4}\right\rangle$. Recall that we assume for contradiction that $C_{5}^{*}$ breaks the soundness of $\left\langle C_{5}, R_{5}\right\rangle$. Combined with the the binding property of $\left\langle C_{5}, R_{5}\right\rangle$, this assumption implies that for infinitely many $\lambda \in \mathbb{N}$, with non-negligible probability over the choice of ( $\left.\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{5}^{*}\right.$, R.Com $\left.{ }_{5}\right\rangle$, we have both Equation (I.1) and the following:

- either there exists $i \in[n]$ such that

$$
\begin{equation*}
\operatorname{Pr}\left[x_{i}=\perp \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{5}^{\mathrm{C}_{5} \cdot \operatorname{Prr}_{5}^{*}\left(\mathrm{st}_{C}, \cdot\right)}(\operatorname{com}, i), \operatorname{R.Dec}_{5}(\operatorname{com}, i)\right\rangle\right] \geq \frac{1}{\operatorname{poly}(\lambda)} \tag{I.3}
\end{equation*}
$$

- or there exists $x_{\text {сом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that we have Equation (I.2), but we also have

$$
\operatorname{Pr}\left[b=1 \wedge f\left(x_{\text {com }}^{*}\right)=0 \left\lvert\, \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{5} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C.Prv}_{5}^{*}\left(\mathrm{st}_{C}, Q\right)  \tag{I.4}\\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{5}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right.\right] \geq 1-\frac{1}{M^{2}}+\frac{1}{\operatorname{poly}(\lambda)} .
$$

Then, we consider two cases.
Case 1. First, we consider the case that for infinitely many $\lambda$, with non-negligible probability over the choice of (st ${ }_{C}$, com) $\leftarrow\left\langle\right.$ C.Com $_{5}^{*}$, R.Com $\left.{ }_{5}\right\rangle$, we have Equations (I.1) and (I.3). Let $\Lambda$ be the set of such $\lambda$. In this case, our goal is to obtain a contradiction by showing that for $\forall \lambda \in \Lambda$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} . \mathrm{Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$, we have

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \begin{array}{l}
\left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{Q}_{4} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C} \cdot \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}, Q\right) \\
b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{4}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array} \tag{I.5}
\end{array}\right] \geq \frac{1}{\lambda^{c}}
$$

and

$$
\begin{equation*}
\operatorname{Pr}\left[x_{i}=\perp \mid\left(\perp, x_{i}\right) \leftarrow\left\langle E_{4}^{\mathrm{CPPrv}_{4}^{*}\left(\mathrm{st}_{C},\right)}(\operatorname{com}, i), \text { R. } \operatorname{Dec}_{4}(\operatorname{com}, i)\right\rangle\right] \geq \frac{1}{\operatorname{poly}(\lambda)} \tag{I.6}
\end{equation*}
$$

For any $\lambda \in \Lambda$ and $\left(\mathrm{st}_{C}\right.$, com $) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{5}^{*}\right.$, R.Com $\left.{ }_{5}\right\rangle$, let us say that ( $\mathrm{st}_{C}, \mathrm{com}$ ) is good for $C_{5}^{*}$ if Equations (I.1) and (I.3) hold w.r.t. (st ${ }_{C}$, com). Similarly, for any $\lambda \in \Lambda$ and (st $\left.{ }_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C.Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$, let us say that (st ${ }_{C}$, com) is good for $C_{4}^{*}$ if $\mathrm{st}_{C}$ can be parsed as ( $\mathrm{com}, \mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}$ ) such that ( $\mathrm{st}_{C}^{\prime}, \mathrm{com}^{\prime}$ ) is good for $C_{5}^{*}$. From the assumption of this case, it follows that for $\forall \lambda \in \Lambda$, an execution of $\left\langle\mathrm{C} . \mathrm{Com}_{5}^{*}, \mathrm{R} . \mathrm{Com}_{5}\right\rangle$ produces good (st ${ }_{C}, \mathrm{com}$ ) with non-negligible probability. Hence, from the construction of $C_{4}^{*}$, it follows that for $\forall \lambda \in \Lambda$, an execution of $\left\langle\mathrm{C} . \mathrm{Com}_{4}^{*}, \mathrm{R} . \mathrm{Com}_{4}\right\rangle$ produces $\operatorname{good}\left(\mathrm{st}_{C}, \mathrm{com}\right)$ with non-negligible probability.

We first observe that for $\forall \lambda \in \Lambda$ and an overwhelming fraction of good (st ${ }_{C}, \mathrm{com}$ ) for $C_{4}^{*}$, we have Equation (I.5). Toward this end, we make a sequence of observations.

1. First, for $\forall \lambda \in \Lambda$ and an overwhelming fraction of good (st $\left.{ }_{C}, \mathrm{com}\right)$ for $C_{4}^{*}$, we have that $\mathrm{st}_{C}$ can be parsed as (com, $\mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}$ ) such that com $\neq \perp$.
This is because under that condition that an execution of $\left\langle\mathrm{C} . \mathrm{Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$ produces good ( $\mathrm{st}_{C}$, com), we have $\operatorname{Pr}\left[b_{\mathrm{BAD}}=0\right] \leq \operatorname{negl}(\lambda)$ during the execution of $\left\langle\mathrm{C} . \mathrm{Com}_{4}^{*}\right.$, R.Com $\left.{ }_{4}\right\rangle$ due to the receiver security of OT (Definition 12) and Equation (I.1), ${ }^{34}$ which is guaranteed by the definition of good (st ${ }_{C}, \mathrm{com}$ ).
2. Next, for $\forall \lambda \in \Lambda$ and an overwhelming fraction of good (st ${ }_{C}, \mathrm{com}$ ) for $C_{4}^{*}$, we have both of the following.

- $\mathrm{st}_{C}$ can be parsed as (com, $\mathrm{com}^{\prime}, \mathrm{st}_{C}^{\prime}$ ) such that com $\neq \perp$.
- The probability that $C . \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}\right)$ outputs ( $f$, proof) such that proof $=\perp$ during an execution of $\left\langle\mathrm{C} . \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}\right)\right.$, R. $\left.\operatorname{Prv}_{4}(\mathrm{com})\right\rangle$ is negligible.

This is because of the above observation and the binding property of SHCom.
3. Finally, for $\forall \lambda \in \Lambda$ and an overwhelming fraction of good (st ${ }_{C}$, com) for $C_{4}^{*}$, we have Equation (I.5).

This is because of the above observation and the receiver security of OT. Indeed, if Equation (I.5) does not hold for non-negligible fraction of good $\left(\mathrm{st}_{C}, \mathrm{com}\right)$ for $C_{4}^{*}$, one can break the receiver security of OT (Definition 12) by emulating an execution of $\left\langle\mathrm{C} \cdot \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}\right)\right.$, R. $\left.\operatorname{Prv}_{4}(\mathrm{com})\right\rangle$ since the internally emulated $\mathrm{C} \cdot \mathrm{Prv}_{5}^{*}\left(\mathrm{st}_{C}\right)$ satisfies Equation (I.1).

We next observe that for $\forall \lambda \in \Lambda$ and a non-negligible fraction of $\operatorname{good}\left(\mathrm{st}_{C}, \mathrm{com}\right)$ for $C_{4}^{*}$, we have Equation (I.6). This can be observed by combining the fact that an execution of

[^21]1. $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} \mathrm{Com}_{4}^{*}\right.$, R. $\left.\mathrm{Com}_{4}\right\rangle$;
2. $\left(\perp, x_{i}\right) \leftarrow\left\langle E_{4}^{\mathrm{C} . \text { Prv }_{4}^{*}\left(\mathrm{st}_{c}, \cdot\right)}(\mathrm{com}, i)\right.$, R. $\left.^{\text {Dec }} 44(\mathrm{com}, i)\right\rangle$
perfectly emulates an execution of
3. $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C} . \mathrm{Com}_{5}^{*}\right.$, R.Com $\left.{ }_{5}\right\rangle$;

(where the first step of $E_{5}$ in the latter execution is emulated in the commit phase in the former execution), and the fact that for $\forall \lambda \in \Lambda$ and any $\operatorname{good}\left(\mathrm{st}_{C}, \mathrm{com}\right)$ for $C_{5}^{*}$, we have Equation (I.3).

By combining what is observed in the above two paragraphs, we conclude that for $\forall \lambda \in \Lambda$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} . \mathrm{Com}_{4}^{*}\right.$, R. $\left.\mathrm{Com}_{4}\right\rangle$, we have Equations (I.5) and (I.6) as desired.

Case 2. We next consider the case that for infinitely many $\lambda \in \mathbb{N}$, with non-negligible probability over the choice of (st ${ }_{C}$, com) $\leftarrow\left\langle\right.$ C.Com $_{5}^{*}$, R.Com $\left.{ }_{5}\right\rangle$, we have Equation (I.1) and the following.

- There exists $x_{\text {сом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that we have Equation (I.2), but we also have Equation (I.4).

Let $\Lambda$ be the set of such $\lambda$. In this case, our goal is to obtain a contradiction by showing that for $\forall \lambda \in \Lambda$, with non-negligible probability over the choice of $\left(\mathrm{st}_{C}, \mathrm{com}\right) \leftarrow\left\langle\mathrm{C}^{2} \mathrm{Com}_{4}^{*}, \mathrm{R} . \mathrm{Com}_{4}\right\rangle$, we have Equations (I.5) and the following.

- There exists $x_{\text {сом }}^{*}:=\left(x_{1}^{*}, \ldots, x_{n}^{*}\right) \in\{0,1\}^{n}$ such that we have Equation (H.2), but we also have

$$
\operatorname{Pr}\left[\begin{array}{l|l}
b=1 & \left(Q, \mathrm{st}_{R}\right) \leftarrow \mathrm{R} . \operatorname{Prv} \cdot \mathrm{Q}_{4} ;\left(f, \pi^{*}\right) \leftarrow \mathrm{C}^{\prime} \operatorname{Prv}_{4}^{*}\left(\mathrm{st}_{C}, Q\right)  \tag{I.7}\\
\wedge f\left(x_{\mathrm{COM}}^{*}\right)=0 & b \leftarrow \mathrm{R} \cdot \operatorname{Prv} \cdot \mathrm{D}_{4}\left(\mathrm{st}_{R}, \operatorname{com}, f, \pi^{*}\right)
\end{array}\right] \geq \frac{1}{\operatorname{poly}(\lambda)} .
$$

The proof for this case is similar to the proof for Case 1. Specifically, we can show that Equation (I.1) implies Equation (I.5) as in Case 1, and we can also show that Equation (I.4) implies Equation (I.7) very similarly to Case 1.

By combining the analysis of the above two cases, we complete the proof of Lemma I.3.

## J ZK Scheme with Standard negl-Soundness

As mentioned in Section 1.1, our constant-sound WI commit-and-prove protocol $\left\langle C_{5}, R_{5}\right\rangle$ in Appendix I can be transformed into a one that is zero-knowledge and negl-sound by using a variant of a transformation of Khurana et al. [KOS18]. For completeness, we briefly explain how $\left\langle C_{5}, R_{5}\right\rangle$ is transformed by it. (An overview of the transformation is given in [KOS18].)

## J. 1 Protocol Description

The transformation uses the following building blocks.

- A three-round commitment scheme two-com such that the committer commits to two values $\hat{s}_{0}, \hat{s}_{1}$ in the first round and then reveals them in the third round, and it is guaranteed that (1) one of the two commitments is binding while the other is equivocal, and (2) the receiver cannot tell which commitment is equivocal. Black-box constructions of such a commitment scheme are given in [ORS15, KOS18].
- A finite field $\boldsymbol{G}$ such that $|\boldsymbol{G}|$ is exponentially large.

We describe the transformed protocol $\left\langle C_{6}, R_{6}\right\rangle$ in Algorithm 21. (The only difference between the transformation that we use and the one given in [KOS18] is the definition of $a$ in Round 2 of the prove phase, where in [KOS18], it is defined as $a:=x \alpha+r$ by viewing the committed value $x$ as an element in $\boldsymbol{G} .{ }^{35}$ )

## J. 2 Proof Sketch of Binding

The binding property follows from that of $\left\langle C_{5}, R_{5}\right\rangle$.

[^22]```
Algorithm 21 Commit Phase, Open Phase, and Prove Phase of \(\left\langle C_{6}, R_{6}\right\rangle\)
Commit Phase
Round 1: R.Com runs \(_{6} \mathrm{hf}_{j} \leftarrow \mathrm{R}^{2} \mathrm{Com}_{5}\) for \(\forall j \in[\lambda]\) and then sends \(\left\{\mathrm{hf}_{j}\right\}_{j \in[\lambda]}\) and the first-round message of two-com \(\left(\hat{s}_{0}, \hat{s}_{1}\right)\)
for randomly chosen \(\hat{s}_{0}, \hat{s}_{1} \in\{0,1\}^{\lambda}\).
Round 2: C.Com \({ }_{6}\) does the following.
```

1. Choose random $r \in \boldsymbol{G}$ and $s \in\{0,1\}^{\lambda}$, and let $x_{\text {сом }}^{\prime}:=x_{\text {сом }}\|r\| s$.
2. Run $\left(\operatorname{com}_{j}\right.$, st $\left._{C}^{j}\right) \leftarrow \operatorname{C.Com}_{5}\left(x_{\text {сом }}^{\prime}\right.$, hf) $)$ for $\forall j \in[\lambda]$.
3. Send $\left\{\operatorname{com}_{j}\right\}_{j \in[\lambda]}$ and the second-round message of two-com as the commitment.

## Prove Phase

Round 1: R.Prv. $\mathrm{Q}_{6}$ runs $\left(Q_{j}, \mathrm{st}_{R}^{j}\right) \leftarrow$ R.Prv. $\mathrm{Q}_{5}$ for $j \in[\lambda]$, and then sends $\left\{Q_{j}\right\}_{j \in[\lambda]}$, randomly chosen $\alpha \in \boldsymbol{G}$, and the third-round message of two-com (which reveals $\hat{s}_{0}$ and $\hat{s}_{1}$ ) as the query.
Round 2: C. Prv $_{6}$ does the following.

1. Compute $a=x_{n} \alpha^{n}+\cdots+x_{1} \alpha+r$ (in the field $\boldsymbol{G}$ ), where $x_{i}$ is the $i$-th bit of $x_{\text {coм }}$.
2. Run $\pi_{j} \leftarrow C . \operatorname{Prv}_{5}\left(\mathrm{st}_{C}^{j}, \hat{f}, Q_{j}\right)$ for $\forall j \in[\lambda]$, where the function $\hat{f}$ checks, given input $x_{\text {сом }}^{\prime}=x_{\text {сом }}\|r\| s$, whether it holds

$$
\left(f(x)=1 \wedge x_{n} \alpha^{n}+\cdots+x_{1} \alpha+r=a\right) \vee\left(s=\hat{s}_{0} \vee s=\hat{s}_{1}\right) .
$$

3. Send $a$ and $\left\{\pi_{j}\right\}_{j \in[\lambda]}$ as the proof.

Verification: R.Prv.D ${ }_{6}$ outputs 1 if and only if R.Prv. $\mathrm{D}_{5}\left(\mathrm{st}_{R}^{j}, \operatorname{com}_{j}, \hat{f}, \pi_{j}\right)=1$ for $\forall j \in[\lambda]$.

## Open Phase

Round 1: R.Dec. $\mathrm{Q}_{6}$ runs $\left(Q_{j}, \mathrm{st}_{R}^{j}\right) \leftarrow$ R.Dec. $\mathrm{Q}_{5}(i)$ for $\forall j \in[\lambda]$ and sends $\left\{Q_{j}\right\}_{j \in[\lambda]}$ and the third-round message of two-com as the query.
Round 2: $\mathrm{C}^{2} \mathrm{Dec}_{6}$ runs $\operatorname{dec}_{j} \leftarrow \mathrm{C} . \operatorname{Dec}_{5}\left(\mathrm{st}_{C}^{j}, Q_{j}\right)$ for $\forall j \in[\lambda]$ and sends $\left\{\mathrm{dec}_{j}\right\}_{j \in[\lambda]}$ as the decommitment.
Verification: R.Dec. D $_{6}$ does the following.

1. Check whether there exists $x_{i}$ such that it holds R.Dec. $D_{5}\left(\operatorname{st}_{R}^{j}, \operatorname{com}_{j}, \operatorname{dec}_{j}\right)=x_{i}$ for more than $\lambda / 2$ values of $j \in[\lambda]$.
2. Output $x_{i}$ as the decommitted value if such $x_{i}$ exists, and output $\perp$ otherwise.

## J. 3 Proof Sketch of Zero-knowledge

The proof of zero-knowledge proceeds identically with that given in [KOS18]. Specifically, it suffices to consider a simulator that first obtains the values $\hat{s}_{0}, \hat{s}_{1}$ in Round 1 of the prove phase and then uses them to simulate the receiver's view in the commit and prove phases by relying on rewinding techniques.

## J. 4 Proof Sketch of Soundness

The proof of soundness proceeds essentially identically with that given in [KOS18]. Specifically, we prove it by considering an extractor that applies the extractor $E_{5}$ of $\left\langle C_{5}, R_{5}\right\rangle$ for each instance of $\left\langle C_{5}, R_{5}\right\rangle$ in the prove phase. Given this extractor, it suffices to show the following.

1. The extraction succeeds in most instances of $\left\langle C_{5}, R_{5}\right\rangle$.
2. Any cheating committer cannot prove false statements on the extracted value in most instances of $\left\langle C_{5}, R_{5}\right\rangle$.
3. The values extracted from most instances of $\left\langle C_{5}, R_{5}\right\rangle$ are equal.

The first two can be shown by following the analysis of $E_{5}$ (Lemma I.3). Specifically, the receiver security of OT guarantees that in at least $\lambda-\omega(\log \lambda)$ instances of $\left\langle C_{5}, R_{5}\right\rangle$, all the values that are committed by SBCom (e.g., decommitments to the hash of the initial MPC states) are correctly generated with non-negligible probability, and thus we can reuse the analysis of $E_{5}$. The last one can be shown by noticing that if different values, $x_{\text {Сом }}\|r\| s$ and $x_{\text {СОм }}^{\prime}\left\|r^{\prime}\right\| s^{\prime}$ such that $x_{\text {Сом }} \neq x_{\text {Сом }}^{\prime}$, are extracted, then with overwhelming probability over the choice of $\alpha \in \boldsymbol{G}$, it holds $x_{n} \alpha^{n}+\cdots+x_{1} \alpha+r \neq x_{n}^{\prime} \alpha^{n}+\cdots+x_{1}^{\prime} \alpha+r^{\prime}$ (which implies that we have either $x_{n} \alpha^{n}+\cdots+x_{1} \alpha+r \neq a$ or $x_{n}^{\prime} \alpha^{n}+\cdots+x_{1}^{\prime} \alpha+r^{\prime} \neq a$ ) and hence the cheating committer is required to prove a false statement in an instance of $\left\langle C_{5}, R_{5}\right\rangle$ unless it can break the security of two-com.

## K Lemmas from Kalai et al. [KRR14] and Subsequent Works

We give several lemmas that are based on those given in Kalai et al. [KRR14] and subsequent works. All the lemmas in this section can be trivially extended for the parallel setting that we consider in Appendix E (i.e., the setting where $\mathcal{A}$ takes multiple sets of queries and outputs multiple functions).

## K. 1 Lemmas on SelfCorr

For each $\lambda \in \mathbb{N}$, let $\boldsymbol{F}, \boldsymbol{H}, m, m_{10}$ be the parameters of our PCP system (PCP.P, PCP.V), and let $D(X), D\left(X^{\xi}\right), D\left(X_{1, \text { in }}^{\xi}\right)$ be defined as in Section D.2.2. Let $\zeta=\omega(\log \lambda)$. Let LD-Test and SelfCorr be the algorithms that are defined in Algorithm 6 and Algorithm 1.

Then, there exists a polynomial ${ }^{36} \kappa_{1}=O\left(\lambda|\boldsymbol{F}|^{2}\right.$ ) such that for any $\kappa_{1}$-CNS adversary $\mathcal{A}$, the following lemmas hold. In the following, unless otherwise specified, we use $(d, D)$ to denote any of $(m|\boldsymbol{H}|, D(X)),\left((m-1)|\boldsymbol{H}|, D\left(X^{\xi}\right)\right)$, and ( $m_{\mathrm{Io}}|\boldsymbol{H}|, D\left(X_{1, \mathrm{in}}^{\xi}\right)$ ).

Lemma K. 1 (Correctness of SelfCorr). If it holds

$$
\operatorname{Pr}\left[b=1 \mid b \leftarrow \text { LD-Test }_{d, D, \zeta}^{\mathcal{P}}\right] \geq 1-\operatorname{negl}(\lambda)
$$

for infinitely many $\lambda \in \mathbb{N}$ (let $\Lambda$ be the set of such $\lambda$ ), then there exists a negligible function $\epsilon$ such that for every sufficiently large $\lambda \in \Lambda$ and $\forall q \in D$, it holds

$$
\operatorname{Pr}\left[\tilde{A}(q)=\perp \mid(\text { out }, \tilde{A}) \leftarrow \text { SelfCorr }_{d, D}^{\mathcal{A}}(\{q\})\right] \leq \epsilon(\lambda) .
$$

Proof. See [KRR13, Theorem 7.27] or [BHK16, Lemma 8].
Lemma K. 2 (Consistency of SelfCorr). If it holds

$$
\operatorname{Pr}\left[b=1 \mid b \leftarrow{\left.\operatorname{LD}-\operatorname{Test}_{d, D, \zeta}^{\mathcal{Y}}\right]}_{\mathcal{A}}[1-\operatorname{negl}(\lambda),\right.
$$

for infinitely many $\lambda \in \mathbb{N}$ (let $\Lambda$ be the set of such $\lambda$ ), then there exists a negligible function $\epsilon$ such that for every sufficiently large $\lambda \in \Lambda$ and $\forall q \in D$, it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\tilde{A}_{0}(q) \neq \tilde{A}_{1}(q) & \begin{array}{l}
\left(Q_{b}, \mathrm{st}_{b}\right) \leftarrow \text { SelfCorr.Q } Q_{D}(\{q\}) \text { for } \forall b \in\{0,1\} \\
\left(\mathrm{out}^{\prime}, A\right) \leftarrow \mathcal{A}\left(Q_{0} \cup Q_{1}\right) \\
\tilde{A}_{b}:=\operatorname{SelfCorrr}^{2} \operatorname{Rec}_{d}\left(\mathrm{st}_{b}, A\right) \text { for } \forall b \in\{0,1\}
\end{array}
\end{array}\right] \leq \epsilon(\lambda)
$$

Proof sketch. This can be proven by slightly modifying the proofs of [KRR13, Theorem 7.27] and [BHK16, Lemma 8]. Specifically, by inspection, one can see that these proofs consider, very roughly speaking, a modified version of SelfCorr that chooses $2 \lambda$ lines $L_{0,1}, \ldots, L_{0, \lambda}, L_{1,1}, \ldots, L_{1, \lambda}$ for each query (instead of choosing $\lambda$ lines), and then show that

1. the same value $c_{i}$ is recovered by self-correction from $L_{0, i}$ and $L_{1, i}$ for most $i \in[\lambda]$, and
2. there exists a single value $c$ such that it holds $c_{i}=c$ for most $i \in[\lambda]$.

Hence, it follows that the same value is recovered by self-correction from most of the $2 \lambda$ lines, and thus, when those $2 \lambda$ lines are considered to be chosen from two instances of SelfCorr, the values that are recovered from them are the same. (For a formal argument, see [Kiy18, Claim 3].)

Lemma K.3. For each $\lambda \in \mathbb{N}$, let $\left(m_{0}, d_{0}, D_{0}\right):=(m, m|\boldsymbol{H}|, D(X))$ and let $\left(m_{1}, d_{1}, D_{1}\right)$ be either $\left(m-1,(m-1)|\boldsymbol{H}|, D\left(X^{\xi}\right)\right)$ or $\left(m_{I O}, m_{I O}|\boldsymbol{H}|, D\left(X_{1, \text { in }}^{\xi}\right)\right)$.

Assume that the following hold for infinitely many $\lambda \in \mathbb{N}$ and let $\Lambda$ be the set of such $\lambda$.

- Low-degree Test on $D_{b}(b \in\{0,1\})$ : For $b \in\{0,1\}$, it holds

$$
\operatorname{Pr}\left[b=1 \mid b \leftarrow \operatorname{LD}^{\operatorname{T-Test}}{ }_{d_{b}, D_{b}, \zeta}^{\mathcal{M}}\right] \geq 1-\operatorname{negl}(\lambda) .
$$

- Low-degree Test, conditioned on $L(0) \in D_{1}$ : It holds $b=1$ with probability at least $1-$ negl $(\lambda)$ in the following probabilistic experiment.

1. Choose $\lambda$ random lines $L_{1}, \ldots, L_{\lambda}: \boldsymbol{F} \rightarrow D_{0}$ such that each $L \in\left\{L_{1}, \ldots, L_{\lambda}\right\}$ satisfies $L(0) \in D_{1}$.
2. Run (out, $A) \leftarrow \mathcal{A}(Q)$, where $Q=\left\{L_{j}(t)\right\}_{j \in[\lambda], t \epsilon \boldsymbol{F}}$.

[^23]3. Let $b=1$ if and only if
$$
\left|\left\{j \in[\lambda] \mid \operatorname{isLD}_{d_{0}}\left(\left\{A\left(L_{j}(t)\right)\right\}_{t \in \boldsymbol{F}}\right)=1\right\}\right| \geq \lambda-\zeta .
$$

- $D_{1}$-parallel Low-degree Test: It holds $b=1$ with probability at least $1-\operatorname{neg|}(\lambda)$ in the following probabilistic experiment.

1. Choose $\lambda$ random lines $L_{1}, \ldots, L_{\lambda}: \boldsymbol{F} \rightarrow D_{0}$ as follows: for each $j \in[\lambda]$, choose random points $\boldsymbol{r} \in D_{0}=\boldsymbol{F}^{m_{0}}$ and $\boldsymbol{r}^{\prime} \in\left\{\left(0, \cdots, 0, v_{m_{0}-m_{1}+1}, \ldots, v_{m_{0}}\right) \mid \forall\left(v_{m_{0}-m_{1}+1}, \ldots, v_{m_{0}}\right) \in \boldsymbol{F}^{m_{1}}\right\} \in D_{0}=\boldsymbol{F}^{m_{0}}$, and define $L_{j}: \boldsymbol{F} \rightarrow D$ as $L_{j}(\alpha)=\boldsymbol{r}+\alpha \cdot \boldsymbol{r}^{\prime}$.
2. Run $($ out, $A) \leftarrow \mathcal{A}(Q)$, where $Q=\left\{L_{j}(t)\right\}_{j \in[\lambda], t \in \boldsymbol{F}}$.
3. Let $b=1$ if and only if

$$
\left|\left\{j \in[\lambda] \mid \operatorname{isLD}_{d_{1}}\left(\left\{A\left(L_{j}(t)\right)\right\}_{t \in \boldsymbol{F}}\right)=1\right\}\right| \geq \lambda-\zeta .
$$

Then, there exists a negligible function $\epsilon$ such that for every sufficiently large $\lambda \in \Lambda$, and for $\forall q \in D_{1}$, it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\tilde{A}_{0}(q) \neq \tilde{A}_{1}(q) & \begin{array}{l}
\left(Q_{b}, \mathrm{st}_{b}\right) \leftarrow \text { SelfCorr.Q } Q_{D_{b}}(\{q\}) \text { for } \forall b \in\{0,1\} \\
(\text { out, } A) \leftarrow \mathcal{A}\left(Q_{0} \cup Q_{1}\right) \\
\tilde{A}_{b}:=\operatorname{SelfCorr.Rec}_{d}\left(\mathrm{st}_{b}, A\right) \text { for } \forall b \in\{0,1\}
\end{array}
\end{array}\right] \leq \epsilon(\lambda)
$$

Proof sketch. This lemma can be proven in a similar spirit to [HR18, Proposition 10.12]. We give a proof sketch for completeness.

From the CNS of $\mathcal{A}$, it suffices to show $\operatorname{Pr}\left[\tilde{A}_{0}(q) \neq \tilde{A}_{1}(q)\right] \leq \epsilon(\lambda)$ in the following probabilistic experiment.

1. Choose $\lambda$ random planes $M_{1}, \ldots, M_{\lambda}: \boldsymbol{F}^{2} \rightarrow D_{0}$ such that each $M \in\left\{M_{1}, \ldots, M_{\lambda}\right\}$ satisfies the following: (1) $M(0,0)=q ;(2)$ the line $M(\cdot, 0)$ is fully contained in $D_{1}$.
2. Run (out, $A) \leftarrow \mathcal{A}(Q)$, where $Q=\left\{M_{j}\left(t, t^{\prime}\right)\right\}_{j \in\left[\{ ], t, t^{\prime} \in \boldsymbol{F}\right.}$.

(To see that it indeed suffices to bound the probability in the above experiment, observe that the lines $M_{1}(0, \cdot), \ldots, M_{\lambda}(0, \cdot)$ are random lines on $D_{0}$ such that each $L \in\left\{M_{1}(0, \cdot), \ldots, M_{\lambda}(0, \cdot)\right\}$ satisfies $L(0)=q$, and the lines $M_{1}(\cdot, 0), \ldots, M_{\lambda}(\cdot, 0)$ are random lines on $D_{1}$ such that each $L \in\left\{M_{1}(\cdot, 0), \ldots, M_{\lambda}(\cdot, 0)\right\}$ satisfies $L(0)=q$.)

Hence, we focus on showing $\operatorname{Pr}\left[\tilde{A}_{0}(q) \neq \tilde{A}_{1}(q)\right] \leq \epsilon(\lambda)$ in the above experiment. First, from "Low-degree Test, conditioned on $L(0) \in D_{1}$," it follows that for each $\alpha \in \boldsymbol{F} \backslash\{0\}$, we have $\operatorname{Pr}\left[\left|J_{0, \alpha}\right| \geq \lambda-\zeta\right] \geq 1-\operatorname{neg} \mid(\lambda)$, where

$$
J_{0, \alpha}:=\left\{j \in[\lambda] \mid \operatorname{isLD}_{d_{0}}\left(\left\{A\left(M_{j}(\alpha, t)\right)\right\}_{t \in \boldsymbol{F}}\right)=1\right\} .
$$

Second, from " $D_{1}$-parallel Low-degree Test," it follows that for each $\beta \in \boldsymbol{F} \backslash\{0\}$, we have $\operatorname{Pr}\left[\left|J_{1, \beta}\right| \geq \lambda-\zeta\right] \geq 1-\operatorname{neg} \mid(\lambda)$, where

$$
J_{1, \beta}:=\left\{j \in[\lambda] \mid \operatorname{isLD}_{d_{1}}\left(\left\{A\left(M_{j}(t, \beta)\right)\right\}_{t \in \boldsymbol{F}}\right)=1\right\} .
$$

Thus, using a union bound, we obtain $\operatorname{Pr}\left[\left|J_{\text {good }}\right| \geq \lambda-2|\boldsymbol{F}| \zeta\right] \geq 1-\operatorname{neg} \mid(\lambda)$, where

$$
J_{\text {good }}:=\bigcap_{b \in\{0,1\}, t \in \boldsymbol{F} \backslash\{0\}} J_{b, t}
$$

Also, it is easy to observe that for every $j \in J_{\text {good }}$, there exists $c_{j}$ such that

$$
\operatorname{Recon}_{d_{0}}\left(\left\{A\left(M_{j}(0, t)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)=\operatorname{Recon}_{d_{1}}\left(\left\{A\left(M_{j}(t, 0)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)=c_{j}
$$

(See, e.g., [KRR13, Proposition 7.23].) Now, note that Lemma K. 1 implies that with overwhelming probability, there exist $c_{0}^{\prime}, c_{1}^{\prime}$ such that

$$
\left|\left\{j \in[\lambda] \mid \operatorname{Recon}_{d_{0}}\left(\left\{A\left(M_{j}(0, t)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)=c_{0}^{\prime}\right\}\right| \geq 0.9 \lambda
$$

and

$$
\left|\left\{j \in[\lambda] \mid \operatorname{Recon}_{d_{1}}\left(\left\{A\left(M_{j}(t, 0)\right)\right\}_{t \in \boldsymbol{F} \backslash\{0\}}\right)=c_{1}^{\prime}\right\}\right| \geq 0.9 \lambda .
$$

Using a union bound and the fact that $0.8 \lambda-2|\boldsymbol{F}| \zeta>0$ for sufficiently large $\lambda$, we obtain that with overwhelming probability, there exists $j$ such that $c_{0}^{\prime}=c_{j}=c_{1}^{\prime}$. From the construction of SelfCorr, this implies that $\operatorname{Pr}\left[\tilde{A}_{0}(q) \neq \tilde{A}_{1}(q)\right] \leq \epsilon(\lambda)$, as desired.

## K. 2 Soundness Amplification Lemma

We give a slightly extended version of the soundness amplification lemma of Brakerski et al. [BHK17].
Lemma K. 4 (Soundness Amplification Lemma). For any polynomials $\kappa, \kappa_{\max }$, any $\kappa$-query verifier $\mathrm{V}=(\mathrm{Q}, \mathrm{D})$, any $c>0$, and the PPT oracle algorithm Amplify ${ }_{c}$ in Algorithm 22, the following holds. For any $\zeta(\lambda)=\omega(\log \lambda)$ and $k \in\{0,1, \ldots$,$\} ,$ if there exists an adaptive $\kappa_{\max }-$ CNS adversary $\mathcal{A}$ such that it holds

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\mathrm{D}^{\geq \lambda-k \zeta}(\mathrm{st}, \text { out, } A)=1 & \begin{array}{l}
(Q, \mathrm{st}) \leftarrow \mathrm{Q}^{\otimes \lambda}\left(1^{\lambda}\right) \\
(\text { out }, A) \leftarrow \mathcal{A}(Q)
\end{array} \tag{K.1}
\end{array}\right] \geq \frac{1}{\lambda^{c}}
$$

for infinitely many $\lambda$ (let $\Lambda$ be the set of such $\lambda$ ), then $\operatorname{Amplify}{ }^{\mathcal{A}}\left(1^{\lambda}, \cdot\right)$ is an adaptive $\left(\kappa_{\max }-\lambda \kappa\right)$-CNS cheating prover such that there is a negligible function negl such that for every $\lambda \in \Lambda$,

$$
\operatorname{Pr}\left[\begin{array}{l|l}
\mathrm{D}^{\geq \lambda-(2 k+1) \zeta}(\text { st, out, } A)=1 & \begin{array}{l}
(Q, \text { st }) \leftarrow \mathrm{Q}^{\otimes \lambda}\left(1^{\lambda}\right) \\
(\text { out }, A) \leftarrow \operatorname{Amplify} \\
c
\end{array}\left(1^{\lambda}, Q\right) \tag{K.2}
\end{array}\right] \geq 1-\operatorname{negl}(\lambda) .
$$

Furthermore, for any sequence of queries $\left\{S_{\lambda}\right\}_{\lambda \in \mathbb{N}}$ such that $\left|S_{\lambda}\right| \leq \kappa_{\max }-2 \lambda \kappa$, the following two distributions on (out, $\left.A\right|_{S_{\lambda}}$ ) are computationally indistinguishable.

1. (out, $\left.A\right|_{S_{\lambda}}$ ) is sampled through the conditional distribution

$$
\begin{array}{l|l}
(Q, \text { st }) \leftarrow \mathrm{Q}^{\otimes \lambda}\left(1^{\lambda}\right) & \mathrm{D}^{\geq \lambda-k \zeta}(\text { st, out, } A)=1  \tag{K.3}\\
(\text { out }, A) \leftarrow \mathcal{A}\left(Q \cup S_{\lambda}\right) &
\end{array}
$$

2. (out, $\left.A\right|_{S_{\lambda}}$ ) is sampled through the distribution

$$
\begin{align*}
& (Q, \text { st }) \leftarrow \mathrm{Q}^{\otimes \lambda}\left(1^{\lambda}\right) \\
& (\text { out }, A) \leftarrow \text { Amplify }_{c}^{\mathcal{A}}\left(1^{\lambda}, Q \cup S_{\lambda}\right) . \tag{K.4}
\end{align*}
$$

```
Algorithm 22 Amplify \(_{c}^{\mathcal{A}}\left(1^{\lambda}, Q\right)\)
    1. Run \(\left(Q_{i}, \mathrm{st}_{i}\right) \leftarrow \mathrm{Q}^{\otimes \lambda}\) and \(\left(\right.\) out \(\left._{i}, A_{i}\right) \leftarrow \mathcal{A}\left(Q \cup Q_{i}\right)\) for each \(i \in\left[\lambda^{c+1}\right]\).
    2. Find the first \(i^{*} \in\left[\lambda^{c+1}\right]\) such that \(\mathrm{D}^{\geq \lambda-k \zeta}\left(\mathrm{st}_{i^{*}}\right.\), out, \(\left.\left.A_{i^{*}}\right|_{Q^{*}}\right)=1\), and output (out \(i^{*}, A_{i^{*}} \mid Q\) ) if such \(i^{*}\) exists, and output
        \(\perp\) otherwise.
```

The above version is extended from the one by Brakerski et al. [BHK17] in that (1) we consider soundness with $D^{\geq \lambda-k \zeta}$ and $D^{\geq \lambda-(2 k+1) \zeta}$ rather than soundness with $D^{\otimes \lambda}$ and $D^{\geq \lambda-\zeta}$, and (2) in the "furthermore" part, we consider the distribution on (out, $\left.A\right|_{S_{\lambda}}$ ) rather than on out.

Proof sketch. Since the proof is a straightforward extension of the one by Brakerski et al. [BHK17], we only give a proof sketch.

Toward proving this lemma, we consider three claims.
Claim K.1. Amplify ${ }^{\mathcal{A}}\left(1^{\lambda}, \cdot\right)$ is an adaptive $\left(\kappa_{\max }-\lambda \kappa\right)$-CNS cheating prover.
Proof. This can be proven in exactly the same way as in Brakerski et al. [BHK17].
Claim K.2. Equation (K.2) holds for every $\lambda \in \Lambda$.
Proof. We first note that in exactly the same way as Brakerski et al. [BHK17], we can show that the probability that Assign ${ }_{c}$ outputs $\perp$ is negligible.

Thus, we focus showing Equation (K.2) under the condition that Assign ${ }_{c}$ does not output $\perp$, i.e., under the condition that a "good" $i^{*}$ exists in the execution of Assign ${ }_{c}$. In this case, it suffices to show that the conditional probability

$$
\begin{equation*}
\operatorname{Pr}\left[\mathrm{D}^{\lambda-(2 k+1) \zeta}\left(\text { st }^{2}, \text { out }_{i^{*}},\left.A_{i^{*}}\right|_{Q}\right)=0 \mid \mathrm{D}^{\lambda-k \zeta}\left(\mathrm{st}_{i^{*}}, \text { out }_{i^{*}},\left.A_{i^{*}}\right|_{Q_{i^{*}}}\right)=1\right] \tag{K.5}
\end{equation*}
$$

is negligible, where the probability is taken over $(Q$, st $) \leftarrow Q^{\otimes \lambda},\left(Q_{i^{*}}, \mathrm{st}_{i^{*}}\right) \leftarrow \mathrm{Q}^{\otimes \lambda}$, and (out $\left.\mathrm{i}_{i^{*}}, A_{i^{*}}\right) \leftarrow \mathcal{A}\left(Q \cup Q_{i^{*}}\right)$. First, from the CNS of $\mathcal{A}$ and Equation (K.1), we have

$$
\begin{equation*}
\operatorname{Pr}\left[\mathrm{D}^{\lambda-k \zeta}\left(\text { st }_{i^{*}}, \text { out }_{i^{*}}, A_{i^{*}} \mid Q_{i^{*}}\right)=1\right] \geq \frac{1}{\lambda^{c}}-\operatorname{negl}(\lambda) . \tag{K.6}
\end{equation*}
$$

Next, we have

$$
\begin{equation*}
\operatorname{Pr}\left[\mathrm{D}^{\lambda-(2 k+1) \zeta}\left(\text { st, out }_{i^{*}},\left.A_{i^{*}}\right|_{Q}\right)=0 \wedge \mathrm{D}^{\lambda-k \zeta}\left(\text { st }_{i^{*}}, \text { out }_{i^{*}},\left.A_{i^{*}}\right|_{Q^{*}}\right)=1\right] \leq \operatorname{negl}(\lambda) . \tag{K.7}
\end{equation*}
$$

This can be obtained from the following observations.

- We consider a mental experiment where, instead of sampling $Q$ and $Q_{i^{*}}$ individually, we first sample $\hat{Q}$ of size $|Q|+\left|Q_{i^{*}}\right|$, next separate it into $\lambda$ pairs $\left\{\left(\hat{Q}_{j, 0}, \hat{Q}_{j, 1}\right)\right\}_{j \in[\lambda]}$, and then for each $j \in[\lambda]$, pick one of $\hat{Q}_{j, 0}$ and $\hat{Q}_{j, 1}$ for $Q$ and the other for $Q_{i^{*}}$. Fix $\hat{Q}$ and $\left\{\left(\hat{Q}_{j, 0}, \hat{Q}_{j, 1}\right)\right\}_{j \in[\lambda]}$ (but not $Q$ or $Q_{i^{*}}$ individually), and further fix $\mathcal{F}$ 's response on $\hat{Q}$. Let $N$ be the number of $j$ 's such that $\mathcal{A}$ gives a rejecting answer to at least one of $\hat{Q}_{j, 0}$ and $\hat{Q}_{j, 1}$.
- Case 1. $N \geq(2 k+1) \zeta$ : In this case, we have $\mathrm{D}^{\lambda-k \zeta}\left(\mathrm{St}_{i^{*}}\right.$, out $\left.t_{i^{*}},\left.A_{i^{*}}\right|_{Q^{*}}\right)=1$ only when at most $k \zeta$ rejecting queries are picked for $Q_{i^{*}}$ in the $N$ bad pairs, but Chernoff Bound implies that the probability that this occurs is at most

$$
\exp \left(-\frac{1}{2} \delta^{2} E\right)=\exp \left(-\frac{1}{8\left(k+\frac{1}{2}\right)} \zeta\right)=\operatorname{negl}(\lambda),
$$

where $\delta=\frac{1}{2} /\left(k+\frac{1}{2}\right)$ and $E=\left(k+\frac{1}{2}\right) \zeta$.

- Case 2. $N<(2 k+1) \zeta$ : In this case, we never have $\mathrm{D}^{\lambda-(2 k+1) \zeta}\left(\right.$ st, out $\left.i_{i^{*}}, A_{i^{*}} \mid Q\right)=0$.

Combining Equation (K.6) and (K.7), we have that the probability in Equation (K.5) is negligible, as desired.
Claim K.3. The "furthermore" part of the lemma holds.
Proof. For any $\left\{S_{\lambda}\right\}_{\lambda \in \mathbb{N}}$, consider a modified version of Amplify ${ }_{c}$, denoted by Amplify ${ }_{c}^{\prime}$, that samples each (out ${ }_{i}, A_{i}$ ) from $\mathcal{A}\left(S_{\lambda} \cup Q_{i}\right)$ instead of from $\mathcal{A}\left(Q \cup S_{\lambda} \cup Q_{i}\right)$. Now, the distribution on (out, $\left.\left.A\right|_{S_{\lambda}}\right)$ that is sampled through the distribution (K.3) is identically distributed with the one that is sampled through the conditional distribution

$$
\begin{array}{l|l}
(Q, \mathrm{st}) \leftarrow \mathrm{Q}^{\otimes \lambda}\left(1^{\lambda}\right) & \text { Amplify }_{c}^{\prime \mathcal{A}} \text { does not output } \perp . \\
(\text { out }, A) \leftarrow \text { Amplify }{ }_{c}^{\prime \mathcal{A}}\left(1^{\lambda}, Q \cup S_{\lambda}\right) & .
\end{array}
$$

Furthermore, since Amplify ${ }_{c}^{\mathcal{A}}$ outputs $\perp$ only with negligible probability, this distribution is statistically close to the one that is sampled through

$$
\begin{aligned}
& (Q, \text { st }) \leftarrow \mathrm{Q}^{\otimes \lambda}\left(1^{\lambda}\right) \\
& (\text { out }, A) \leftarrow \text { Amplify }^{\prime \mathcal{F}}\left(1_{c}^{\lambda}, Q \cup S_{\lambda}\right)
\end{aligned} .
$$

Finally, due to the CNS of $\mathcal{A}$, this distribution is computationally indistinguishable from the one that is sampled through the distribution (K.4)

This completes the proof sketch of Lemma K.4.


[^0]:    This article is a full version of the following article: Round-optimal Black-box Commit-and-prove with Succinct Communication, in Proceedings of CRYPTO 2020, ©IACR 2020. Parts of this work were done while the author was a member of NTT Secure Platform Laboratories.

[^1]:    ${ }^{1}$ For example, the definition in [IW14] considers non-deterministic statements on committed values but the statements are assumed to be fixed in the commit phase, whereas our definition considers deterministic statements but the statements are allowed to be chosen after the commit phase is completed.
    ${ }^{2}$ Roughly speaking, this is because in a setting where statements to be proven are chosen after the commit phase (e.g., the delayed-input setting), techniques in [GOSV14, IW14] require that (1) the commit phase has 2 rounds as it needs to be succinct and (2) the prove phase has 3 rounds as it has a commit-challenge-response structure.

[^2]:    ${ }^{3}$ Such an MPC protocol can be obtained unconditionally (e.g., the 2-private five-party protocol by Ben-Or et al. [BGW88, AL17]).
    ${ }^{4} \mathrm{We}$ assume that the set $[M] \times[M]$ is identified with the set $\left[M^{2}\right]$ in a canonical way.
    ${ }^{5}$ Formally, complexity leveraging is required in this argument since the receiver security of OT needs to hold even against adversaries that extract the committed initial states and views by brute force.

[^3]:    ${ }^{6} \mathrm{We}$ assume that the length of the message to be committed, $n$, is implicitly given to the receiver as input.
    ${ }^{7}$ We assume that $\operatorname{Time}(f)$ is known to the both parties in advance (where $f$ is expressed as, e.g., a Turing machine).

[^4]:    ${ }^{8}$ Roughly speaking, the soundness condition requires that if a cheating prover convinces the verifier with sufficiently high probability, then there exists a value $x^{*}$ such that (1) the extractor can decommit com to $x^{*}$ and (2) the cheating prover cannot prove false statements about $x^{*}$.

[^5]:    ${ }^{9}$ The next-message function takes as input an internal state and incoming messages of a round, and it outputs the internal state and outgoing messages of the round. (We assume that the internal state implicitly includes all the incoming messages of the previous rounds.)

[^6]:    ${ }^{10}$ We follow the modularization by Paneth and Rothblum [PR14].
    ${ }^{11}$ Concretely, in this overview we assume that Assign is perfect no-signaling, i.e., that the RHS of the equation in Definition 10 is 0 even against computationally unbounded distinguishers.

[^7]:    ${ }^{12}$ Each $\mathrm{i}-\mathrm{msgs}_{1}^{\mu}$ is the dummy incoming messages of the first round (cf. Section 3.1).

[^8]:    ${ }^{13}$ We think that each round of $\Pi$ starts when each party receives incoming messages from the other parties, and ends when each party sends outgoing messages to the other parties.

[^9]:    ${ }^{14}$ Note that we cannot use this argument if we try to reuse the analysis of Kalai et al. [KRR14] for each $\varphi^{\mu: v}$ individually (rather than in the round-byround manner) since we show the correctness in $\varphi^{\mu: \nu}$ by using the correctness in $\varphi^{\mu: \xi}$.
    ${ }^{15}$ Concretely, a low-degree extension of $\boldsymbol{x}$ (Section C.4).

[^10]:    ${ }^{16}$ Concretely, we use layer-parallel low-degree tests [HR18] to guarantee that the initial states (resp., the views) that are recovered through selfcorrection in p-Assign do not change when the queries are sampled from $D\left(X_{1, \text { in }}^{\mu}\right)$ (resp., from $D\left(X^{\mu}\right)$ ) rather than from $D(X)$.
    ${ }^{17}$ Specifically, the receiver make queries for a low-degree test (just like the verifier of KRR succinct argument does) so that we can reuse analyses of Kalai et al. [KRR14] as in the proof of soundness.

[^11]:    ${ }^{18}$ Formally, as in the case of the non-succinct protocol in Section 1.2, complexity leveraging is required.

[^12]:    ${ }^{19}$ We remark that the schemes by Khurana et al. [KOS18] are designed to prevent such an attack. What we claim is that the definition by Khurana et al. [KOS18] does not prevent such an attack.

[^13]:    ${ }^{20}$ The adaptive delayed-input property is required to, e.g., upgrade our WI commit-and-prove protocol to a ZK one by using the transformation of Khurana et al. [KOS18].
    ${ }^{21}$ Our version of the receiver privacy might look unusual. Yet, it is easy to see that our version is implied by the standard one (e.g., the one in [BK18]).

[^14]:    ${ }^{22} \kappa_{0}$ is a polynomial in $\lambda$ and $\log N$, where $N$ is the number of the variables in the 3 CNF formula given as the statement. Note that when $N$ is expressed as a polynomial in $\lambda, \kappa_{0}$ is a polynomial in $\lambda$.

[^15]:    ${ }^{23}$ Roughly speaking, SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}^{\text {PCP. }}$, does "self-correction" (see, e.g., [Sud00, Lecture 2 , Section 4]) on $X$ by making queries to PCP.P*, where $m|\boldsymbol{H}|$ is an upper bound on the total degree of $X$.
    ${ }^{24} \kappa_{\text {max }}$ is a polynomial in $\lambda$ and $\log N$ just like $\kappa_{0}$; see Footnote 22.
    ${ }^{25} \mathcal{I}_{\text {in }}, \mathcal{I}_{\text {in,LDE }}, \mathcal{I}_{\text {out,LDE }}$ and $\mathcal{I}_{\text {out }}$ only depend on $N^{\prime}$ and $n$, and thus are fixed before $f, x$ are fixed.

[^16]:    ${ }^{26}$ Recall that we assume for simplicity that each party uses the same next message function in every round in $\Pi$.
    ${ }^{27}$ Intuitively, $\varphi_{1}$ checks whether each end state and outgoing messages are correctly computed from the corresponding start state and incoming messages.
    ${ }^{28}$ Intuitively, $\varphi_{2}$ checks whether (1) $P^{\xi}$ 's start state at round $\ell+1$ is equal to its end state at round $\ell$, (2) $P^{\nu}$ 's incoming message from $P^{\mu}$ at round $\ell+1$ is equal to $P^{\mu}$ 's outgoing message to $P^{\nu}$ at round $\ell$ and vice versa, and (3) $P^{\xi}$ 's output is 1 .

[^17]:    ${ }^{29}$ Concretely, we assume that $X^{\mu: v}$ is computed as follows. View $x^{\mu: v}$ as a function with domain $\{\mu, v\} \times\left[N_{\text {round }}\right] \times\left[N_{\text {Aug }}\right]$, and fix a mapping from $\{\mu, \nu\} \times\left[N_{\text {round }}\right] \times\left[N_{\text {Aug }}\right]$ to $\boldsymbol{H}^{m}$ by arbitrarily fixing a map from $[M]$ to $\boldsymbol{H}$, a map from $\left[N_{\text {round }}\right]$ to $\boldsymbol{H}^{m_{\text {round }}}$, and a map from $\left[N_{\text {Aug }}\right]$ to $\boldsymbol{H}^{m_{\text {Aug }} \text {. Then, let }}$ $X^{\mu: v}$ be the LDE of $x^{\mu: v}$ w.r.t. $\boldsymbol{F}, \boldsymbol{H}, m$ and the above mapping.

[^18]:    ${ }^{30}$ Note that SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}^{\mathrm{PCP}} \mathrm{P}^{*}$, outputs a function $f$ rather than a 3 CNF formula since we assume that PCP.P* outputs a function rather than a 3 CNF formula. Here, we view SelfCorr ${ }_{m|\boldsymbol{H}|, D(X)}^{\mathrm{PCP} . \mathrm{P}^{*}}$ as an a local assignment generator for $\varphi_{f}^{\alpha ; \beta}$.
    ${ }^{31}$ i.e., the execution of $\Pi$ where the functionality to be computed is $f$, the initial states of the parties are $\left\{\mathrm{st}_{0}^{\mu}\right\}_{\mu \in[M]}$, and the dummy incoming messages of the first round are $\left\{i-\mathrm{msgs}_{1}^{\mu}\right\}_{\mu \in[M]}$.

[^19]:    ${ }^{32}$ The everywhere local consistency implies that if the three variables of a clause in $\varphi_{f}^{\alpha: \beta}$ are assigned by SelfCorr, the values assigned to them are 0 or 1 with overwhelming probability. Then, CNS implies that even when each of the three variables is assigned by SelfCorr individually, the value assigned to it is still 0 or 1 with overwhelming probability.

[^20]:    ${ }^{33}$ We use $\kappa_{\mathrm{V}}$ as an upper bound on the query complexity of each test in LD-Test.

[^21]:    ${ }^{34}$ Specifically, the receiver security of OT and Equation (I.1) imply that during the execution of $\left\langle\right.$ C.Com ${ }_{4}^{*}$, R.Com $\left.{ }_{4}\right\rangle$, each trial for obtaining $\left\{\mathrm{rt}_{1, \mathrm{in}}^{\mu}, \operatorname{dec}_{\mathrm{sH}}^{\mu}\right\}_{\mu \in[M]}$ succeeds with probability at least $1 / \lambda^{c}$. Thus, from Marcov's inequality, the probability that all the $T_{\mathrm{SB}}$ trials fail is negligible.

[^22]:    ${ }^{35}$ Roughly speaking, we can still use the same security proof as Khurana et al. [KOS18] since, essentially, the only property that they use is that for any $(x, r)$ and $\left(x^{\prime}, r^{\prime}\right)$ such that $x \neq x^{\prime}$, we have $x \alpha+r \neq x^{\prime} \alpha+r^{\prime}$ with high probability when $\alpha$ is chosen uniformly randomly.

[^23]:    ${ }^{36}$ Note that $\kappa_{1}$ is a polynomial in $\lambda$ and $|\boldsymbol{F}|=$ polylog $N$. Note that when $N$ is expressed as a polynomial in $\lambda$, $\kappa_{1}$ is a polynomial in $\lambda$; cf. Footnote 22 , Footnote 24.

