Statistical ZAP Arguments

Saikrishna Badrinarayanan* Rex Fernando* Aayush Jain* Dakshita Khurana†
Amit Sahai*

Abstract

Dwork and Naor (FOCS'00) first introduced and constructed two message public coin witness indistinguishable proofs (ZAPs) for NP based on trapdoor permutations. Since then, ZAPs have also been obtained based on the decisional linear assumption on bilinear maps, and indistinguishability obfuscation, and have proven extremely useful in the design of several cryptographic primitives.

However, all known constructions of two-message public coin (or even publicly verifiable) proof systems only guarantee witness indistinguishability against computationally bounded verifiers. In this paper, we construct the first public coin two message witness indistinguishable (WI) arguments for NP with *statistical* privacy, assuming quasi-polynomial hardness of the learning with errors (LWE) assumption. Prior to this, there were no known constructions of two-message publicly verifiable WI protocols under lattice assumptions, even satisfying the weaker notion of computational witness indistinguishability.

*University of California Los Angeles.
†University of Illinois Urbana-Champaign.

Contents

1 Introduction

Witness indistinguishability (WI) is one of the most widely used notions of privacy for proof systems. Informally, WI protocols [FS90] allow a prover to convince a verifier that some statement X belongs to an NP language L, with the following privacy guarantee: if there are two witnesses w_0, w_1 that both attest to the fact that $X \in L$, then a verifier should not be able to distinguish an honest prover using witness w_0 from an honest prover using witness w_1. WI is a relaxation of zero-knowledge and has proven to be surprisingly useful. Since WI is a relaxation, unlike zero-knowledge, there are no known lower bounds on the rounds of interaction needed to build WI protocols in the plain model.

Indeed, Dwork and Naor [DN00, DN07] introduced the notion of two-message public-coin witness indistinguishable proofs (ZAPs) without any setup assumptions, and also constructed it assuming trapdoor permutations. We observe that the public-coin feature of ZAPs yield public verifiability of the resulting proof system, since a third party can use the public coins of the verifier to determine whether or not the prover's response constitutes a valid proof. Subsequently, Groth et al. [GOS06] constructed ZAPs assuming the decisional linear assumption, Bitansky and Paneth [BP15] constructed ZAPs from indistinguishability obfuscation and one way functions.

Our goal: ZAPs with statistical privacy. As originally introduced, ZAPs satisfied soundness against unbounded provers (i.e. were *proofs*), and witness indistinguishability against computationally bounded verifiers. In this work, we examine whether these requirements can be reversed: can we achieve witness indistinguishability against computationally unbounded verifiers, while achieving soundness against computationally bounded cheating provers? We call such objects *statistical ZAP arguments*.

An analogue of this question has a long history of study in the context of zero-knowledge protocols. Indeed, zero-knowledge protocols for NP were originally achieved guaranteeing privacy to hold only against computationally bounded verifiers [GMW86]. In the case of zero-knowledge, the notion of *statistical* zero-knowledge arguments was achieved soon after [BCR86, Cha86], that strengthened the privacy requirement to hold against computationally unbounded verifiers, while requiring soundness to hold only against computationally bounded provers.

Because ZAPs require a single message each from the verifier and the prover, a better comparison would perhaps be to non-interactive zero-knowledge (NIZK) [BFM88]. Even in the case of NIZKs, we have had arguments for NP satisfying statistical zero-knowledge since 2006 [GOS06]. And yet, the following natural question has remained open since the introduction of ZAPs nearly two decades ago.

Do there exist statistical ZAP arguments for NP in the plain model?

Statistical witness indistinguishability, just like its zero-knowledge counterpart, guarantees *everlasting privacy* against malicious verifiers, long after protocols have completed execution. Of course, to achieve statistical privacy, we must necessarily sacrifice soundness against unbounded provers. But such a tradeoff could often be desirable, since soundness is usually necessary only in an online setting: in order to convince a verifier of a false statement, a cheating prover must find a way to cheat *during* the execution of the protocol.

The main challenge: achieving a public-coin protocol. The recent work of Kalai et al. [KKS18] constructed the first two message statistically witness indistinguishable arguments in the plain model under standard sub-exponential assumptions. However, their arguments are only *privately verifiable*.

The blueprint of [KKS18], which builds on other similar approaches in the computational witness indistinguishability setting [BGI$^+$17, JKKR17], uses oblivious transfer (OT) to compress a Σ-protocol. In all these approaches, the verifier obtains the third message of the Σ-protocol via the

output of the OT, and therefore these approaches fundamentally require the use of private coins for verification. It is also worth noting that these protocols are not sound against provers that have access to the private coins of the verifier, which restricts their applicability. Additionally, the verifier's message is *not reusable*, which means that soundness is not guaranteed if the same verifier message is reused across multiple executions.

On the other hand, a *public coin* argument, which is the focus of this work, does not suffer from any of these limitations. In fact, where the verifier's message only needs to be a uniformly random string. Such a string can easily be generated, for example, via an MPC protocol, and can then be reused across multiple executions with no loss in soundness.

We stress that prior to our work, even two message statistically witness indistinguishable arguments that were only publicly verifiable (and not necessarily public coin) were not known.

1.1 Our Results

In this paper, we construct the first two message public coin statistically witness indistinguishable arguments for NP in the plain model. Our constructions assume quasi-polynomial hardness of the learning with errors (LWE) problem. In fact, these are the first known two-message public coin (or even publicly verifiable) arguments based on lattice assumptions, satisfying *any notion of witness indistinguishability* (computational/statistical). We provide an informal theorem below.

Informal Theorem 1. *Assuming quasi-polynomial hardness of the learning with errors (LWE) assumption, there exist two message public-coin statistically witness indistinguishable arguments for NP in the plain model.*

Our results are obtained by combining two recent results in a new way: recent constructions of correlation-intractable hash functions based on LWE [CCH$^+$19] and the statistically hiding extractable commitments of [KKS18] (which are built upon [KS17]). This yields a new method of using correlation intractable hash functions to instantiate the Fiat-Shamir transform, by extracting messages from *statistically hiding commitments*, instead of from statistically binding trapdoor commitments – that we believe may be of independent interest.

2 Overview of Techniques

In this section, we provide a brief overview of the techniques we use to build a two message public coin statistical WI argument (henceforth referred to as a ZAP).

Our starting point is the popular technique to construct ZAPs for NP, due to Dwork and Naor [DN02]. Their construction makes use of a statistically sound NIZK in the common random string model, and can be described as follows.

- In the first round, the verifier picks uniformly random strings $\mathsf{crs}_1, ..., \mathsf{crs}_\lambda$, where λ denotes the security parameter, and sends them to the prover.
- In the second round, the prover samples a uniformly random string crs'. It computes proofs $(\pi_1, ..., \pi_\ell)$ where π_i is a NIZK proof for the instance x that verifies under $\mathsf{crs}'_i = \mathsf{crs}' \oplus \mathsf{crs}_i$ The prover sends crs' along with proof strings $(\pi_1, ..., \pi_\ell)$ to the verifier.

The soundness of this protocol can be proven based on the statistical soundness of NIZK, in the following way. Fix an instance $x \notin L$. Statistical soundness of the NIZK implies that with probability at least $1/2$ over the choice of crs from the domain of the common random string of NIZK, *there does not exist a proof* π that verifies for instance x with respect to crs. Put another way, for fixed

x, for at least $1/2$ of the strings in the domain of the common random string of the NIZK, *there does not exist a proof for* x. One can use this fact to argue combinatorially that over the choice of random $\mathsf{crs}_1, ..., \mathsf{crs}_\lambda$, the probability that there exists crs' for which there exist proofs with respect to every member of the set $\{\mathsf{crs}'_i = \mathsf{crs}' \oplus \mathsf{crs}_i\}_{i\in[\ell]}$, is negligible.

The proof of witness indistinguishability follows quite simply, by switching the witness in each of the proofs one by one.

But when applied to our context, this approach immediately encounters the following problems.

1. The soundness argument outlined above crucially requires that with high probability over the CRS of the NIZK, there just should not exist a proof for any fixed false instance. This translates to requiring *statistical soundness* of the underlying NIZK.

2. One cannot hope to get a WI argument secure against unbounded verifiers via this transform, unless the underlying NIZK also satisfies privacy against unbounded verifiers, i.e. satisfies statistical zero-knowledge.

3. It is believed that statistically sound and statistical zero-knowledge NIZKs for all of NP cannot exist.

4. Even if we only desired *computational* witness indistinguishability based on lattice assumptions, no statistically sound NIZKs in the common random string model are known from lattice assumptions.

As an intermediate objective, we will first try to tackle problem #4 and build a publicly verifiable computational WI argument based on LWE.

2.1 A Simple Two-Message Public-Coin Computational WI Argument

We make a few modifications the template above so as to obtain a publicly verifiable computational WI argument based on LWE.

Before we describe these modifications, we list a few ingredients. We will assume that there exists a dense public key encryption scheme PKE, that is, a scheme for which every string in $\{0,1\}^{|pk|}$ corresponds to a valid public key (and therefore every string has a valid secret key). We will further assume the existence of a correlation intractable hash function family. Informally, a hash function family $\mathcal{H}$ is correlation-intractable for a function family $\mathcal{F}$ if:

- Given a fixed function $f \in \mathcal{F}$, and a randomly generated key K (that can depend on f), the probability that an adversary outputs x such that $(x, \mathcal{H}(K, x)) = (x, f(x))$ is at most ϵ.

- The hash key K statistically hides the function f, such that adversaries cannot distinguish a random key from a key for f with advantage better than ϵ.

We will set $\epsilon = 2^{-2|pk|}$. We will use Π to denote a parallel repetition of Blum's Σ-protocol for Graph Hamiltonicity, represented as $\{\mathsf{com}(a_i)\}_{i\in[\lambda]}, \{e_i\}_{i\in[\lambda]}, \{z_i\}_{i\in[\lambda]}$, where $\{a_i\}_{i\in[\lambda]}$ represents the first committed message sent by the prover, $\{e_i\}_{i\in[\lambda]}$ is a challenge string sent by the verifier and $\{z_i\}_{i\in[\lambda]}$ represents the corresponding third message by the prover. Let the instance be x and its witness be w. Then, the protocol is described as follows.

1. In the first round, the verifier randomly samples a key K for the correlation intractable hash function $\mathcal{H}$ for bounded size NC_1 functions.

2. In the second round, the prover picks a key pair (pk, sk) for the scheme PKE. Then the prover uses $\mathsf{PKE.Enc}(pk, \cdot)$ as a commitment scheme to commit to $\{a_i\}_{i\in[\lambda]}$. Next, the prover computes $e = \mathcal{H}(K, x, \{a_i\}_{i\in[\lambda]}) \in \{0,1\}^\lambda$, and uses (x, w, a, e) to compute $z = (z_1, ..., z_\lambda)$ according to the protocol Π. It outputs $(pk, \{\mathsf{PKE.Enc}(pk, a_i)\}_{i\in[\lambda]}, e, z)$

While witness indistinguishability of this protocol is easy to see, arguing soundness is trickier. In order to argue soundness, the reduction will simple try to *guess* the public key pk^* that the prover will use, and will abort if this guess is not correct. Note that such a guess is correct with probability at least $2^{-|pk^*|}$.

Suppose a cheating prover convinces a verifier to accept false statements with probability $\frac{1}{p(\lambda)}$ for some polynomial $p(\cdot)$. Then, with probability at least $\frac{1}{p(\cdot)} \cdot 2^{-|pk^*|}$, the reduction guesses pk^* correctly, and the prover provides a convincing proof of a false statement using pk^*.

In the next hybrid, the challenger guesses pk^* together with the corresponding secret key sk^*, and then samples a correlation intractable hash key for a specific function $f_{sk^*}(\cdot)$. The function $f_{sk^*}(\cdot)$ on input x, along with a (the messages committed in the Σ-protocol), outputs the only possible string e_{bad} for which there exists a string z such that (a, e_{bad}, z) verifies for $x \notin L$[1]. Note that this function is in NC_1. By ϵ-security of the correlation intractable hash family (where $\epsilon = 2^{-2|pk|}$), with probability at least $\left(\frac{1}{p(\cdot)} - 2^{-|pk|}\right) \cdot 2^{-|pk|}$, the reduction guesses pk^* correctly, and the prover provides a convincing proof of a false statement using pk^*.

Finally, since the correlation intractable hash function is ϵ-secure, in the final hybrid adversary cannot produce a proof for x with probability greater than ϵ, as this will mean that he output a^*, e^*, z^* such that $e^* = f_{bad}(x, a^*)$.

The protocol sketched above is public-coin, because when we instantiate the correlation-intractable hash family with the LWE-based one by [PS19], the hash keys are statistically close to uniform.

In the description above, we also relied on a dense public key encryption scheme, which is unfortunately not known to exist based on LWE. However, we note that we can instead use a scheme with the property that at least $1/2$ of the strings in $\{0,1\}^{\ell_{\mathsf{PKE}}}$ correspond to correct encryption keys with a valid secret key, and the property that public keys are pseudorandom. Then, the verifier sends λ public keys $pk_1, \ldots, pk_\lambda$, and the prover outputs pk', and then uses the public keys $\{(pk' \oplus pk_i)\}_{i\in[\lambda]}$ to compute λ proofs. Soundness can be obtained by arguing that with overwhelming probability, there will exist an index $i \in [\lambda]$ such that $(pk' \oplus pk_i)$ has a secret key, just like the [DN02] technique described at the beginning of this overview.

However, the construction above falls short of achieving statistical witness indistinguishability against malicious verifiers. The reason is the following: arguing that the construction described above satisfies soundness requires relying on correlation intractability of the hash function. In order to invoke the correlation intractable hash function, it is crucial that the prover be "committed" to a well-defined, unique message $\{a_i\}_{i\in[\lambda]}$, that can be extracted using the secret key sk^* of the public key encryption scheme. At first, statistical hiding, together with such extraction, may appear to be contradictory objectives.

Indeed, we will try obtain a weaker version of these contradictory objectives, and specifically, we will rely on a two-message statistically hiding extractable commitment scheme [KKS18].

[1]Note that this property is satisfied by any Σ-Protocol with a $1/2-$special soundness, such as Blum's Σ-protocol.

2.2 Using Correlation-Intractable Hashing with Statistically Hiding Extractable Commitments

In the recent exciting work on using LWE-based correlation-intractable hashing [PS19, CCH+19] for achieving soundness, as well as in the "warm up" ZAP protocol described above, the correlation-intractable hash function is used as follows. Because the LWE-based CI-hash function is designed to avoid an *efficiently computable* function f of the prover's first message, it is used together with a public-key encryption scheme: the prover's first message is encrypted using the public key, and the function f is built to contain the secret key of the encryption scheme, so that it can decrypt the prover's first message in order to calculate the challenge that must be avoided.

Our work imagines a simple modification of this strategy of using correlation-intractable hashing for arguing soundness. The main idea is that we want to replace the encryption scheme (which necessarily can only at most provide computational hiding) with an *extractable* statistically hiding commitment scheme. We will describe what this object entails in more detail very shortly, but the main observation is that such an extractable commitment in fact reveals the value being committed to with a tiny (but tunable) probability – crucially in a way that prevents a malicious prover from learning whether the commitment will reveal the committed value or not. With such a commitment scheme, the efficient function f underlying the correlation-intractable hash function will only "work" in the rare case that the commitment reveals the value being committed. But since a cheating prover can't tell whether its committed values will be revealed or not, soundness will still hold overall, even though the actual guarantee of the correlation-intractable hash function is only invoked with a tiny probability in the proof of soundness. We now elaborate.

2.3 Statistically Hiding Extractable Commitments

Any statistically hiding commitment must lose all information about the committed message, except with negligible probability. This makes it challenging to define notions of extraction for statistically hiding commitments. In 4 rounds or more, this notion is easier to define, as extraction is possible even from statistically hiding commitments, simply by rewinding the adversary. However, traditional rewinding techniques break down completely when considering two-message commitments.

Nevertheless, the recent work of [KKS18], building on [KS17], defined and constructed two-message statistically hiding extractable commitments, which they used to construct two-message statistical WI arguments, that were *privately verifiable*. In what follows, we abstract out the properties of a statistically hiding extractable commitment. A more formal description can be found in Section 5. We point out that we only need to rely on significantly simpler definitions than the ones in [KKS18], and we give much simpler proofs that the constructions in [KKS18] according to our new definitions. This may be of independent interest.

Defining Statistically Hiding Extractable Commitments. We start with an important observation about statistically hiding commitments, which gives a hint about how one can possibly define (and construct) two-message statistically hiding extractable commitments. Namely, any statistically hiding commitment must lose all information about the committed message, *but may retain this information with some small negligible probability.* Specifically,

- A commitment that leaks the committed message with probability ϵ (where ϵ is a fixed negligible function in the security parameter) and statistically hides the message otherwise, will continue to be statistically hiding.

- At the same time, one could ensure that no matter the behavior of the committer, the message being committed *does get leaked to the honest receiver with probability at least* ϵ.
- Moreover, the committer does not know whether or not the committed message was leaked to the receiver. This property is important and will be crucially used in our proofs.

In spirit, this corresponds to establishing an erasure channel over which the committer transmits his message to the receiver. This channel almost always erases the committed message, but is guaranteed to transmit the committed message with a very small probability (ϵ). Moreover, just like cryptographic erasure channels, the committer does not know whether or not his message was transmitted. Additionally, because this is a commitment, we require computational binding: once the committer transmits his message (that is, commits), he should not be able to change his mind about the message, *even if* the message did not get transmitted. Finally, we say that "extraction occurs" whenever the message does get transmitted, and we require that extraction occur with probability at least ϵ, even against a malicious committer.

Next, we describe how we interface these commitments with correlation intractable hash functions to obtain two-message statitical ZAP arguments.

2.4 Statistical ZAP Arguments

With this tool in mind, we make the following observations:

1. We would like to replace the encryption scheme used for generating the first message a for the sigma protocol, sent by the prover in second round, with a statistically hiding commitment.
2. The first message of this commitment will be generated by the verifier. Furthermore, because we want a public coin protocol, we require this message to be pseudorandom.
3. We will require that with some small probability (say $\lambda^{-\omega(\log \lambda)}$), *all* messages committed by the prover get transmitted to the verifier, that is with probability $\lambda^{-\omega(\log \lambda)}$, the verifier can recover all the messages committed by the prover in polynomial time given his secret state. Next, using an insight from the simple protocol in Section 2.1, we will set the security of the correlation intractable hash function, so that it is infeasible for any polynomially sized adversary to break correlation intractability with probability $\lambda^{-\omega(\log \lambda)}$.

The protocol is then as follows:

- In the first round, the verifier samples a hash key K for the correlation intractable hash function $\mathcal{H}$, for the same function family $\mathcal{F}$ as Section 2.1. The verifier also samples strings $q = \{c_{1,j}\}_{j \in [\mathsf{poly}(\lambda)]}$ uniformly at random, where poly is a polynomial denoting the number of commitments made by the prover. The verifier sends q and K over to the prover.
- In the second round, the prover computes the first message of the sigma protocol a (where the number of parallel repetitions equals the output length of correlation intractable hash function). This message a is generated using the statistically hiding extractable commitment scheme com with q as the first message. The prover computes $e = \mathcal{H}(K, x, q, a)$ and uses e to compute the third message z of the sigma protocol, by opening some subset of the commitments made by the prover. The prover outputs (a, e, z).

We now provide some intuition for the security of this protocol.

- **Soundness:** To argue soundness, we follow an approach that is similar to the soundness proof for the computational ZAP argument described in Section 2.1 (although with some additional technical subtleties). We discuss one such subtlety here:

  Let $\ell = |e|$. Then, the correlation-extractable hash function can be at most $2^{-\ell^\delta}$-secure[2]. For this reason, we require the commitments to be *jointly* extractable in polynomial time with probability at least $2^{-\ell^\delta}$. Note that the total number of commitments is $N = \ell \cdot \mathsf{poly}(\lambda)$.

  However, statistically hiding commitments, as originally constructed in [KKS18], are such that if a single commitment can be extracted with probability ϵ, then N commitments can be extracted with probability roughly ϵ^N. Setting $N = \ell \cdot \mathsf{poly}(\lambda)$ as above implies that trivially, the probability of extraction will be roughly $O(2^{-\ell \cdot \mathsf{poly}(\lambda)})$, which is smaller than the required probability $2^{-\ell^\delta}$.

  However, we observe that the commitments constructed in [KKS18] can be modified very slightly so that the probability of extraction can be $2^{-g(\lambda)}$ for any efficiently computable function g that is bounded by any polynomial in λ. Thus, for example, the probability of extraction can be made to be $\lambda^{-\log(\lambda)}$. In other words, this extraction probability can be made to be *independent of the total number of commitments, N*. We describe this modification in additional detail in Section 4.2.

  Using commitments that satisfy the property stated above, we observe that we can switch to a hybrid where the challenger samples the commitment messages on behalf of the verifier, and hardwires the secret state used for extraction inside the hash key. The function is defined such that in the event that extraction occurs (given the secret state), the verifier can use the extracted values to compute the bad challenge e_{bad} (just as in Section 2.1), by evaluating a depth bounded function f_{bad} on the extracted values, and otherwise e_{bad} is set to 0. If the adversary breaks soundness with noticeable probability ϵ, then with probability roughly at least $2^{-g(\lambda)} \cdot \epsilon$, the outputs of the adversary satisfy:

$$\mathcal{H}(K, x, q, a) = e_{bad}$$

  As already alluded to previously, we set the function g and the (quasi-polynomial) security of the hash function such that the event above suffices to contradict correlation intractability.

- **Statistical Witness Indistinguishability:** Statistical witness indistinguishability composes under parallel repetition, and therefore can be proven index-by-index based on the statistical hiding property of the commitment.

  Additional details about the construction and the proof can be found in Section 5.

2.5 Organization

The rest of this paper is organized as follows. In Section 3, we describe some of the preliminaries such as correlation intractability, oblivious transfer and proof systems. In Section 4, we define a simplified variant and present a slightly modified construction of extractable statistically hiding commitments, first proposed by [KKS18]. Finally, in Section 5, we construct and prove the security of our statistical ZAP argument.

[2]More formally, if the output of the hash function is ℓ bits long, then even if we rely on sub-exponential assumptions, we cannot hope to have the guessing advantage be smaller than $2^{-\ell^\delta}$ for a small positive constant $\delta < 1$.

3 Preliminaries

Notation. Throughout this paper, we will use λ to denote the security parameter, and $\mathsf{negl}(\lambda)$ to denote any function that is asymptotically smaller than $\frac{1}{\mathsf{poly}(\lambda)}$ for any polynomial $\mathsf{poly}(\cdot)$.

The statistical distance between two distributions D_1, D_2 is denoted by $\Delta(D_1, D_2)$ and defined as:

$$\Delta(D_1, D_2) = \frac{1}{2}\Sigma_{v \in V} |\Pr_{x \leftarrow D_1}[x = v] - \Pr_{x \leftarrow D_2}[x = v]|.$$

We say that two families of distributions $D_1 = \{D_{1,\lambda}\}, D_2 = \{D_{2,\lambda}\}$ are statistically indistinguishable if $\Delta(D_{1,\lambda}, D_{2,\lambda}) = \mathsf{negl}(\lambda)$. We say that two families of distributions $D_1 = \{D_{1,\lambda}\}, D_2 = \{D_{2,\lambda}\}$ are computationally indistinguishable if for all non-uniform probabilistic polynomial time distinguishers $\mathcal{D}$,

$$\left|\Pr_{r \leftarrow D_{1,\lambda}}[\mathcal{D}(r) = 1] - \Pr_{r \leftarrow D_{2,\lambda}}[\mathcal{D}(r) = 1]\right| = \mathsf{negl}(\lambda).$$

Let Π denote an execution of a protocol. We use $\mathsf{View}_A(\Pi)$ denote the view, including the randomness and state of party A in an execution Π. We also use $\mathsf{Output}_A(\Pi)$ denote the output of party A in an execution of Π.

Remark 1. *In what follows we define several 2-party protocols. We note that in all these protocols both parties take as input the security parameter 1^λ. We omit this from the notation for the sake of brevity.*

Definition 1 (Σ-protocols)**.** *Let $L \in \mathsf{NP}$ with corresponding witness relation R_L. A protocol $\Pi = \langle P, V \rangle$ is a Σ-protocol for relation R_L if it is a three-round public-coin protocol which satisfies:*

- **Completeness:** *For all $(x, w) \in R_L$, $\Pr[\mathsf{Output}_V\langle P(x, w), V(x)\rangle = 1] = 1 - \mathsf{negl}(\lambda)$, assuming P and V follow the protocol honestly.*
- **Special Soundness:** *There exists a polynomial-time algorithm A that given any x and a pair of accepting transcripts $(a, e, z), (a, e', z')$ for x with the same first prover message, where $e \neq e'$, outputs w such that $(x, w) \in R_L$.*
- **Honest verifier zero-knowledge:** *There exists a probabilistic polynomial time simulator $\mathcal{S}_\Sigma$ such that for all $(x, w) \in R_L$, the distributions $\{\mathcal{S}_\Sigma(x, e)\}$ and $\{\mathsf{View}_V\langle P(x, w(x)), V(x, e)\rangle\}$ are statistically indistinguishable. Here $\mathcal{S}_\Sigma(x, e)$ denotes the output of simulator $\mathcal{S}$ upon input x and e, such that V's random tape (determining its query) is e.*

3.1 Correlation Intractable Hash Functions

We adapt definitions of a correlation intractable hash function family from [PS19, CCH$^+$19].

Definition 2. *For any polynomials $k, (\cdot), s(\cdot) = \omega(k(\cdot))$ and any $\lambda \in \mathbb{N}$, let $\mathcal{F}_{\lambda,s(\lambda)}$ denote the class of NC^1 circuits of size $s(\lambda)$ that on input $k(\lambda)$ bits output λ bits. Namely, $f : \{0,1\}^{k(\lambda)} \to \{0,1\}^\lambda$ is in $\mathcal{F}_{\lambda,s}$ if it has size $s(\lambda)$ and depth bounded by $O(\log \lambda)$.*

Definition 3. *[Quasi-polynomially Correlation Intractable Hash Function Family] A hash function family $\mathcal{H} = (\mathsf{Setup}, \mathsf{Eval})$ is quasi-polynomially correlation intractable (CI) with respect to $\mathcal{F} = \{\mathcal{F}_{\lambda,s(\lambda)}\}_{\lambda \in \mathbb{N}}$ as defined in Definition 2, if the following two properties hold:*

- ***Correlation Intractability:*** *For every $f \in \mathcal{F}_{\lambda,s}$, every non-uniform polynomial-size adversary $\mathcal{A}$, every polynomial s, and every large enough $\lambda \in \mathbb{N}$,*

$$\Pr_{K \leftarrow \mathcal{H}.\mathsf{Setup}(1^\lambda, f)}\Big[\mathcal{A}(K) \to x \text{ such that } (x, \mathcal{H}.\mathsf{Eval}(K, x)) = (x, f(x))\Big] \leq \frac{1}{\lambda^{(\omega(\log^* \lambda))^2}}.$$

- ***Statistical Indistinguishability of Hash Keys:*** *Moreover, for every $f \in \mathcal{F}_{\lambda,s}$, for every non-uniform polynomial-size adversary $\mathcal{A}$,and every large enough $\lambda \in \mathbb{N}$,*

$$\left|\Pr_{K \leftarrow \mathcal{H}.\mathsf{Setup}(1^\lambda, f)}[\mathcal{A}(K) = 1] - \Pr_{K \leftarrow \{0,1\}^\ell}[\mathcal{A}(K) = 1]\right| \leq 2^{-\lambda^{O(1)}},$$

where ℓ denotes the size of the output of $\mathcal{H}.\mathsf{Setup}(1^\lambda, f)$.

The work of [PS19] gives a construction of correlation intractable hash functions with respect to $\mathcal{F} = \{\mathcal{F}_{\lambda,s(\lambda)}\}_{\lambda\in\mathbb{N}}$, based on polynomial LWE with polynomial approximation factors. We observe that their construction also satisfies Definition 3, assuming quasi-polynomial LWE with polynomial approximation factors.

3.2 Oblivious Transfer

Definition 4 (Oblivious Transfer)**.** *Oblivious transfer is a protocol between two parties, a sender S with input messages (m_0, m_1) and receiver R with input a choice bit b. The correctness requirement is that R obtains output m_b at the end of the protocol (with probability 1). We let $\langle S(m_0, m_1), R(b)\rangle$ denote an execution of the OT protocol with sender input (m_0, m_1) and receiver input bit b. We require OT that satisfies the following properties:*

- **Computational Receiver Security.** *For any non-uniform PPT sender S^* and any $(b, b') \in \{0, 1\}$, the views $\mathsf{View}_{S^*}(\langle S^*, R(b)\rangle)$ and $\mathsf{View}_{S^*}(\langle S^*, R(b')\rangle)$ are computationally indistinguishable.*

  We say that the OT scheme is T-secure if all PPT malicious senders have distinguishing advantage less than $\frac{1}{T}$.

- **Statistical Sender Security.** *This is defined using the real-ideal paradigm, and requires that for any distribution on the inputs (m_0, m_1) and any unbounded* adversarial *receiver R^*, there exists a (possibly unbounded) simulator Sim_{R^*} that interacts with an ideal functionality $\mathcal{F}_{\mathsf{ot}}$ on behalf of R^*. Here $\mathcal{F}_{\mathsf{ot}}$ is an oracle that obtains the inputs (m_0, m_1) from S and b from Sim_{R^*} (simulating the malicious receiver), and outputs m_b to Sim_{R^*}. Then $\mathsf{Sim}_{R^*}^{\mathcal{F}_{\mathsf{ot}}}$ outputs a receiver view that is statistically indistinguishable from the real view of the malicious receiver $\mathsf{View}_{R^*}(\langle S(m_0, m_1), R^*\rangle)$. We say that the OT protocol satisfies $(1 - \delta)$ statistical sender security if the statistical distance between the real and ideal distributions is at most δ.*

We use the following sender security property in our protocols (which follows from the definition of sender security in Definition 4 above).

Claim 1. *For any two-message OT protocol satisfying Definition 4, for every malicious receiver R^* and every "valid" first message m_{R^*} generated by R^*, we require that there exists an unbounded machine E which extracts b such that either of the following statements is true:*

- *For all m_0, m_1, m_2, $\mathsf{View}_{R^*}\langle S(m_0, m_1), R^*\rangle$ and $\mathsf{View}_{R^*}\langle S(m_0, m_2), R^*\rangle$ are statistically indistinguishable and $b = 0$, or,*

- *For all m_0, m_1, m_2, $\mathsf{View}_{R^*}\langle S(m_0, m_1), R^*\rangle$ and $\mathsf{View}_{R^*}\langle S(m_2, m_1), R^*\rangle$ are statistically indistinguishable and $b = 1$.*

Proof. From the (unbounded) simulation property of the two-message OT, there exists a simulator that extracts a receiver input bit b from the first message of R^*, sends it to the ideal functionality, obtains m_b and generates an indistinguishable receiver view. Then, by the definition of

sender security, when $b = 0$, the simulated view must be close to both $\mathsf{View}_{R^*}\langle S(m_0, m_1), R^*\rangle$, and $\mathsf{View}_{R^*}\langle S(m_0, m_2), R^*\rangle$. Similarly, when $b = 1$, the simulated view must be statistically close to both $\mathsf{View}_{R^*}\langle S(m_0, m_1), R^*\rangle$, and $\mathsf{View}_{R^*}\langle S(m_2, m_1), R^*\rangle$. □

Throughout the paper, we focus on two-message oblivious transfer. We now discuss an additional specific property of two-message OT protocols.

Property 1. *The message sent by the receiver is pseudorandom - in particular, this means that the receiver can just sample and send a uniformly random string as a valid message to the sender.*

Such two-message OT protocols with this additional property have been constructed based on the DDH assumption [NP01], LWE assumption [BD18], and a stronger variant of smooth-projective hashing, which can be realized from DDH as well as the N^{th}-residuosity and Quadratic Residuosity assumptions [Kal05, HK12]. Such two-message protocols can also be based on witness encryption or indistinguishability obfuscation (iO) together with one-way permutations [SW14].

3.3 Proof Systems

Delayed-Input Interactive Protocols. An n-message delayed-input interactive protocol for deciding a language L with associated relation R_L proceeds in the following manner:

- At the beginning of the protocol, P and V receive the size of the instance and security parameter, and execute the first $n - 1$ messages.
- Before sending the last message, P receives input $(x, w) \in R_L$. P sends x to V together with the last message of the protocol. Upon receiving the last message from P, V outputs 1 or 0.

An execution of this protocol with instance x and witness w is denoted by $\langle P(x, w), V(x)\rangle$. A delayed-input interactive protocol is a protocol satisfying the completeness and soundness condition in the delayed input setting. One can consider both proofs – with soundness against unbounded provers, and arguments – with soundness against computationally bounded provers. In particular, a delayed-input interactive argument satisfies *adaptive soundness* against malicious PPT provers. That is, soundness is required to hold even against PPT provers who choose the statement adaptively (maliciously), depending upon the first $n - 1$ messages of the protocol.

Definition 5 (Delayed-Input Interactive Arguments)**.** *An n-message delayed-input interactive protocol (P, V) for deciding a language L is an interactive argument for L if it satisfies the following properties:*

- **Completeness:** *For every $(x, w) \in R_L$,*

$$\Pr\big[\mathsf{Output}_V\langle P(x, w), V(x)\rangle = 1\big] = 1 - \mathsf{negl}(\lambda),$$

  where the probability is over the random coins of P and V, and where in the protocol V receives x together with the last message of the protocol.

- **Adaptive Soundness:** *For every (non-uniform) PPT prover P^* that given 1^λ chooses an input length 1^p, and then chooses $x \in \{0, 1\}^p \setminus L$ adaptively, depending upon the transcript of the first $n - 1$ messages,*

$$\Pr\big[\mathsf{Output}_V\langle P^*, V\rangle(x) = 1\big] = \mathsf{negl}(\lambda),$$

  where the probability is over the random coins of V.

Witness Indistinguishability. A proof system is witness indistinguishable if for any statement with at least two witnesses, proofs computed using different witnesses are indistinguishable. In this paper, we only consider statistical witness indistinguishability, which we formally define below.

Definition 6 (Statistical Witness Indistinguishability)**.** *A delayed-input interactive argument (P, V) for a language L is said to be* statistical witness-indistinguishable *if for every unbounded verifier V^*, every polynomially bounded function $n = n(\lambda) \leq \mathsf{poly}(\lambda)$, and every $(x_n, w_{1,n}, w_{2,n})$ such that $(x_n, w_{1,n}) \in R_L$ and $(x_n, w_{2,n}) \in R_L$ and $|x_n| = n$, the following two ensembles are statistically indistinguishable:*

$$\left\{\mathsf{View}_{V^*}\langle P(x_n, w_{1,n}), V^*(x_n)\rangle\right\} \text{ and } \left\{\mathsf{View}_{V^*}\langle P(x_n, w_{2,n}), V^*(x_n)\rangle\right\}$$

Zero-Knowledge with Super-polynomial Simulation. We now define zero-knowledge with super-polynomial simulation in the same way as [Pas03], except we prove statistical security against malicious verifiers.

Definition 7 (Statistical ZK with Super-polynomial Simulation)**.** *We say that a two message protocol (P, V) for an* NP *language L is statistical zero-knowledge with super-polynomial T_{Sim}-time simulation, if it satisfies the following properties:*

- **Delayed-Input Completeness.** *For every $(x, w) \in R_L$, $\Pr[\mathsf{Output}_V\langle P(x, w), V(x)\rangle] = 1 - \mathsf{negl}(\lambda)$, where the probability is over the random coins of P and V.*
- **Adaptive Soundness.** *For every every non-uniform polynomial-size P^*, that upon receiving a security parameter 1^λ chooses an instance length 1^p, and chooses the instance $x \in \{0,1\}^p$ adaptively after observing the verifier's message, it holds that*

  $$\Pr[\mathsf{Output}_V\langle P^*, V\rangle(x) = 1 \ \wedge \ x \notin L] = \mathsf{negl}(\lambda),$$

  where the probability is over the random coins of V.
- **Statistical Zero-Knowledge.** *There exists a (uniform) simulator $\mathcal{S}$ that runs in time T_{Sim}, such that for every polynomial $n = n(\lambda) \leq \mathsf{poly}(\lambda)$, and for every $(x_n, w_n) \in R_L$ where each $|x_n| = n$, and every unbounded verifier V^*, the two distributions $\mathcal{S}^{V^*}(x_n)$ and $\mathsf{View}_{V^*}\langle P(x_n, w_n), V^*(x_n)\rangle$ are statistically close.*

4 Extractable Commitments

4.1 Definitions

We take the following definition of statistically hiding extractable commitments from [KKS18]. As before, we use λ to denote the security parameter, and we let $p = \mathsf{poly}(\lambda)$ be an arbitrary fixed polynomial such that the message space is $\{0,1\}^{p(\lambda)}$.

We restrict ourselves to commitments with non-interactive decommitment, and where the (honest) receiver is not required to maintain any state at the end of the commit phase in order to execute the decommit phase. Our construction will satisfy this property and this will be useful in our applications to constructing statistically private arguments.

Definition 8 (Statistically Hiding Commitment Scheme)**.** *A commitment $\langle \mathcal{C}, \mathcal{R}\rangle$ is a two-phase protocol between a committer $\mathcal{C}$ and receiver $\mathcal{R}$, consisting of algorithms* $\mathsf{Commit}, \mathsf{Decommit}$ *and* Verify*. At the beginning of the protocol, $\mathcal{C}$ obtains as input a message $M \in \{0,1\}^p$. Next, $\mathcal{C}$ and $\mathcal{R}$*

execute the commit phase, and obtain a commitment transcript, denoted by τ, together with private states for $\mathcal{C}$ and $\mathcal{R}$, denoted by $\mathsf{state}_{\mathcal{C},\tau}$ and $\mathsf{state}_{\mathcal{R},\tau}$ respectively. We use the notation

$$(\tau, \mathsf{state}_{\mathcal{C},\tau}, \mathsf{state}_{\mathcal{R},\tau}) \leftarrow \mathsf{Commit}\langle \mathcal{C}(M), \mathcal{R}\rangle.$$

Later, $\mathcal{C}$ and $\mathcal{R}$ possibly engage in a decommit phase, where the committer $\mathcal{C}$ computes and sends message $y = \mathsf{Decommit}(\tau, \mathsf{state}_{\mathcal{C},\tau})$ to $\mathcal{R}$. At the end, $\mathcal{R}$ computes $\mathsf{Verify}(\tau, y)$ to output $\perp$ or a message $\widetilde{M} \in \{0,1\}^p$.[3]

A statistically hiding commitment scheme is required to satisfy three properties:

- **(Perfect) Completeness.** *If $\mathcal{C}, \mathcal{R}$ honestly follow the protocol, then for every $M \in \{0,1\}^p$:*

  $$\Pr[\mathsf{Verify}(\tau, \mathsf{Decommit}(\tau, \mathsf{state}_{\mathcal{C},\tau})) = M] = 1$$

  where the probability is over $(\tau, \mathsf{state}_{\mathcal{C},\tau}) \leftarrow \mathsf{Commit}\langle \mathcal{C}(M), \mathcal{R}\rangle$.

- **Statistical Hiding.** *For every two messages $M_1, M_2 \in \{0,1\}^{2p}$, every unbounded malicious receiver $\mathcal{R}^*$ and honest committer $\mathcal{C}$, a commitment is $\delta(\lambda)$-statistically hiding if the statistical distance between the distributions $\mathsf{View}_{\mathcal{R}^*}(\mathsf{Commit}\langle \mathcal{C}(M_1), \mathcal{R}^*\rangle)$ and $\mathsf{View}_{\mathcal{R}^*}(\mathsf{Commit}\langle \mathcal{C}(M_2), \mathcal{R}^*\rangle)$ is at most $\delta(\lambda)$. The scheme is statistically hiding if $\delta(\lambda) \leq \frac{1}{\mathsf{poly}(\lambda)}$ for every polynomial $\mathsf{poly}(\cdot)$.*

- **Computational Binding.** *Consider any non-uniform PPT committer $\mathcal{C}^*$ that produces $\tau \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$, and then outputs y_1, y_2. Let $\widetilde{M}_1 = \mathsf{Verify}(\tau, y_1)$ and $\widetilde{M}_2 = \mathsf{Verify}(\tau, y_2)$. Then, we require that*

  $$\Pr\big[(\widetilde{M}_1 \neq \perp) \ \wedge \ (\widetilde{M}_2 \neq \perp) \ \wedge \ (\widetilde{M}_1 \neq \widetilde{M}_2)\big] = \mathsf{negl}(\lambda),$$

  over the randomness of sampling $\tau \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$.

We also define an extractor $\mathcal{E}$ that given black-box access to $\mathcal{C}^*$, and then without executing any decommitment phase with $\mathcal{C}^*$, outputs message $\widetilde{M}$ committed by $\mathcal{C}^*$ with probability at least ϵ: we require "correctness" of this extracted message $\widetilde{M}$. We also require that no PPT adversary can distinguish transcripts where extraction is successful from those where it is unsuccessful. This is formally described in Definition 9.

Definition 9 (ϵ-Extractable Statistically Hiding Commitment)**.** *We say that a statistically hiding commitment scheme is ϵ-extractable if the following holds: Denote $(\tau, \mathsf{state}_{\mathcal{C},\tau}, \mathsf{state}_{\mathcal{R},\tau}) \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$. We require that there exists a deterministic polynomial time extractor $\mathcal{E}$ that on input $(\tau, \mathsf{state}_{\mathcal{R},\tau})$ outputs $\widetilde{M}$ such that the following properties hold.*

- **Frequency of Extraction.** *For every PPT committer $\mathcal{C}^*$,*

  $$\Pr[\mathcal{E}(\tau, \mathsf{state}_{\mathcal{R},\tau}) \neq \perp] = \epsilon$$

  where the probability is over $(\tau, \mathsf{state}_{\mathcal{C},\tau}, \mathsf{state}_{\mathcal{R},\tau}) \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$.

- **Correctness of Extraction.** *For every PPT committer $\mathcal{C}^*$, every $(\tau, \mathsf{state}_{\mathcal{C},\tau}, \mathsf{state}_{\mathcal{R},\tau}) \in \mathsf{Supp}(\mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle)$, and every y, denoting $\widetilde{M} = \mathcal{E}(\tau, \mathsf{state}_{\mathcal{R},\tau})$ and $M = \mathsf{Verify}(\tau, y)$, if $\widetilde{M} \neq \perp$ and $M \neq \perp$, then $\widetilde{M} = M$.*

[3] We note that in our definition, $\mathcal{R}$ *does not* need to use private state $\mathsf{state}_{\mathcal{R},\tau}$ from the commitment phase in order to execute the Verify algorithm in the decommitment phase.

- **Indistinguishability of Extractable Transcripts.** *For every $\mathcal{C}^*$ and every PPT adversary $\mathcal{A}$,*

$$\left|\Pr[\mathcal{C}^*(\tau) = 1 \mid \mathcal{E}(\tau, \mathsf{state}_{\mathcal{R},\tau}) \neq \bot] - \Pr[\mathcal{C}^*(\tau) = 1 \mid \mathcal{E}(\tau, \mathsf{state}_{\mathcal{R},\tau}) = \bot]\right| = \mathsf{negl}(\lambda)$$

where the probability is over $(\tau, \mathsf{state}_{\mathcal{R},\tau}) \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$.

We also consider a stronger definition, of ϵ-extractable statistically hiding ℓ multi-commitments, where we require that an entire sequence of ℓ commitments can be extracted with probability ϵ, that is independent of ℓ. We will also modify the Verify algorithm so that it obtains as input the transcript $\tau := (\tau_1, \tau_2, \ldots \tau_\ell)$ of all ℓ commitments, together with an index $i \in [\ell]$ and the decommitment $\mathsf{state}_{C,\tau,i}$ to a single commitment.

Definition 10 (ϵ-Extractable Statistically Hiding ℓ Multi-Commitments)**.** *A sequence of commitments $\langle \mathcal{C}, \mathcal{R}\rangle$ is a two-phase protocol between a committer $\mathcal{C}$ and receiver $\mathcal{R}$, consisting of algorithms $\mathsf{Commit}, \mathsf{Decommit}$ and Verify. At the beginning of the protocol, $\mathcal{C}$ obtains as input ℓ messages $(M_1, \ldots, M_\ell) \in \{0,1\}^{p\ell}$. Next, $\mathcal{C}$ and $\mathcal{R}$ execute the commit phase, and obtain a commitment transcript, denoted by $\tau := (\tau_1, \ldots \tau_\ell)$, together with private states for $\mathcal{C}$ and $\mathcal{R}$, denoted by $\{\mathsf{state}_{\mathcal{C},\tau,i}\}_{i\in[\ell]}$ and $\{\mathsf{state}_{\mathcal{R},\tau}\}_{i\in[\ell]}$ respectively. We use the notation*

$$(\tau, \{\mathsf{state}_{\mathcal{C},\tau,i}\}_{i\in[\ell]}, \{\mathsf{state}_{\mathcal{R},\tau,i}\}_{i\in[\ell]}) \leftarrow \mathsf{Commit}\langle \mathcal{C}(M), \mathcal{R}\rangle.$$

Later, $\mathcal{C}$ and $\mathcal{R}$ possibly engage in a decommit phase, where the committer $\mathcal{C}$ computes and sends message $y = \mathsf{Decommit}(\tau_i, \mathsf{state}_{\mathcal{C},\tau,i})$ to $\mathcal{R}$. At the end, $\mathcal{R}$ computes $\mathsf{Verify}(\tau, i, y_i)$, where $i \in [\ell]$, to output $\bot$ or a message $\widetilde{M}_i \in \{0,1\}^p$.[4] *A statistically hiding commitment scheme is required to satisfy three properties:*

- **(Perfect) Completeness.** *If $\mathcal{C}, \mathcal{R}$ honestly follow the protocol, then for every $M_1, \ldots, M_\ell \in \{0,1\}^{p\ell}$ and every $i \in [\ell]$:*

$$\Pr[\mathsf{Verify}(\tau, i, \mathsf{Decommit}(\tau_i, \mathsf{state}_{\mathcal{C},\tau,i})) = M_i] = 1$$

where the probability is over $(\tau, \{\mathsf{state}_{\mathcal{C},\tau,i}\}_{i\in[\ell]}) \leftarrow \mathsf{Commit}\langle \mathcal{C}(M), \mathcal{R}\rangle$.

- **Statistical Hiding.** *A set of commitments is $\delta(\lambda)$-statistically hiding if for every set $I \subseteq [\ell]$, every set of messages $\{M_i\}_{i\in[\ell]}, \{M'_i\}_{i\in[\ell]} \in \{0,1\}^{2p\ell}$ such that $(M_i = M'_i)$ for every $i \in I$, every unbounded malicious receiver $\mathcal{R}^*$ and honest committer $\mathcal{C}$, the statistical distance between the distributions $\mathsf{View}_{\mathcal{R}^*}(\mathsf{Commit}\langle \mathcal{C}(M_1, \ldots M_\ell), \mathcal{R}^*\rangle), \{\mathsf{Decommit}(\tau_i, \mathsf{state}_{C,\tau,i})\}_{i\in I}$ and $\mathsf{View}_{\mathcal{R}^*}(\mathsf{Commit}\langle \mathcal{C}(M'_1, \ldots, M'_\ell), \mathcal{R}^*\rangle), \{\mathsf{Decommit}(\tau_i, \mathsf{state}_{C,\tau,i})\}_{i\in I}$ is at most $(\ell - |I|) \cdot \delta(\lambda)$. The scheme is statistically hiding if $\delta(\lambda) \leq \frac{1}{\mathsf{poly}(\lambda)}$ for every polynomial $\mathsf{poly}(\cdot)$.*

- **Computational Binding.** *Consider any non-uniform PPT committer $\mathcal{C}^*$ that produces $\tau \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$, and then outputs y_1, y_2. For any $i \in [\ell]$, let $\widetilde{M}_1 = \mathsf{Verify}(\tau, i, y_1)$ and $\widetilde{M}_2 = \mathsf{Verify}(\tau, i, y_2)$. Then, we require that*

$$\Pr\left[(\widetilde{M}_1 \neq \bot) \ \wedge \ (\widetilde{M}_2 \neq \bot) \ \wedge \ (\widetilde{M}_1 \neq \widetilde{M}_2)\right] = \mathsf{negl}(\lambda),$$

over the randomness of sampling $\tau \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$.

[4] We point out that in this definition, $\mathcal{R}$ *does not* need to use private state $\mathsf{state}_{\mathcal{R},\tau}$ from the commitment phase in order to execute the Verify algorithm in the decommitment phase. We also point out that the verify algorithm can be modified so that the committer can open a subset of commitments together, instead of one at a time.

Definition 11 (ϵ-Extractable Sequence of ℓ Statistically Hiding Commitments)**.** *We say that a sequence of ℓ statistically hiding commitments is ϵ-extractable if the following holds:*
For every $i \in [\ell]$, Denote $\{(\tau_i, \mathsf{state}_{\mathcal{C},\tau,i}, \mathsf{state}_{\mathcal{R},\tau,i})\}_{i\in[\ell]} \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$ and let $\tau := (\tau_1, \tau_2, \ldots \tau_\ell)$ denote the transcript of all ℓ commitments. We require that there exists a deterministic polynomial time extractor $\mathcal{E}$ that on input $\{(\tau_i, \mathsf{state}_{\mathcal{R},\tau,i})\}_{i\in[\ell]}$ outputs $\{\widetilde{M}_i\}_{i\in[\ell]}$ such that the following properties hold.

- **Frequency of Extraction.** *Let $(\widetilde{M}_1, \ldots, \widetilde{M}_\ell) := \mathcal{E}(\{(\tau_i, \mathsf{state}_{\mathcal{R},\tau,i})\}_{i\in[\ell]})$, then*

$$\Pr[\exists i \text{ such that } \widetilde{M}_i = \bot] \leq (1 - \epsilon)$$

  where the probability is over τ and over the random coins of $\mathcal{R}$.

- **Correctness of Extraction.** *For every $\{(\tau_i, \mathsf{state}_{\mathcal{C},\tau,i}, \mathsf{state}_{\mathcal{R},\tau,i})\}_{i\in[\ell]} \in (\mathsf{Supp}(\mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle))^\ell$, denoting $\widetilde{M}_1, \ldots \widetilde{M}_\ell = \mathcal{E}(\{(\tau_i, \mathsf{state}_{\mathcal{R},\tau,i})\}_{i\in[\ell]})$, we require the following. For every $i \in [\ell]$ and every string y, denoting $M = \mathsf{Verify}(\tau, i, y)$, if $M \neq \bot$ and $\widetilde{M}_i \neq \bot$, then $\widetilde{M}_i = M$.*

- **Indistinguishability of Extractable Transcripts.** *For every PPT adversary $\mathcal{C}^*$,*

$$\left|\Pr[\mathcal{C}^*(\tau) = 1 \mid \exists i \text{ such that } \widetilde{M}_i = \bot] - \Pr[\mathcal{C}^*(\tau) = 1 \mid \nexists i \text{ such that } \widetilde{M}_i = \bot]\right| = \mathsf{negl}(\lambda)$$

  where $\widetilde{M}_1, \ldots, \widetilde{M}_\ell := \mathcal{E}(\tau, \mathsf{state}_{\mathcal{R},\tau})$ and where the probability is over $(\tau, \mathsf{state}_{\mathcal{R},\tau}) \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle$.

4.2 Protocol

In this section, we construct two-message statistically hiding, extractable commitments according to Definition 9 assuming the existence of two message oblivious transfer (OT). Our construction is described in Figure 1.

Primitives Used. Let $\mathsf{OT} = (\mathsf{OT}_1, \mathsf{OT}_2)$ denote a two-message string oblivious transfer protocol according to Definition 4, also satisfying Property 1. Let $\mathsf{OT}_1(b; r_1)$ denote the first message of the OT protocol with receiver input b and randomness r_1, and let $\mathsf{OT}_2(M_0, M_1; r_2)$ denote the second message of the OT protocol with sender input strings M_0, M_1 and randomness r_2.[5]

Observe that the protocol satisfies the property mentioned in the definition that the verify algorithm in the decommitment phase does not require the private randomness used by the receiver in the commit phase. Further, observe that if the oblivious transfer protocol satisfies Property 1, the receiver's message can alternately be generated by just sampling a uniformly random string. Thus, this would give an extractable commitment protocol where the receiver's algorithms are public coin.

We will now prove the following main theorem.

Theorem 1. *Set $\epsilon = 2^{-m}(1-2^{-\log\lambda \log^*\lambda})$. Assuming that the underlying OT protocol is $(\lambda^{\log^*\lambda}/\epsilon)$-secure against malicious senders, $(1 - \delta_{\mathsf{OT}})$ secure against malicious receivers according to Definition 4, and satisfies Property 1, the scheme in Figure 1 is a $(1 - 2^{-m} - \delta_{\mathsf{OT}})$ statistically hiding, ϵ-extractable commitment scheme according to Definition 9. Further, the receiver's algorithms are public coin.*

Recall that such two-message OT protocols with this additional property have been constructed based on the DDH assumption [NP01], LWE assumption [BD18], and a stronger variant of smooth-projective hashing, which can be realized from DDH as well as the N^{th}-residuosity and Quadratic

[5]Note that OT_2 also depends on OT_1. We omit this dependence in our notation for brevity.

Extraction parameter: m.
Committer Input: Message $M \in \{0,1\}^p$.

Commit Stage:
Receiver Message.

1. Pick challenge string $\mathsf{ch} \xleftarrow{\$} \{0,1\}^m$.
2. Sample uniform randomness $\{r_{1,i}\}_{i\in[m]}$.
3. Compute and send $\{\mathsf{OT}_1(\mathsf{ch}_i, r_{1,i})\}_{i\in[m]}$ using m instances of two-message OT.

Committer Message.

1. Sample a random string $r \xleftarrow{\$} \{0,1\}^m$.
   For every $i \in [m]$ and every $b \in \{0,1\}$, sample $M_i^b \xleftarrow{\$} \{0,1\}^p$ subject to $\bigoplus_{i\in[m]} M_i^{r_i} = M$.
2. For every $i \in [m]$ compute $o_{2,i} = \mathsf{OT}_2(M_i^0, M_i^1; r_{2,i})$ with uniform randomness $r_{2,i}$.
3. Send $(r, \{o_{2,i}\}_{i\in[m]})$.

Reveal Stage: The committer reveals M, and all values $\{M_i^0, M_i^1\}_{i\in[m]}$ as well as the randomness $r_{2,i}$. The receiver accepts the decommitment to message M if and only if:

1. For all $i \in [m]$, $o_{2,i} = \mathsf{OT}_2(M_i^0, M_i^1; r_{2,i})$,
2. $\bigoplus_{i\in[m]} M_i^{r_i} = M$.

Figure 1: Extractable Commitments

Residuosity assumptions [Kal05, HK12]. Further, we note that in all these OT constructions, computing the output of the protocol can be represented by an NC^1 circuit and we will use this fact later. Instantiating the OT protocol in the above construction, we get the following corollary:

Corollary 2. *Set $\epsilon = 2^{-m}(1-2^{-\log\lambda \log^*\lambda})$. Assuming quasi-polynomially secure LWE/DDH/QR/N^{th}-residuosity, the scheme in Figure 1 is a $(1-2^{-m}-\delta_{\mathsf{OT}})$ statistically hiding, ϵ-extractable commitment scheme according to Definition 9 where the receiver's algorithms are public coin.*

We now prove the above theorem by showing statistical hiding, computational binding, and extractability in Lemma 1, Lemma 2 and Lemma 3 below respectively.

Lemma 1. *Assuming the underlying OT satisfies $(1-\delta_{\mathsf{OT}})$ statistical sender security according to Definition 4 and satisfies Property 1, the scheme in Figure 1 is $(1-2^m-\delta_{\mathsf{OT}})$ statistically hiding according to Definition 8.*

Proof. Fix any (unbounded) malicious receiver $\mathcal{R}^*$. Let $m_{\mathcal{R}^*}$ be the message sent by $\mathcal{R}^*$ during the commit phase. By Property 1, $m_{\mathcal{R}^*}$ uniquely defines a receiver challenge ch. With probability 2^{-m}, $r \neq \mathsf{ch}$ for r chosen uniformly at random by an honest committer. Conditioned on $r \neq \mathsf{ch}$, there exists at least one index $j \in [m]$ such that $r_j \neq \mathsf{ch}_j$. When $r_j \neq \mathsf{ch}_j$, by $(1-\delta_{\mathsf{OT}})$ statistical sender security of OT, $M_j^{r_j}$ is $(1-\delta_{\mathsf{OT}})$-statistically hidden from any malicious receiver. Since $M_j^{r_j}$ is one of the shares in an XOR secret sharing of M, the message M is $(1-2^{-m}-\delta_{\mathsf{OT}})$ statistically hidden from any malicious receiver. □

Lemma 2. *Assuming that the underlying OT satisfies receiver security according to Definition 4, the scheme in Figure 1 is computationally binding against non-uniform PPT malicious committers, according to Definition 8.*

Proof. Suppose for contradiction that there exists a non-uniform malicious PPT cheating committer $\mathcal{C}^*$ and a polynomial $p(\cdot)$ such that outputs transcript $\tau \leftarrow \mathsf{Commit}\langle \mathcal{C}^*, \mathcal{R}\rangle, y_1, y_2$ such that $\widetilde{M}_1 = \mathsf{Verify}(\tau, y_1), \widetilde{M}_2 = \mathsf{Verify}(\tau, y_2)$, and

$$\mathsf{Pr}[(\widetilde{M}_1 \neq \bot) \ \wedge \ (\widetilde{M}_2 \neq \bot) \ \wedge \ (\widetilde{M}_1 \neq \widetilde{M}_2)] = \frac{1}{p(\lambda)}$$

We will construct a reduction $\mathcal{A}$ that has black-box access to such a committer $\mathcal{C}^*$, and breaks receiver OT security according to Definition 4. $\mathcal{A}$ takes as input $o' = \mathsf{OT}_1(b)$ and is required to distinguish the case when $b = 0$ from when $b = 1$. $\mathcal{A}$ does the following:

1. Sample $\ell \xleftarrow{\$} [m], \{\mathsf{ch}_i\}_{i\in[m]\setminus\{\ell\}} \xleftarrow{\$} \{0,1\}^{m-1}$.
2. Sample $r_{1,i}$ uniformly at random and compute $\{o_{1,i} = \mathsf{OT}_1(\mathsf{ch}_i; r_{1,i})\}_{i\in[m]\setminus\{\ell\}}$. Set $o_{1,\ell} = o'$.
3. Forward $\{o_{1,i}\}_{i\in[m]}$ as the first message of the scheme in Figure 1 to the adversary $\mathcal{C}^*$.
4. Obtain $r, \{o_{2,i}\}_{i\in[m]}$ from $\mathcal{C}^*$. This is the end of the commit phase, denote transcript by τ.
5. Obtain y_1, y_2 from $\mathcal{C}^*$. If $\mathsf{Verify}(\tau, \mathsf{y_1}) = \mathsf{Verify}(\tau, y_2)$ or if either of them are $\bot$, abort and output $\bot$. Else continue.
6. Parse y_1 as $M, \{M_i^b, r_{2,i}\}_{i\in[m],b\in\{0,1\}}$ and y_2 as $\widetilde{M}, \{\widetilde{M}_i^b, \widetilde{r}_{2,i}\}_{i\in[m],b\in\{0,1\}}$.
7. Define $\mathbb{S} = \{j \in [m] : M_j^{r_j} \neq \widetilde{M}_j^{r_j}\}$. Since $M \neq \widetilde{M}$, $|\mathbb{S}| > 1$. If $\ell \in \mathbb{S}$, output $1 - r_\ell$, else output $\bot$.

We prove the following claim, which will contradict OT security according to Definition 4 and complete the proof of the lemma.

Claim 2. *Let b' denote the output of $\mathcal{A}$. Then,* $\Pr[b' = b] \geq \frac{1}{m\cdot p(\lambda)}$.

Proof of Claim. By assumption, with probability at least $\frac{1}{p(\lambda)}$, the following event $\mathbb{E}$ occurs: $\mathcal{A}$ proceeds to Step 5 and obtains $M \neq \widetilde{M}$.

After $\mathcal{A}$ proceeds to Step 5, it first creates the set $\mathbb{S}$ of indices $j \in [m]$ where $M_j^{r_j} \neq \widetilde{M}_j^{r_j}$. Since $M \neq \widetilde{M}$, $|\mathbb{S}| > 1$. Since $\mathcal{A}$ samples ℓ independently and uniformly at random, $\mathsf{Pr}\left[\ell \in \mathbb{S} \middle| \mathbb{E}\right] \geq \frac{1}{m}$. Thus, $\mathsf{Pr}[(\ell \in \mathbb{S}) \ \wedge \ \mathbb{E}] \geq \frac{1}{m\cdot p(\lambda)}$.

Let us now condition on $(\ell \in \mathbb{S}) \wedge \mathbb{E}$. For any $i \in [m]$, by correctness of the i^{th} parallel OT, the i^{th} OT statistically binds the sender to a unique input value $M_i^{\mathsf{ch}_i}$. Thus, for any $i \in [m]$, the existence of $M_i^{r_i} \neq \widetilde{M}_i^{r_i}$ that reconstruct to $o_{2,i}$ implies that $\mathsf{ch}_i \neq r_i$ (except with probability $\mathsf{negl}(\lambda)$). Since we conditioned on $\ell \in \mathbb{S}$, we have that $M_\ell^{r_\ell} \neq \widetilde{M}_\ell^{r_\ell}$, and therefore $(b = \mathsf{ch}_\ell) = (1 - r_\ell) = b'$.

Thus, $\Pr\left[b' = b \middle| (\ell \in \mathbb{S}) \wedge \mathbb{E}\right] = 1$. This implies that $\Pr\left[b' = b \ \wedge \ (\ell \in \mathbb{S}) \wedge \mathbb{E}\right] \geq \frac{1}{m\cdot p(\lambda)}$. Since $\mathcal{A}$ outputs $\bot$ if either of the events $(\ell \in \mathbb{S})$ and $\mathbb{E}$ did not occur, we have that $\Pr[b' = b] \geq \frac{1}{m\cdot p(\lambda)}$, proving the claim. □

Lemma 3. *The scheme in Figure 1 is an ϵ-extractable commitment scheme, where $\epsilon = 2^{-m}(1 - 2^{-\log^* \lambda})$.*

Proof. We begin by describing the extractor $\mathcal{E}$ from Definition 9. We denote the first message of transcript τ by τ_1 and the second message by τ_2. $\mathcal{E}$ obtains transcript (τ_1, τ_2) together with $\mathsf{state}_{\mathcal{R},\tau}$, and does the following:

- Parse $\tau_2 = (r, \{o_{2,i}\}_{i\in[m]})$.
- If $r \neq ch$, output $\perp$.
- Else, use $\mathsf{state}_{\mathcal{R},\tau}$ to obtain $\{\widetilde{M}_i^{\mathsf{ch}_i}\}_{i\in[m]}$ from $\{o_{2,i}\}_{i\in[m]}$.
- Compute $\widetilde{M} = \oplus_{i\in[m]} \widetilde{M}_i^{\mathsf{ch}_i}$ and output $\widetilde{M}$.

We will now analyze the extractor $\mathcal{E}$. First, before describing the properties, we quickly observe that the extractor can be represented by an NC^1 circuit since computing the output of the OT protocol can be represented by an NC^1 circuit.

Frequency of Extraction. Recall that extraction succeeds from a trial when the r chosen by the committer is equal to the ch chosen by the extractor in this trial. We now prove the following claim, which asserts that the event $r = \mathsf{ch}$ occurs with probability at least ϵ.

Claim 3. *The probability (over the coins of $\mathcal{R}$) that $r = \mathsf{ch}$ is at least $2^{-m} \cdot (1 - 2^{-\log\lambda \log^*\lambda})$.*

Proof of Claim. Suppose the claim is not true. We will use this to contradict the receiver security of the OT protocol. To do so, we describe our reduction algorithm $\widehat{\mathcal{A}}$ and distinguisher $\widehat{\mathcal{D}}$ for the receiver security game.

The reduction $\widehat{\mathcal{A}}$ picks two challenges $\mathsf{ch}_1, \mathsf{ch}_2 \xleftarrow{\$} \{0,1\}^m$ at random, and creates auxiliary information consisting of these two challenges. We will use $\mathsf{OT}_1(\mathsf{ch}; R)$ to denote $\{\mathsf{OT}_1(\mathsf{ch}_i; R_i)\}_{i\in[m]}$, created using randomness R.

Now, $\widehat{\mathcal{D}}$ will obtain from the OT challenger as challenge either $\tau_1 = \mathsf{OT}_1(\mathsf{ch}_1; R)$ or $\tau_1 = \mathsf{OT}_1(\mathsf{ch}_2; R)$. $\hat{\mathcal{D}}$ runs the malicious sender $\mathcal{C}^*$ on the message τ_1, obtaining τ_2. It obtains r from τ_2: if $r = \mathsf{ch}_1$, it outputs 1. Otherwise, it aborts and outputs $\perp$.

We will now analyze the probability that $\hat{\mathcal{D}}$ outputs 1 in the two cases where $\mathsf{ch} = \mathsf{ch}_1$ or $\mathsf{ch} = \mathsf{ch}_2$. If $\mathsf{ch} = \mathsf{ch}_1$, then by assumption, we have that $\Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_1] < 2^{-m} \cdot (1 - 2^{-\log\lambda \log^*\lambda})$. On the other hand, if $\mathsf{ch} = \mathsf{ch}_2$, then no information about ch_1 is given to the distinguisher $\mathcal{D}$. Therefore, $\Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_2] = 2^{-m}$.

Thus, we have that $|\Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_1] - \Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_2]| \geq 2^{-m} \cdot 2^{-\log\lambda \log^*\lambda}$, which is a contradiction since the underlying OT is $\lambda^{\log^*\lambda}/\epsilon$ secure for $\epsilon < 2^{-m}$. □

Correctness of Extraction. The following claim proves that the extraction is "correct".

Claim 4. *Whenever the extraction algorithm $\mathcal{E}(\tau, \mathsf{state}_{\tau,\mathcal{R}})$ outputs $\widetilde{M} \neq \perp$, the following holds: The transcript (τ_1, τ_2) statistically binds $\mathcal{C}^*$ to either an invalid message, or a single message M, such that $\widetilde{M} = M$.*

Proof. First, when $r \neq \mathsf{ch}$, the extractor outputs $\perp$.

When $r = \mathsf{ch}$, the extractor obtains $\{M_i^{\mathsf{ch}_i}\}_{i\in[m]}$ as OT ouptut, and outputs $\widetilde{M} = \bigoplus_{i\in[m]} M_i^{\mathsf{ch}_i}$. By correctness of OT and since $r = \mathsf{ch}$, $\widetilde{M} = \bigoplus_{i\in[m]} M_i^{r_i}$, correctness of extracted value follows. Moreover, by correctness of OT, the committer is statistically bound to a single input for each index $i \in [m]$, and therefore to a unique message in all transcripts where $r = \mathsf{ch}$. □

Indistinguishability of Extractable Transcripts. Towards a contradiction, suppose there is a committer $\mathcal{C}^*$ and a polynomial $p(\cdot)$ such that:

$$\left|\Pr[\mathcal{C}^*(\tau) = 1 | \widetilde{M} = \perp] - \Pr[\mathcal{C}^*(\tau) = 1 | \widetilde{M} \neq \perp]\right| > \frac{1}{p(\lambda)}$$

We will use this committer to contradict the receiver security of the OT protocol. We now describe our reduction algorithm $\hat{\mathcal{A}}$ and distinguisher $\hat{\mathcal{D}}$ for the receiver security game.

The reduction $\hat{\mathcal{A}}$ chooses two challenges $\mathsf{ch}_1, \mathsf{ch}_2 \xleftarrow{\$} \{0,1\}^m$ at random, and creates auxiliary information consisting of these two challenges. Next, $\hat{\mathcal{A}}$ obtains as input (from the OT challenger) either $\tau_1 = \mathsf{OT}_1(\mathsf{ch} = \mathsf{ch}_1; R)$ or $\tau_1 = \mathsf{OT}_1(\mathsf{ch} = \mathsf{ch}_2; R)$.

$\mathcal{A}$ now runs the malicious sender $\mathcal{C}^*$ on the message τ_1, obtaining τ_2, aux. It generates the joint distribution $(\tau_1, \tau_2, \mathsf{aux})$. If $r \neq \mathsf{ch}_1$, it outputs $\mathcal{C}^*(\tau_1, \tau_2, \mathsf{aux})$. Otherwise, it aborts and outputs $\perp$.

Note that $\widetilde{M} \neq \perp \iff r = \mathsf{ch}$, and therefore, we have that

$$\left|\Pr[\mathcal{C}^*(\tau) = 1 | r \neq \mathsf{ch}] - \Pr[\mathcal{C}^*(\tau) = 1 | r = \mathsf{ch}]\right| > \frac{1}{p(\lambda)}$$

We now analyze two cases.

- Suppose $\mathsf{ch} = \mathsf{ch}_1$. Then, $r \neq \mathsf{ch} \implies r \neq \mathsf{ch}_1$, and $r = \mathsf{ch} \implies r = \mathsf{ch}_1$. In this case,

$$\left|\Pr[\mathcal{C}^*(\tau) = 1 | r \neq \mathsf{ch}_1, \mathsf{ch} = \mathsf{ch}_1] - \Pr[\mathcal{C}^*(\tau) = 1 | r = \mathsf{ch}_1, \mathsf{ch} = \mathsf{ch}_1]\right| > \frac{1}{p(\lambda)}$$

  which implies that

$$\begin{aligned}
&\Pr[\widehat{\mathcal{D}}(\tau) = 1 | \mathsf{ch} = \mathsf{ch}_1] = \\
&\Pr[\widehat{\mathcal{D}}(\tau) = 1 \wedge r \neq \mathsf{ch}_1 | \mathsf{ch} = \mathsf{ch}_1] > \\
&\Pr[\mathcal{C}^*(\tau) = 1 \wedge r \neq \mathsf{ch}_1 | \mathsf{ch} = \mathsf{ch}_1] = \\
&\Pr[\mathcal{C}^*(\tau) = 1 | r \neq \mathsf{ch}_1, \mathsf{ch} = \mathsf{ch}_1] \cdot \Pr[r \neq \mathsf{ch}_1 | \mathsf{ch} = \mathsf{ch}_1] > \\
&\frac{1}{p(\lambda)} \cdot (1 - 2^{-m} - \epsilon) > \frac{1}{2p(\lambda)}.
\end{aligned}$$

- Suppose $\mathsf{ch} = \mathsf{ch}_2$. In this case, no information about ch_1 is given to $\mathcal{C}^*$, and therefore

$$\Pr[\mathcal{D}^*(\tau) = 1 | \mathsf{ch} = \mathsf{ch}_2] \leq \Pr[\mathcal{C}^*(\tau) = 1 \ \wedge \ r \neq \mathsf{ch}_1 | \mathsf{ch} = \mathsf{ch}_2] = 2^{-m} \cdot 1 - 2^{-m}$$

Therefore,

$$|\Pr[\mathcal{D}^*(\tau) = 1 | \mathsf{ch} = \mathsf{ch}_1] - \Pr[\mathcal{D}^*(\tau) = 1 | \mathsf{ch} = \mathsf{ch}_2]| > \frac{1}{2p(\lambda)} - 2^{-m} > \frac{1}{3p(\lambda)}$$

This contradicts receiver security, and therefore the lemma follows. □

We will rely on a simple variant of this commitment scheme that for any $\ell = \mathsf{poly}(\lambda)$, results in an ϵ-extractable sequence of ℓ commitments where $\epsilon = 2^{-m}(1 - 2^{-\log \lambda \log^* \lambda})$ (independent of ℓ). This variant requires the committer to sample a *single* random string $r \xleftarrow{\$} \{0,1\}^m$ (refer to Step 1 of the committer message), and reuse the same string r across all ℓ commitments. Since the committer outputs r in the clear, a receiver obtaining such a sequence can efficiently check, during the verify phase of decommitment, whether all ℓ commitments use the same randomness r, and

output $\perp$ if this is not the case. This sequence of ℓ commitments is statistically hiding by a simple hybrid argument, and is computationally binding for the same reason as Lemma 2.

The extractor $\mathcal{E}$ is modified so that on input a transcript of ℓ commitments, it outputs 0^ℓ if the random string r is not identical across all commitments. Therefore, the extractor outputs $(\widetilde{M}_1, \ldots \widetilde{M}_\ell)$, where there exists an $i \in [\ell]$ such that $\widetilde{M}_i = \perp$ if and only if the committer used r identically across all commitments, and $\mathsf{ch} \neq r$. Therefore, frequency of extraction follows by Claim 3, correctness of extraction follows by Claim 4 together with the fact that when r is not identical across all commitments, the Verify algorithm outputs $\perp$. Finally, indistinguishability of extractable transcripts follows identically as in the case of a single commitment.

Setting $m = (\log \lambda \log^* \lambda)$ for this construction gives us the following corollary.

Corollary 3. *For any $\ell = \mathsf{poly}(\lambda)$, there exists an ϵ-extractable sequence of ℓ statistically hiding commitments, where $\epsilon = \lambda^{-\Theta(\log^* \lambda)}$, assuming quasi-polynomially secure LWE/DDH/QR/N^{th}-residuosity. Further, the extractor $\mathcal{E}$ can be represented by an NC^1 circuit.*

5 Our Statistical WI Protocol

5.1 Modified Blum Protocol

We begin by describing a very simple modification to the Blum Σ-protocol for Graph Hamiltonicity. The protocol we describe will have soundness error $\frac{1}{2} - \mathsf{negl}(\lambda)$ against adaptive PPT provers, and will satisfy *statistical* zero-knowledge. Since Graph Hamiltonicity is NP-complete, this protocol can also be used to prove any statement in NP via a Karp reduction. This protocol is described in Figure 2.

We give an overview of the protocol here. Note that the only modification to the original protocol of Blum [Blu86] is that we use two message statistically hiding, extractable commitments instead of non-interactive statistically binding commitments. The proofs of soundness and statistical honest-verifier zero-knowledge are fairly straightforward. They roughly follow the same structure as [Blu86], replacing statistically binding commitments with statistically hiding commitments.

Lemma 4. *Assuming that* extcom *is computationally binding, the protocol in Figure 2 satisfies soundness against PPT provers that may choose x adaptively in the second round of the protocol.*

Proof. The proof of soundness follows by the computational binding property of extcom and the soundness of the (original) Blum protocol.

Let L denote the language consisting of all graphs that have a Hamiltonian cycle. Consider a cheating prover P^* that convinces a malicious verifier about a statement $x \notin L$ with probability $\frac{1}{2} + h(n)$, where $h(\cdot) > \frac{1}{\mathsf{poly}(\cdot)}$ for some polynomial $\mathsf{poly}(\cdot)$. By an averaging argument, this means that there exists at least one transcript prefix τ consisting of the first two messages of the protocol, where for $G \notin L$ sent by the prover in the third message, $\Pr[V \text{ accepts} | \tau, G \notin L] > \frac{1}{2}$. This implies that there exists a cheating prover that generates a transcript prefix τ, for which it provides an accepting opening corresponding to both $b = 0$ and $b = 1$, with probability at least $h(n)$. Next, we argue that such a cheating prover must break the (computational) binding of com.

Since $G \notin L$, it is information theoretically impossible for any cheating prover to generate a commitment to a unique string $\pi, \pi(G)$ such that there exists a Hamiltonian cycle in $\pi(G)$. Therefore, any prover that opens a transcript prefix τ, G corresponding to both $b = 0$ and $b = 1$ for $G \notin L$, must open at least one commitment in the set $\{\mathsf{extcom}_P, \{\mathsf{extcom}_{i,j}\}_{i,j \in p \times p}\}$ to two different values, thereby giving a contradiction to the binding of the commitment scheme. $\square$

Modified Blum Argument

1. **Verifier Message:** The verifier does the following:
   - Sample the first message $\mathsf{extcom}_{1,i,j}$ for independent instances of the extractable commitment, where $i, j \in [p(\lambda)] \times [p(\lambda)]$, *uniformly at random.*
   - Send an additional first message $\mathsf{extcom}_{1,P}$ for another independent instance of the extractable commitment, again sampled *uniformly at random.*
2. **Prover Message:** The prover gets input graph $G \in \{0,1\}^{p(\lambda)\times p(\lambda)}$ represented as an adjacency matrix, with $(i,j)^{th}$ entry denoted by $G[i][j]$), Hamiltonian cycle $H \subseteq G$. Here $p(\cdot)$ is an a-priori fixed polynomial. The prover does the following:
   - Sample a random permutation π on $p(\lambda)$ nodes, and compute $c_P = \mathsf{extcom}_{2,P}(\pi)$ as a commitment to π using extcom.
   - Compute $\pi(G)$, which is the adjacency matrix corresponding to the graph G when its nodes are permuted according to π. Compute $c_{i,j} = \mathsf{extcom}_{2,i,j}(\pi(G)[i][j])$ for $(i,j) \in [p(\lambda)] \times [p(\lambda)]$.
   - Send $G, c_P, c_{i,j}$ for $(i,j) \in [p(\lambda)] \times [p(\lambda)]$.
3. **Verifier Message:** Sample and send $c \xleftarrow{\$} \{0,1\}$ to the prover.
4. **Prover Message:** The prover does the following:
   - If $c = 0$, send π and the decommitments of $\mathsf{extcom}_P, \mathsf{extcom}_{i,j}$ for $(i,j) \in [p(\lambda)] \times [p(\lambda)]$.
   - If $c = 1$, send the decommitment of $\mathsf{extcom}_{i,j}$ for all (i,j) such that $\pi(H)[i][j] = 1$.
5. **Verifier Output:** The verifier does the following:
   - If $c = 0$, accept if and only if all extcom openings were accepted and $\pi(G)$ was computed correctly by applying π on G.
   - If $c = 1$, accept if and only if all extcom openings were accepted and all the opened commitments form a Hamiltonian cycle.

Remark: Observe that since the receiver's algorithms in the extractable commitment scheme are public coin, the above protocol is also public coin.

Figure 2: Modified Blum SZK Argument

Lemma 5. *Assuming that* extcom *is statistically hiding, the protocol in Figure 2 satisfies honest-verifier statistical zero-knowledge.*

Proof. The simulation strategy is identical to that of [Blu86]. The simulator Sim first guesses the challenge bit c'. It begins an interaction with the malicious verifier. On obtaining the first message from the verifier, if $c' = 0$, it samples π uniformly at random and generates a commitment to $\pi, \pi(G)$ following honest prover strategy to generate the commitment. If $c' = 1$, it samples π, H' uniformly at random where H' is an arbitrary hamiltonian cycle, and generates a commitment to $\pi, \pi(H')$ following honest prover strategy to generate the commitment. Next, it waits for the

verifier to send c, and if $c \neq c'$, it aborts and repeats the experiment. If $c = c'$, then it decommits to the commitments according to honest prover strategy.

Note that when $c = c' = 1$, the resulting simulation is perfect zero-knowledge since the simulated view of the verifier is identical to the view generated by an honest prover. On the other hand when $c = c' = 0$, it follows from the statistical hiding property of the commitment extcom that the verifier cannot distinguish the case where extcom is a commitment to $\pi, \pi(G)$ and a hamiltonian cycle is opened in $\pi(G)$, from the case where extcom is not a commitment to $\pi(G)$, but instead to some $\pi(H')$ for a hamiltonian cycle H'. □

Since honest-verifier zero-knowledge composes under parallel repetition, we can repeat the protocol several times in parallel to get negligible soundness error. Formally, we have the following lemma:

Lemma 6. *Assuming that* extcom *is statistically hiding, the protocol in Figure 2 satisfies honest verifier statistical zero-knowledge under parallel repetition.*

Finally, Cramer et al. [CDS94] showed that honest verifier zero knowledge where the receiver's algorithms are public coin implies witness indistinguishability even against malicious verifiers. As a result, we get the following lemma:

Lemma 7. *Assuming that* extcom *is statistically hiding, the protocol in Figure 2 satisfies statistical witness indistinguishability under parallel repetition.*

5.2 Statistical ZAPs

In this section, we prove the following theorem:

Theorem 4. *There exists a two message public-coin statistical witness indistinguishable argument system for NP in the plain model assuming that the following primitives exist:*

- *Two-message oblivious transfer (OT) that is quasi-polynomially secure against malicious senders, satisfying Definition 4 and Property 1, and,*
- *Quasi-polynomially correlation intractable hash functions.*

Recall from previous sections that we can use the above OT to build the extractable commitment which is then used to build a four message Σ-protocol that is a modification to Blum's protocol. As mentioned before, we can instantiate both the OT and the correlation intractable hash function assuming the learning with errors (LWE) assumption. Therefore, instantiating both the primitives in the above theorem gives us the following:

Theorem 5. *Assuming quasi-polynomially secure LWE, there exists a two message public-coin statistical witness indistinguishable argument system for NP in the plain model.*

Notations and Primitives used.

- Let λ be the security parameter.
- Let $\Sigma := (\Sigma_1, \ldots, \Sigma_\lambda)$ denote λ parallel repetitions of the modified Blum Sigma protocol constructed in Section 5.1, where for $i \in [\ell], \Sigma_i = (q_i, a_i, e_i, z_i)$. Let the underlying commitment scheme be instantiated with extraction success probability $\epsilon = \lambda^{-\theta(\log^* \lambda)}$.

- Let $\mathcal{H}$ be a correlation intractable hash function with respect to $\{\mathcal{F}_{\lambda,s(\lambda)}\}_{\lambda\in\mathbb{N}}$ according to Definition 3 that outputs strings of length λ, where $s(\lambda) = 2s_1(\lambda)$ where s_1 is the size of the extractor $\mathcal{E}$ used in the commitment scheme and $\mathcal{F}$ denotes the class of all NC^1 circuits of size $s(\lambda)$ as defined in Definition 2. Recall the correlation-intractability advantage is assumed to be at most $\frac{1}{\lambda^{(\omega(\log^* \lambda))^2}}$.

Construction. Let x be any instance in $\{0,1\}^\lambda$ and let w be the corresponding witness for the statement $x \in L$.

1. **Verifier's message to the Prover:**
   - Sample $q := \{q_i\}_{i\in[\lambda]}$.
   - Sample $K \leftarrow \mathcal{H}.\mathsf{Setup}(1^\lambda, 0^\ell)$.
   - Output (q, K).
2. **Prover's message to the Verifier:**
   - Compute $\{a_i\}_{i\in[\lambda]}$ as a response to $\{q_i\}_{i\in[\lambda]}$.
   - Compute $e \leftarrow \mathcal{H}.\mathsf{Eval}(K, x, (q, a))$.
   - Compute $\{z_i\}_{i\in[\lambda]}$ with respect to the challenge string e.
   - Output (x, a, e, z).
3. **Verification:** The verifier does the following:
   - If $\mathcal{H}.\mathsf{Eval}(K, x, a) \neq e$, output reject.
   - Else if (x, q, a, e, z) does not verify according to the Σ protocol, output reject.
   - Else output accept.

Completeness. Completeness of the protocol can be easily observed from the correctness of the underlying primitives: the protocol Σ and the hash function H.

Public Coin. Recall from the statistical indistinguishability of hash keys property that an honest verifier can just sample a uniformly random string as the hash key K. This, along with the fact that the underlying protocol Σ is public coin results in the above protocol also being public coin.

Soundness. We now prove computational soundness of the protocol above. Towards a contradiction, fix any adversary $\mathcal{A}$ that breaks soundness of the protocol with probability $\frac{1}{p(\lambda)}$ for some polynomial $p(\cdot)$.

We consider a sequence of computationally indistinguishable hybrids where the first hybrid corresponds to the real soundness experiment.

- Hybrid_0 : This hybrid corresponds to the experiment where the challenger behaves identically to the verifier in the actual protocol.
- Hybrid_1: In this hybrid, instead of generating the verifier's first message as uniformly random string, the challenger Ch now computes the first message of the extractable commitment scheme used in the underlying protocol Σ as done in the protocol description in Figure 1. In particular, the underlying OT receiver messages are not sampled as uniformly random strings but instead are computed by running the OT receiver algorithm. As a result, Ch now has some internal state r_{state} as part of the extractable commitment scheme that is not public.

- Hybrid_2: This hybrid is the same as the previous hybrid except that the hash key K is generated as follows. $K \leftarrow \mathcal{H}.\mathsf{Setup}(1^\lambda, R)$ where the relation R^* consists of tuples of the form $((x, q, a), y)$ where y is computed by an efficient function f_{bad} described below. f_{bad} takes as input the statement x, the verifier's secret state r_{state}, the prover's message a and does the following.

  1. Run the extractor algorithm $\mathcal{E}$ on input $(r_{\mathsf{state}}, \tau = (q, a))$ to compute m. Note that $\mathcal{E}$ can be represented by an NC^1 circuit of size $s_1(\lambda)$ for some polynomial s_1.
  2. If $m \neq \perp$, this means that m is the tuple of messages committed to in the set of λ commitment tuples $(c_P, \{c_{i,j}\})$. For each $k \in [\lambda]$, check whether the message committed to by the tuple $\{c_{i,j}\}$ is indeed equal to $\pi(G)$ where π is the permutation committed to in c_P. If so, then set $e_k = 0$ and else set $e_k = 1$. Set $y = (e_1, \ldots, e_\lambda)$.[6]
  3. If $m = \perp$, set $y = 0^\lambda$.

Before proving the indistinguishability of hybrids, we define an event that helps us in the proof.

Event E: Let τ denote the transcript of an execution of the above protocol and let $\tau_{\mathcal{C}}$ denote the transcript of the commitment scheme in the execution. Let $\mathsf{state}_{\mathcal{R}}$ denote the state of the verifier when it runs the receiver algorithm of the commitment scheme. We will say that the event **E** occurs if for any honest verifier V:

$$[V(\tau) = 1 \wedge \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp].$$

We now continue the proof of soundness with the following claims.

Lemma 8. *Assuming the pseudorandomness of receiver messages of the OT protocol used in the underlying extractable commitment scheme (Property 1),*

$$|\Pr[V(\tau) = 1|\mathsf{Hybrid}_1] - \Pr[V(\tau) = 1|\mathsf{Hybrid}_0]| = \mathsf{negl}(\lambda)$$

Proof. The only difference between the two hybrids is that in Hybrid_0, the OT receiver messages in the extractable commitment scheme used in the underlying protocol Σ are generated as uniformly random strings while in Hybrid_1, they are generated by running the algorithm OT_1 on behalf of the OT receiver. It is easy to see that if the difference in the adversary's success probability in breaking soundness between these two hybrids is non-negligible, we can break the pseudorandomness of receiver messages property (Property 1) of the underlying two message OT protocol, which is a contradiction. □

Lemma 9. *Assuming the frequency of extraction property and the indistinguishability of extractable transcripts property of the extractable commitment scheme, there exists a polynomial $p(\cdot)$ such that*

$$\Pr[\mathbf{E} \textit{ occurs in } \mathsf{Hybrid}_1] \geq \epsilon \cdot \frac{1}{p(\lambda)},$$

where the probability is over the randomness of V, and where $\epsilon = \lambda^{-\theta(\log^* \lambda)}$ is the extraction probability of the underlying commitment scheme.

[6] Essentially, since $x \notin L$, if the cheating prover has to succeed, it can either generate a successful response z_k for verifier's query bit $e_k = 0$ or $e_k = 1$ and this function determines which bit it is.

Proof. Fix $x \notin L$. We will consider a reduction $\mathcal{B}$ that interacts with the adversary and relies on the frequency of extraction property and the indistinguishability of extractable transcripts property of the extractable commitment scheme to prove the lemma.

$\mathcal{B}$ interacts with a challenger Ch for the commitment scheme and receives a first round message com_1 for the ℓ-extractable commitment scheme. It then interacts with the adversary $\mathcal{A}$ as the verifier in the ZAP protocol, setting com_1 as its message on behalf of the receiver in the underlying commitment scheme, and sampling the hash key $K \leftarrow \mathcal{H}.\mathsf{Setup}(1^\lambda, 0^\ell)$. After completing the protocol execution with $\mathcal{A}$, $\mathcal{B}$ forwards the commitments sent by $\mathcal{A}$ as its message com_2 of the commitment scheme to the challenger Ch. Further, $\mathcal{B}$ outputs 1 in its interaction with Ch if the proof provided by $\mathcal{A}$ verifies, and 0 otherwise.

Let τ denote the transcript of the ZAP protocol and $\tau_{\mathcal{C}}$ the transcript of the underlying commitment scheme. Let state_r be the state of the receiver in the commitment scheme as sampled by the challenger Ch.

First, we observe that by Lemma 8, there exists a polynomial $p(\cdot)$ such that adversary $\mathcal{A}$ breaks the soundness property in Hybrid_1 with non-negligible probability $\frac{1}{p(\lambda)}$. This implies that $\Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1] \geq \frac{1}{p(\lambda)}$ over the random coins of $\mathcal{B}, \mathsf{Ch}$. This gives us the following equation.

$$\begin{aligned}\Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1] = (&\Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1 \mid \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] \cdot \Pr[\mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] \\ &+ \Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1 \mid \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) = \perp] \cdot \Pr[\mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) = \perp]) \geq \frac{1}{p(\lambda)}\end{aligned} \tag{1}$$

From the indistinguishability of extractable transcripts property, we have that:

$$\left|\Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1 \mid \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] - \Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1 \mid \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) = \perp]\right| = \mathsf{negl}(\lambda) \tag{2}$$

From the frequency of extraction property, we have that :

$$\Pr[\mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] \geq \epsilon \tag{3}$$

where all equations are over the random coins of the challenger Ch and reduction $\mathcal{B}$. Combining Equations (1) and (2) implies that there exists a polynomial $q(\cdot)$ such that

$$\Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1 \mid \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] \geq \frac{1}{q(\lambda)},$$

which, by Equation (3), implies that

$$\Pr[\mathcal{B}(\tau) = 1 \wedge \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] = \Pr[\mathcal{B}(\tau_{\mathcal{C}}) = 1 \mid \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] \cdot \Pr[\mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] \geq \frac{1}{q(\lambda)} \cdot \epsilon.$$

Thus we have

$$\Pr[\mathbb{E} \text{ occurs in } \mathsf{Hybrid}_1] \geq \epsilon \cdot \frac{1}{q(\lambda)}.$$

This completes the proof of the Lemma. □

Lemma 10. *Assuming the statistical indistinguishability of hash keys of the correlation intractable hash function, there exists a polynomial $p(\cdot)$ such that*

$$\Pr[\mathbf{E} \textit{ occurs in } \mathsf{Hybrid}_2] \geq \epsilon \cdot \frac{1}{p(\lambda)},$$

where the probability is over the randomness of V, and where $\epsilon = \lambda^{-\Theta(\log^* \lambda)}$ is the extraction probability of the underlying commitment.

Proof. Assume for the sake of contradiction that the lemma is not true. We will show that we can break the statistical indistinguishability of hash keys property of the correlation intractable hash function.

We will design a reduction $\mathcal{B}$ that interacts with $\mathcal{A}$, where $\mathcal{B}$ acts as verifier in the above ZAP protocol. $\mathcal{B}$ interacts with a challenger Ch for the correlation intractable hash function. Initially, $\mathcal{B}$ samples the first round message q for the underlying Sigma protocol just as in Hybrid_1, along with associated receiver state $\mathsf{state}_{\mathcal{R}}$ for the commitment scheme, and sends both to Ch. $\mathcal{B}$ obtains a hash key K sampled either uniformly at random (as in Hybrid_1) or by running the setup algorithm of the hash function as described in Hybrid_2. $\mathcal{B}$ uses this key K in its interaction with the adversary $\mathcal{A}$ and completes executing the ZAP protocol. Observe that if Ch sampled a hash key uniformly at random, the interaction between $\mathcal{A}$ and $\mathcal{B}$ is identical to Hybrid_1 and if Ch sampled as hash key as described in Hybrid_2, the interaction between $\mathcal{A}$ and $\mathcal{B}$ is identical to Hybrid_2.

Now, $\mathcal{B}$ tests if event $\mathbf{E}$ occurs. That is, it checks if the ZAP protocol verifies and if so, runs the extractor $\mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}})$ using the transcript $\tau_{\mathcal{C}}$ for the commitment scheme. If the extractor cE does not output $\perp$, then event $\mathbf{E}$ occurs and $\mathcal{B}$ guesses that the hash key was uniformly sampled in its interaction with the challenger Ch. Otherwise, it guesses that the hash key was not uniformly sampled. Thus, if the event $\mathbf{E}$ occurs with probability $\geq \epsilon \cdot \frac{1}{p(\lambda)}$ in Hybrid_1, and occurs with probability $\epsilon \cdot \mathsf{negl}(\lambda)$ in Hybrid_2, $\mathcal{B}$ can distinguish between the hash keys with advantage $\frac{\epsilon}{q(\lambda)}$ for some polynomial q. This is a contradiction, and this completes the proof of the lemma. □

Lemma 11. *Assuming the quasi-polynomial correlation intractable property of the hash function, the soundness of the underlying protocol Σ and the correctness of extraction of the extractable commitment scheme,*

$$\Pr[\mathbb{E} \textit{ occurs in } \mathsf{Hybrid}_2] \leq \epsilon \cdot \mathsf{negl}(\lambda).$$

Proof. Suppose the claim is not true. This implies that

$$\Pr[V(\tau) = 1 \wedge \mathcal{E}(\tau_{\mathcal{C}}, \mathsf{state}_{\mathcal{R}}) \neq \perp] = \epsilon \cdot \frac{1}{p(\lambda)}$$

for some polynomial p. Let us consider any transcript on which event $\mathbf{E}$ occurs. Let (q, K) denote the verifier's message and (x, a, e, z) denote the prover's message. Then, from the correctness of the ZAP protocol, it must be the case that (q, a, e, z) verifies according to protocol Σ and $e = H(K, q, x, a)$. Further, since the extractor $\mathcal{E}$ succeeds on this transcript, the commitment scheme is statistically binding. Therefore, we can invoke the special soundness of the underlying modified Blum Σ protocol (as in the case of the regular Blum protocol) to state that for the statement $x \notin L$ and prefix (q, a) there can exist at most one pair (e^*, z^*) such that (q, a, e^*, z^*) verifies successfully. Therefore, the adversary's message e must be equal to this value e^*.

Now, from the description of the relation R used in defining the hash key K in Hybrid_2, we observe that, by the correctness of extraction, $f_{bad}(q, x, a) = e^* = H(K, q, x, a)$. Thus, for any transcript that satisfies the conditions in event $\mathbf{E}$, $f_{bad}(q, x, a) = e^* = H(K, q, x, a)$.

Thus, we can build a reduction $\mathcal{B}$ that, using the adversary $\mathcal{A}$, produces (x, q, a) such that $f_{bad}(q, x, a) = e^* = H(K, q, x, a)$ with probability at least $\epsilon \cdot \frac{1}{p(\lambda)} = \frac{1}{\lambda^{\Theta(\log^* \lambda)} \cdot p(\lambda)}$. Since by Definition 3 the advantage of any polynomial-time adversary in this game must be at most $\frac{1}{\lambda^{(\omega(\log^* \lambda))^2}}$, this yields a contradiction. □

Note that Lemma 10 and Lemma 11 contradict each other, and therefore the adversary does not break soundness in the real experiment. This completes the proof of soundness. □

Statistical Witness Indistinguishability. Let $\mathcal{A}$ denote the unbounded time adversarial verifier and Ch denote the challenger. Let x be the challenge instance of length λ and w_0 and w_1 be a pair of witnesses for $x \in L$. Consider a pair of hybrids where the first hybrid Hybrid_0 corresponds to Ch running the honest prover algorithm with witness w_0 being used and the second hybrid Hybrid_1 corresponds to Ch running the honest prover algorithm with witness w_1 being used. We now show that these two hybrids are statistically indistinguishable to complete the proof.

Claim 5. *Assuming the Σ-protocol is statistical witness indistinguishable, Hybrid_0 is statistically indistinguishable from Hybrid_1.*

Proof. We now show that if there exists an unbounded time adversary $\mathcal{A}$ for which the two hybrids are not statistically indistinguishable, we can build a reduction $\mathcal{B}$ that can break the witness indistinguishability of the underlying modified Blum's Sigma protocol which is a contradiction to Lemma 7. $\mathcal{B}$ acts as the challenger in its interaction with the adversary $\mathcal{A}$ that is trying to distinguish between these two hybrids. Further, $\mathcal{B}$ acts as the adversary in its interaction with a challenger $\mathcal{C}$ in trying to break the WI property of the modified Blum Sigma protocol. Initially, $\mathcal{A}$ sends a statement x, a pair of witnesses (w_0, w_1) and a first round message (q, K) for the above ZAP construction. $\mathcal{B}$ forwards (x, w_0, w_1) to the challenger $\mathcal{C}$ and sends q as its first message of the underlying protocol Σ. $\mathcal{C}$ responds with its round two message a on behalf of the prover. $\mathcal{B}$ computes $e \leftarrow \mathcal{H}.\mathsf{Eval}(K, x, (q, a))$ and sends it to $\mathcal{C}$. Finally, $\mathcal{C}$ responds with the last round message z on behalf of the prover. Now, $\mathcal{B}$ sends the tuple (x, a, e, z) to $\mathcal{A}$ as the prover message for the above ZAP protocol. Observe that if the challenger $\mathcal{C}$ interacted using witness w_0, then the interaction between the reduction $\mathcal{B}$ and the adversary $\mathcal{A}$ is identical to Hybrid_0 and if the challenger $\mathcal{C}$ interacted using witness w_1, then the interaction between the reduction $\mathcal{B}$ and the adversary $\mathcal{A}$ is identical to Hybrid_1. Thus, if these two hybrids are not statistically indistinguishable to $\mathcal{A}$, $\mathcal{B}$ can use the same guess used by $\mathcal{A}$ to distinguish them, to break the statistical witness indistinguishability property of the protocol Σ which is a contradiction. □

References

[BCR86] Gilles Brassard, Claude Crépeau, and Jean-Marc Robert. Information theoretic reductions among disclosure problems. In *27th Annual Symposium on Foundations of Computer Science, Toronto, Canada, 27-29 October 1986*, pages 168–173, 1986.

[BD18] Zvika Brakerski and Nico Döttling. Two-message statistically sender-private OT from LWE. In Amos Beimel and Stefan Dziembowski, editors, *Theory of Cryptography - 16th International Conference, TCC 2018, Panaji, India, November 11-14, 2018, Proceedings, Part II*, volume 11240 of *Lecture Notes in Computer Science*, pages 370–390. Springer, 2018.

[BFM88] Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications (extended abstract). In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA*, pages 103–112, 1988.

[BGI+17] Saikrishna Badrinarayanan, Sanjam Garg, Yuval Ishai, Amit Sahai, and Akshay Wadia. Two-message witness indistinguishability and secure computation in the plain model from new assumptions. In *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part III*, pages 275–303, 2017.

[Blu86] Manuel Blum. How to prove a theorem so no one else can claim it. In *Proceedings of the International Congress of Mathematicians, Berkeley, CA*, pages 1444–1451, 1986.

[BP15] Nir Bitansky and Omer Paneth. Zaps and non-interactive witness indistinguishability from indistinguishability obfuscation. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *Theory of Cryptography - 12th Theory of Cryptography Conference, TCC 2015, Warsaw, Poland, March 23-25, 2015, Proceedings, Part II*, volume 9015 of *Lecture Notes in Computer Science*, pages 401–427. Springer, 2015.

[CCH+19] Ran Canetti, Yilei Chen, Justin Holmgren, Alex Lombardi, Guy N. Rothblum, Ron D. Rothblum, and Daniel Wichs. Fiat-shamir: from practice to theory. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019.*, pages 1082–1090, 2019.

[CDS94] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In Yvo Desmedt, editor, *Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1994, Proceedings*, volume 839 of *Lecture Notes in Computer Science*, pages 174–187. Springer, 1994.

[Cha86] David Chaum. Demonstrating that a public predicate can be satisfied without revealing any information about how. In *Advances in Cryptology - CRYPTO '86, Santa Barbara, California, USA, 1986, Proceedings*, pages 195–199, 1986.

[DN00] Cynthia Dwork and Moni Naor. Zaps and their applications. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA*, pages 283–293. IEEE Computer Society, 2000.

[DN02] Cynthia Dwork and Moni Naor. Zaps and their applications. *Electronic Colloquium on Computational Complexity (ECCC)*, (001), 2002.

[DN07] Cynthia Dwork and Moni Naor. Zaps and their applications. *SIAM J. Comput.*, 36(6):1513–1543, 2007.

[FS90] Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In Harriet Ortiz, editor, *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, pages 416–426. ACM, 1990.

[GMW86] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity and a methodology of cryptographic protocol design (extended abstract). In *27th Annual Symposium on Foundations of Computer Science, Toronto, Canada, 27-29 October 1986*, pages 174–187, 1986.

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Non-interactive zaps and new techniques for NIZK. In Cynthia Dwork, editor, *Advances in Cryptology - CRYPTO 2006, 26th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2006, Proceedings*, volume 4117 of *Lecture Notes in Computer Science*, pages 97–111. Springer, 2006.

[HK12] Shai Halevi and Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. *J. Cryptology*, 25(1):158–193, 2012.

[JKKR17] Abhishek Jain, Yael Tauman Kalai, Dakshita Khurana, and Ron Rothblum. Distinguisher-dependent simulation in two rounds and its applications. In *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II*, pages 158–189, 2017.

[Kal05] Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 78–95. Springer, 2005.

[KKS18] Yael Tauman Kalai, Dakshita Khurana, and Amit Sahai. Statistical witness indistinguishability (and more) in two messages. In *Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part III*, pages 34–65, 2018.

[KS17] Dakshita Khurana and Amit Sahai. Two-message non-malleable commitments from standard sub-exponential assumptions. *IACR Cryptology ePrint Archive*, 2017:291, 2017.

[NP01] Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In S. Rao Kosaraju, editor, *Proceedings of the Twelfth Annual Symposium on Discrete Algorithms, January 7-9, 2001, Washington, DC, USA.*, pages 448–457. ACM/SIAM, 2001.

[Pas03] Rafael Pass. Simulation in quasi-polynomial time, and its application to protocol composition. In Eli Biham, editor, *Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings*, volume 2656 of *Lecture Notes in Computer Science*, pages 160–176. Springer, 2003.

[PS19] Chris Peikert and Sina Shiehian. Noninteractive zero knowledge for NP from (plain) learning with errors. *IACR Cryptology ePrint Archive*, 2019:158, 2019.

[SW14] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 475–484. ACM, 2014.