# Statistical Zero-Knowledge Arguments for **NP** from Any One-Way Function[*]

Minh-Huyen Nguyen Shien Jin Ong Salil Vadhan

Division of Engineering and Applied Sciences
Harvard University
Cambridge, Massachusetts, USA.
E-mail: {mnguyen,shienjin,salil}@eecs.harvard.edu

June 19, 2006

**Abstract**

We show that every language in **NP** has a *statistical* zero-knowledge argument system under the (minimal) complexity assumption that one-way functions exist. In such protocols, even a computationally unbounded verifier cannot learn anything other than the fact that the assertion being proven is true, whereas a polynomial-time prover cannot convince the verifier to accept a false assertion except with negligible probability. This resolves an open question posed by Naor, Ostrovsky, Venkatesan, and Yung (CRYPTO '92, J. Cryptology '98).

Departing from previous works on this problem, we do not construct standard statistically hiding commitments from any one-way function. Instead, we construct a relaxed variant of commitment schemes called "1-out-of-2-binding commitments," recently introduced by Nguyen and Vadhan (STOC '06).

**Keywords:** cryptography, one-way functions, zero-knowledge arguments, statistically hiding and computationally binding commitments.

[*] All three authors were supported by NSF grant CNS-0430336 and ONR grant N00014-04-1-0478.

# 1 Introduction

As first discovered by Shannon [Sha49] for the case of encryption, most interesting cryptographic tasks are impossible to achieve with absolute, information-theoretic security. Thus, modern cryptography aims to design protocols that are computationally intractable to break. Specifically, following Diffie and Hellman [DH76], this is typically done by showing that breaking the protocol is as hard as some intractable problem from complexity theory. Unfortunately, proving lower bounds of the sort needed seems beyond the reach of current techniques in complexity theory, and indeed would require at least proving $\mathbf{P} \neq \mathbf{NP}$.

Given this state of affairs, research in the foundations of cryptography has aimed to design cryptographic protocols based on complexity assumptions that are as weak and general as possible. This project was enormously successful in the 1980's. In a beautiful sequence of works, it was shown that many cryptographic primitives, such as pseudorandom generators, pseudorandom functions, private-key encryption and authentication, digital signatures, (computationally hiding) bit commitment, and (computational) zero-knowledge proofs could be constructed from any one-way function [HILL99, GGM86, Rom90, Nao91, GMW91], and moreover this complexity assumption is minimal in the sense that each of these primitives (and indeed almost any cryptographic task) implies the existence of one-way functions [IL89, OW93]. Moreover, it was shown that many of the remaining primitives, such as public-key encryption, collision-resistant hashing, and oblivious transfer, could not be reduced to the existence of one-way functions in a "black-box" manner [IR89, Sim98].

However, a few important primitives have resisted classification into the above categories. That is, it is only known how to build these primitives from seemingly stronger assumptions than the existence of one-way functions, yet there is no black-box separation between these primitives and one-way functions. In this work, we are interested in an example involving zero-knowledge protocols.

## 1.1 The Complexity of Zero Knowledge

*Zero-knowledge proofs* are protocols whereby one party, the *prover*, convinces another party, the *verifier*, that some assertion is true with the remarkable property that the verifier "learns nothing" other than the fact that the assertion being proven is true. Since their introduction by Goldwasser, Micali, and Rackoff [GMR89], zero-knowledge proofs have played a central role in the design and study of cryptographic protocols. Part of the reason for their vast applicability is the fact that, under certain complexity assumptions (discussed below), every language $L$ in $\mathbf{NP}$ has a zero-knowledge proof system [GMW91]. That is, a prover can efficiently convince a verifier that $x \in L$ in a zero-knowledge manner, provided that the prover possesses an $\mathbf{NP}$ witness to the membership of $x$ in $L$. This means that when designing cryptographic protocols, any time that one party needs to convince others of some fact (e.g., that it has followed the specified protocol) without revealing additional knowledge, it can do so provided that it possesses a witness to the fact (e.g., its own secret keys and coin tosses).

Zero-knowledge protocols come in several flavors, depending on how one formulates the two security conditions: (1) the zero-knowledge condition, which says that the verifier "learns nothing" other than the fact the assertion being proven is true, and (2) the soundness conditions, which says that the prover cannot convince the verifier of a false assertion. In *statistical zero knowledge*, the zero-knowledge condition holds regardless of the computational resources the verifier invests into trying to learn something from the interaction. In *computational zero knowledge*, we only require that a probabilistic polynomial-time verifier learn nothing from the interaction.[1] Similarly, for soundness, we have *statistical soundness*, a.k.a. *proof systems*, where even a computationally unbounded prover cannot convince the verifier of a false statement (except with negligible probability), and *computational soundness*, a.k.a. *argument systems* [BCC88], where we only require that a polynomial-time prover cannot convince the verifier of a false statement.

Of course, it would be ideal to have both security conditions be statistical, and thus hold against computationally unbounded adversaries. Unfortunately, the resulting notion, *statistical zero-knowledge proofs*, while quite interesting and nontrivial (cf., [Vad99]), can only be achieved for languages in $\mathbf{AM} \cap \mathbf{coAM}$ [For89, AH91]. Because $\mathbf{AM} \cap \mathbf{coAM}$ is not believed to contain $\mathbf{NP}$ (cf., [BHZ87]), it is unlikely that every problem in $\mathbf{NP}$ possess statistical zero-knowledge proofs. Thus at best, we can have one of the security conditions be statistical.

Computational zero-knowledge *proofs* (with statistical soundness) was the original notion proposed in [GMR89]. Goldreich, Micali, and Wigderson [GMW91] showed that we can construct such proof systems for all of $\mathbf{NP}$ from any bit-commitment scheme that is computationally hiding and statistically binding. By [Nao91, HILL99], such commitment schemes can be constructed from any one-way function, and thus we obtain computational zero-knowledge proofs for $\mathbf{NP}$ from any one-way function. This complexity assumption is essentially minimal due to results of Ostrovsky and Wigderson [OW93], who showed that zero-knowledge proofs for any non-trivial language imply a weak form of one-way functions.

Brassard, Chaum, and Crepeau [BCC88] proposed instead the notion of *statistical zero-knowledge arguments*,[2] between the which is what we study in this paper. One reason that this variant of zero-knowledge proofs may be preferable to the original one is that breaking the soundness property must be done "on-line" during the interaction with the verifier and thus we need only protect against the adversary's present-day computational resources, whereas breaking the zero-knowledge property can involve the adversary investing effort long after the interaction to try and learn something from the transcript of the interaction. Thus, it seems preferable for the zero-knowledge property to be the one with the stronger, statistical guarantee.

It is evident from the constructions of [GMW91, BCC88] that to construct statistical zero-knowledge arguments for all of $\mathbf{NP}$, it suffices to construct bit-commitment schemes that are statistically hiding and computationally binding. The early constructions of such schemes were based on specific number-theoretic complexity assumptions [BCC88, BKK90], and were later generalized to any family of claw-free permutations [GK96], and then to any family of collision-resistant hash functions [NY89] (see also [DPP98]).[3]

In 1992, Naor, Ostrovsky, Venkatesan, and Yung [NOVY98] showed that the collision resistance criterion[4] is not necessary, by giving a beautiful construction of statistically hiding commitments

[1] More precisely, in statistical zero knowledge, we require that the verifier's view of the interaction can be efficiently simulated up to negligible statistical distance, whereas in computational zero knowledge, we only require that the simulation be computationally indistinguishable from the verifier's view.

[2] Actually, [BCC88] and some subsequent works (such as [NOVY98]) constructed *perfect* zero-knowledge arguments, which intuitively guarantee that the verifier learns something from the interaction with *zero* probability (as opposed to negligible probability, as in statistical zero knowledge). However, this distinction is minor in comparison to the distinction between the statistical and computational zero knowledge, which refer to computationally unbounded and polynomial-time verifier strategies, respectively.

[3] The fact that claw-free permutations imply collision-resistant hash functions was shown in [GMR88, Dam87], and the early constructions of claw-free permutations based on specific number-theoretic complexity assumptions were given by [GMR88, BKK90].

[4] We note that one-way permutations and collision-resistant hashing are known to be incomparable under "black-

(in fact perfectly hiding ones) and thus statistical zero-knowledge arguments for **NP** from any one-way *permutation*. They left as an open question whether these primitives could be based on arbitrary one-way functions, which would again be essentially minimal by [Ost91, OW93].[5]

The only progress in the past decade came in 2005 when Haitner et al. [HHK$^+$05] showed how to construct statistically hiding commitments from any "approximable preimage size" one-way function, that is a one-way function where we can efficiently approximate the preimage size of points in the range.

Motivated by this recent development, in this paper we resolve the complexity of statistical zero-knowledge arguments for **NP**:

**Theorem 1.1.** *If one-way functions*[6] *exist, then every language in* **NP** *has a statistical zero-knowledge argument system.*

Deviating from prior works on this problem, we do *not* prove this theorem by constructing the standard notion of statistically hiding commitments from any one-way function. Instead, as described below, we work with a relaxed variant of commitment schemes recently introduced by Nguyen and Vadhan [NV06], which we describe in the next section.

We also remark that our protocol has a polynomial number of rounds, while a constant number of rounds can be achieved based on collision-resistant hashing [NY89, BCY91]. However, achieving a subpolynomial ($n^{o(1)}$) number of rounds is open even assuming the existence of one-way permutations (cf., [NOVY98]).

## 1.2 Techniques

We begin by recalling the notion of a *commitment scheme*. A commitment scheme is a two-stage protocol between a sender and a receiver. In the first stage, the sender 'commits' to a value $v$, and in the second, the sender 'reveals' this value to the receiver. We want two security properties from a commitment scheme. The *hiding* property says that the receiver does not learn anything about the value $v$ during the commit stage. The *binding* property says that after the commit stage, there is at most one value that the sender can successfully open (without the receiver rejecting). As with zero-knowledge protocols, each of these security properties can be computational or statistical. For commitments, it is impossible to have both properties be statistical. As mentioned earlier, statistically binding commitments can be constructed from any one-way function [Nao91, HILL99], but our interest is in statistically hiding commitments.

Recently, Nguyen and Vadhan [NV06] introduced a new relaxation of commitment schemes, called *1-out-of-2-binding commitment schemes*, symbolically written as $\binom{2}{1}$-binding commitment schemes. These are commitment schemes with two phases, each consisting of a commit stage and a reveal stage. In the first phase, the sender commits to and reveals one value $v_1$, and subsequently, in the second phase, the sender commits to and reveals a second value $v_2$. We require that both phases are hiding, but only that one of them is binding. That is, the binding property only requires that with high probability, the sender will be forced to reveal the correct committed value in at least

box reductions" [Sim98, Rud88, KSS00].

[5] The results of [Ost91, OW93] are stated only for proof systems, but they also hold for argument systems.

[6] As in most treatments of zero knowledge, we use a nonuniform notion of security, and thus require our one-way functions to be secure against nonuniform algorithms (i.e., circuits). Uniform treatments of zero-knowledge proofs and arguments are possible (see [Gol93, BLV04]) but are much more cumbersome.

one of the phases (but which of the two phases can be determined dynamically by the malicious sender).

In [NV06], it was shown that such commitment schemes still suffice to construct zero-knowledge protocols for all of **NP**. Thus, our task is reduced to constructing $\binom{2}{1}$-binding commitment schemes that are statistically hiding and computationally binding from any one-way function. Unfortunately, we do not know how to do this. Instead, we construct polynomially many two-phase commitment schemes, with the guarantee that *at least one* of the schemes is hiding (in both phases), and *all* of the schemes are $\binom{2}{1}$-binding. Fortunately, a similar issue arose also in [NV06] and it was shown how even such a collection could be used to construct zero-knowledge protocols for **NP**.

Even though we draw upon [NV06] for the notion of $\binom{2}{1}$-binding commitments and its utility for zero knowledge, there are many differences between the contexts of the two works and the constructions of $\binom{2}{1}$-binding commitments. In [NV06], the goal was to prove *unconditional* results about prover efficiency in zero-knowledge proofs (that one can transform zero-knowledge proofs with inefficient provers into ones with efficient provers). This was done by showing that every problem having a zero-knowledge proof has an "instance-dependent" $\binom{2}{1}$-binding commitment scheme, where the sender and receiver get an instance $x$ of the problem as auxiliary input and we only require hiding to hold when $x$ is a "yes instance" and binding when $x$ is a "no instance." Here, we are giving *conditional* results (assuming the existence of one-way functions) and are obtaining standard (as opposed to instance-dependent) $\binom{2}{1}$-binding commitments. Moreover, the focus in [NV06] is on proof systems and *statistically* $\binom{2}{1}$-binding commitments; thus here we need to develop new formulations to work with argument systems and the computational binding property.

Our initial construction, which gives a $\binom{2}{1}$-binding commitment scheme satisfying a "weak hiding" property, is inspired by the construction of [NV06]. Indeed, the second phase in [NV06] was also introduced to deal with non-regular functions (corresponding to "non-flat distributions" in their setting), and our construction can be seen as applying the same idea to a variant of the protocol of [HHK$^+$05]. However, in [NV06], this construction immediately gives a "strong hiding" property, whereas much of the technical work in the current paper comes from amplifying the "weak hiding" property we obtain into a strong one.

### 1.3 Outline

We present the basic notations and definitions in Section 2. The statements of our main results are in Section 3. As a warm up, we present a construction (of $\binom{2}{1}$-binding commitments) based on *regular* one-way functions in Section 4, and in Section 5, we present our main ideas on how we can base our construction on *any general* one-way function. The details of the construction are in Sections 6 and 7. In Section 8, we show how to construct statistical zero-knowledge arguments from $\binom{2}{1}$-binding commitment schemes.

## 2 Preliminaries

Let $X$ be a *random variable* taking values in a finite set $T$. We write $x \leftarrow X$ to indicate that $x$ is selected according to $X$. For a finite set $S$, we write $x \leftarrow S$ to indicate that $x$ is selected uniformly amongst all the elements of $S$. The *support* of a random variable $X$ is $\mathrm{Supp}(X) = \{x : \Pr[X = x] > 0\}$. Random variable $X$ is *flat* if it is uniform over its support.

A negligible function, denoted by neg, is a function that vanishes more quickly than any inverse polynomial. That is, for all $c \in \mathbb{N}$, $\text{neg}(n) < n^{-c}$ for all sufficiently large $n$. Let $\text{poly}(n)$ denote any polynomial, that is $\text{poly}(n) \leq n^c$ for some $c \in \mathbb{N}$, and for all sufficiently large $n$.

The *statistical difference* between two random variables $A$ and $B$ over $\{0,1\}^n$ is defined as

$$\Delta(A,B) \stackrel{\text{def}}{=} \max_{T \subseteq \{0,1\}^n} |\Pr[A \in T] - \Pr[B \in T]| = \frac{1}{2} \sum_{x \in \{0,1\}^n} |\Pr[A \in T] - \Pr[B \in T]| \,.$$

We say that distributions $A$ and $B$ are *$\varepsilon$-close* if $\Delta(A,B) \leq \varepsilon$.

Let $I$ be a set of strings. A *probability ensemble* of a sequence of random variables indexed by $I$ is denoted as $\{A_x\}_{x \in I}$. We say that two ensembles $\{A_x\}_{x \in I}$ and $\{B_x\}_{x \in I}$ are *statistically indistinguishable* if there exists a negligible function $\varepsilon$ such that $X_x$ and $Y_x$ are $\varepsilon(|x|)$-close for every $x \in I$. We write $\{A_x\}_{x \in I} \approx_s \{B_x\}_{x \in I}$ to denote that the two ensembles are statistically indistinguishable.

For a probabilistic algorithm $A$, we write $A(x;r)$ to denote the output of $A$ on input $x$ and coin tosses $r$. PPT refers to probabilistic algorithms that run in *strict* polynomial time. A *nonuniform* PPT algorithm is a pair $(A, \bar{z})$, where $\bar{z} = z_1, z_2, \ldots$ is an infinite series of strings where $|z_n| = \text{poly}(n)$, and $A$ is a PPT algorithm that receives pairs of input of the form $(x, z_{|x|})$. (The string $z_n$ is called the *advice string* for $A$ for inputs of length $n$.) Nonuniform PPT algorithms are equivalent to families of polynomial-sized Boolean circuits.

**Definition 2.1** (one-way function)**.** Let $s \colon \mathbb{N} \to \mathbb{N}$ be any function. A function $f \colon \{0,1\}^* \to \{0,1\}^*$ is a *$s(n)$-secure one-way function* if $f$ is computable in polynomial time and for every non-uniform PPT $A$,

$$\Pr_{x \leftarrow \{0,1\}^n}[A(1^n, f(x)) \in f^{-1}(f(x))] < 1/s(n),$$

for all sufficiently large $n$. We say that $f$ is a *one-way function* if $f$ is $s(n)$-secure for every polynomial $s$.

Without loss of generality, we only consider one-way functions that are length-preserving, that is for all $x \in \{0,1\}^*$, $|f(x)| = |x|$. This is because general one-way functions can be converted into ones that are length-preserving (cf., [Gol01, p. 39]). We say that a one-way function $f$ is *regular* with *preimage size* $g(n)$ if there exists a function $g \colon \mathbb{N} \to \mathbb{N}$ such that $\forall y \in \text{Supp}(f(U_n))$, $|\{x \in \{0,1\}^n : f(x) = y\}| = g(n)$.

## 2.1 Statistical Zero-Knowledge Arguments

We follow the standard definitions of zero-knowledge arguments, as in [Gol01, Sec. 4.8]. For an interactive protocol $(A,B)$, we write $(A(a), B(b))(x)$ to denote the random process obtained by having $A$ and $B$ interact on common input $x$, (private) auxiliary inputs $a$ and $b$ to $A$ and $B$, respectively (if any), and independent random coin tosses for $A$ and $B$. We call the protocol $(A,B)$ *public coin* if all of the messages sent by $B$ simply consist of coin tosses (independent of the history), except for the final `accept/reject` message which is computed as a deterministic function of the transcript.

**Definition 2.2.** An interactive protocol $(P,V)$ is an *argument (or computationally sound proof system)* for a language $L$ if the following three conditions hold:

1. (Efficiency) $P$ and $V$ are computable in probabilistic polynomial time.

2. (Completeness) If $x \in L$, then $V$ accepts in $(P, V)(x)$ with probability at least 2/3.

3. (Soundness) If $x \notin L$, then for every nonuniform PPT adversarial prover $P^*$, $V$ accepts in $(P^*, V)(x)$ with probability at most 1/3.

By sequential repetition, both the completeness and soundness error probabilities can be reduced to exponentially small; that is from the completeness from 2/3 to $1 - 2^{-n}$, and soundness from 1/3 to $2^{-n}$. (*Parallel* repetition does not seem to reduce the soundness error of argument systems [BIN97].) All the protocols presented in this paper will have *perfect completeness,* i.e., $V$ accepts with probability 1 when $x \in L$.

We write $\mathsf{view}_A(A(a), B(b))(x)$ to denote $A$'s view of the interaction, i.e., a transcript $(x, \gamma_1, \gamma_2, \ldots, \gamma_t; r)$, where the $\gamma_i$'s are all the messages exchanged and $r$ is $A$'s coin tosses. (Similarly, we can define $\mathsf{view}_B(A(a), B(b))(x)$ to denote $B$'s view of the interaction.) Let $\mathsf{output}_A(A(a), B(b))$ denote $A$'s private output after the interaction. (Similarly, we can define $\mathsf{output}_B(A(a), B(b))(x)$ to denote $B$'s private output after the interaction.) Let $\mathsf{transcript}(A(a), B(b))(x)$ denote the messages exchanged in the protocol including the common input $x$, i.e., $(x, \gamma_1, \gamma_2, \ldots, \gamma_t)$.

**Definition 2.3** (statistical zero knowledge)**.** We say an argument system $(P, V)$ is (black-box) *statistical zero knowledge* if there exists a universal PPT simulator $S$ such that for all verifiers $V^*$, we have

$$\{\mathsf{view}_{V^*}(P, V^*)(x)\}_{x \in L} \approx_s \{S^{V^*}(x)\}_{x \in L}.$$

The above definition of zero knowledge is a *black box* definition in the sense that the simulator is universal for all (even computationally unbounded) verifier strategies $V^*$, and in particular does not depend on the code of $V^*$. The zero-knowledge protocols we construct will all be black-box zero Wknowledge and thus satisfy the above definition.

## 2.2 1-out-of-2-Binding Commitments

We now introduce the notion of $\binom{2}{1}$-binding commitments that will play a central role in establishing our results. These are commitment schemes with two *sequential* and *related* stages such that in each stage, the sender commits to and reveals a value.

**Definition 2.4.** A *2-phase commitment scheme* $(S, R)$, with security parameter $n$ and message length $k = k(n)$, consists of four interactive protocols: $(S_c^1, R_c^1)$ the first commitment stage, $(S_r^1, R_r^1)$ the first reveal stage, $(S_c^2, R_c^2)$ the second commitment stage, and $(S_r^2, R_r^2)$ the second reveal stage. For us, both reveal phases will always be noninteractive, consisting of a single message from the sender to the receiver. Throughout, both parties receive the security parameter $1^n$ as input.

1. In the first commitment stage, $S_c^1$ receives a private input $\sigma^{(1)} \in \{0, 1\}^k$ and a sequence of coin tosses $r_S$. At the end, $S_c^1$ and $R_c^1$ receive as common output a commitment $z^{(1)}$. (Without loss of generality, we can assume that $z^{(1)}$ is the transcript of the first commitment stage.)

2. In the first reveal stage, $S_r^1$ and $R_r^1$ receive as common input the commitment $z^{(1)}$ and a string $\sigma^{(1)} \in \{0, 1\}^k$ and $S_r^1$ receives as private input $r_S$. At the end, $S_r^1$ and $R_r^1$ receive a common output $\tau$. (Without loss of generality, we can assume that $\tau$ is the transcript of the first commitment stage and the first reveal stage and includes $R_r^1$'s decision to accept or reject.)

3. In the second commitment stage, $S_c^2$ and $R_c^2$ both receive the common input $\tau \in \{0,1\}^*$, and $S_c^2$ receives a private input $\sigma^{(2)} \in \{0,1\}^k$ and the coin tosses $r_S$. $S_c^2$ and $R_c^2$ receive as common output a commitment $z^{(2)}$. (Without loss of generality, we can assume that $z^{(2)}$ is the concatenation of $\tau$ and the transcript of the second commitment stage.)

4. In the second reveal stage, $S_r^2$ and $R_r^2$ receive as common input the commitment $z^{(2)}$ and a string $\sigma^{(2)} \in \{0,1\}^k$, and $S_r^2$ receives as private input $r_S$. At the end, $R_r^2$ accepts or rejects.

- $S = (S^1, S^2) = ((S_c^1, S_r^1), (S_c^2, S_r^2))$ and $R = (R^1, R^2) = ((R_c^1, R_r^1), (R_c^2, R_r^2))$ are computable in probabilistic polynomial time.

- We say that $(S, R)$ is *public-coin* if it is public-coin for $R$.

Note that instead of providing $S$ with decommitment values as private outputs of the commitment phases, we simply provide it with the same coin tosses throughout (so it can recompute any private state from the transcripts of the previous phases).

As for standard commitment schemes, we define the security of the sender in terms of a hiding property. Loosely speaking, the hiding property for a 2-phase commitment scheme says that *both* commitment phases are hiding. Note that since the phases are run sequentially, the hiding property for the second commitment stage is required to hold even given the receiver's view of the first stage.

**Definition 2.5** (hiding)**.** 2-phase commitment scheme $(S, R)$, with security parameter $n$ and message length $k = k(n)$, is *statistically hiding* if for all adversarial receiver $R^*$,

1. The views of $R^*$ when interacting with the sender in the first phase on any two messages are statistically indistinguishable. That is, for all $\sigma^{(1)}, \widetilde{\sigma}^{(1)} \in \{0,1\}^k$,

$$\left\{\mathsf{view}_{R^*}(S_c^1(\sigma^{(1)}), R^*)(1^n)\right\}_{n\in\mathbb{N}} \approx_s \left\{\mathsf{view}_{R^*}(S_c^1(\widetilde{\sigma}^{(1)}), R^*)(1^n)\right\}_{n\in\mathbb{N}}.$$

2. The views of $R^*$ when interacting with the sender in the second phase are statistically indistinguishable no matter what the sender committed to in the first phase. That is, for all $\sigma^{(1)}, \sigma^{(2)}, \widetilde{\sigma}^{(2)} \in \{0,1\}^k$,

$$\left\{\mathsf{view}_{R^*}(S_c^2(\sigma^{(2)}), R^*)(\Lambda, 1^n)\right\}_{n\in\mathbb{N}} \approx_s \left\{\mathsf{view}_{R^*}(S_c^2(\widetilde{\sigma}^{(2)}), R^*)(\Lambda, 1^n)\right\}_{n\in\mathbb{N}},$$

where $\Lambda = \mathsf{transcript}(S^1(\sigma^{(1)}), R^*)(1^n)$.

We stress that the second condition of the above hiding definition (Definition 2.5) requires that the view of receiver in the second phase be indistinguishable for any two messages even given the transcript of the first phase, $\Lambda = \mathsf{transcript}(S^1(\sigma^{(1)}), R^*)(1^n)$.

Loosely speaking, the binding property says that *at least* one of the two commitment phases is (computationally) binding. In other words, for every polynomial-time sender $S^*$, there is at most one "bad" phase $j \in \{1,2\}$ such that given a commitment $z^{(j)}$, $S^*$ can open $z^{(j)}$ successfully both as $\sigma^{(1)}$ and $\widetilde{\sigma}^{(1)} \neq \sigma$ with nonnegligible probability. Actually, we allow this bad phase to be determined dynamically by $S^*$. Moreover, we require that the second phase be *statistically* binding if the sender breaks the first phase. Our construction achieves this stronger property, and using it simplifies some of our proofs.

**Definition 2.6** (1-out-of-2-binding)**.** 2-phase commitment scheme $(S, R)$, with security parameter $n$ and message length $k = k(n)$, is *computationally* $\binom{2}{1}$*-binding* if there exist a set $\mathcal{B}$ of first phase transcripts and a negligible function $\varepsilon$ such that:

1. For every (even unbounded) sender $S^*$, the first-phase transcripts in $\mathcal{B}$ make the second phase statistically binding, i.e. $\forall S^*, \forall \tau \in \mathcal{B}$, with probability at least $1-\varepsilon(n)$ over $z^{(2)} = (S^*, R_c^2)(\tau)$, there is at most one value $\sigma^{(2)} \in \{0,1\}^k$ such that $\mathtt{output}(S^*, R_r^2)(z^{(2)}, \sigma^{(2)}) = \mathtt{accept}$.

2. $\forall$ nonuniform PPT $S^*$,[7] $S^*$ succeeds in the following game with probability at most $\varepsilon(n)$ for all sufficiently large $n$:

   (a) $S^*$ and $R_c^1$ interact and output a first-phase commitment $z^{(1)}$.

   (b) $S^*$ outputs two full transcripts $\tau$ and $\widetilde{\tau}$ of *both* phases with the following three properties:
   - Transcripts $\tau$ and $\widetilde{\tau}$ both start with prefix $z^{(1)}$.
   - The transcript $\tau$ contains a successful opening of $z^{(1)}$ to the value $\sigma^{(1)} \in \{0,1\}^k$ using a first-phase transcript not in $\mathcal{B}$, and $R_r^1$ and $R_r^2$ both accept in $\tau$.
   - The transcript $\widetilde{\tau}$ contains a successful opening of $z^{(1)}$ to the value $\widetilde{\sigma}^{(1)} \in \{0,1\}^k$ using a first-phase transcript not in $\mathcal{B}$, and $R_r^1$ and $R_r^2$ both accept in $\widetilde{\tau}$.

   (c) $S^*$ succeeds if all of the above conditions hold and $\sigma^{(1)} \neq \widetilde{\sigma}^{(1)}$.

# 3 Our Results

Our main theorem, Theorem 1.1, is established via the following theorems.

**Theorem 3.1.** *If one-way functions exist, then on security parameter $n$, we can construct in time* $\mathrm{poly}(n)$ *a collection of public-coin 2-phase commitment schemes* $\mathrm{Com}_1, \cdots, \mathrm{Com}_m$ *for* $m = \mathrm{poly}(n)$ *such that:*

- *There exists an index $i \in [m]$ such that scheme* $\mathrm{Com}_i$ *is statistically hiding.*
- *For every index $j \in [m]$, scheme* $\mathrm{Com}_j$ *is computationally* $\binom{2}{1}$*-binding.*

**Theorem 3.2.** *Assume that on security parameter $n$, we can construct in time* $\mathrm{poly}(n)$ *a collection of public-coin 2-phase commitment schemes* $\mathrm{Com}_1, \cdots, \mathrm{Com}_m$ *for* $m = \mathrm{poly}(n)$ *such that:*

- *There exists an index $i \in [m]$ such that scheme* $\mathrm{Com}_i$ *is statistically hiding.*
- *For every index $j \in [m]$, scheme* $\mathrm{Com}_j$ *is* $\binom{2}{1}$*-computationally binding.*

*Then, every language in* **NP** *has a public-coin statistical zero-knowledge argument system.*

The proof of Theorem 3.2 is very similar to that in [NV06] for $\binom{2}{1}$-statistically binding commitments, with a bit more work to handle the *computational* binding property, and can be found in Section 8.

[7] Definitions of cryptographic primitives in the literature often use the reverse order of quantifiers, asking that for every (nonuniform) PPT adversary $S^*$, there exists a negligible function $\varepsilon(n)$ such that the success probability of $S^*$ is at most $\varepsilon(n)$. However, the two resulting definitions turn out to be equivalent [Bel02].

# 4 Warm-up: 1-out-of-2-Binding Commitments from Regular One-Way Functions

In this section, as a warm-up to Section 5 where we construct secure 2-phase commitments from general one-way functions, we first describe a standard commitment scheme from a regular one-way function with known preimage size (based on [HHK$^+$05]), and then show how to construct a collection of statistically hiding, computationally $\binom{2}{1}$-binding commitments from regular one-way functions with unknown pre-image size.

The tools used in these commitments schemes are pairwise-independent hash functions and interactive hashing protocols, both described in the next subsections.

## 4.1 Hashing and Randomness Extraction

**Entropy.** The *entropy* of a random variable $X$ is $H(X) = \mathrm{E}_{x \overset{\mathrm{R}}{\leftarrow} X}[\log(1/\Pr[X = x])])$, where here and throughout the paper all logarithms are to base 2. Intuitively, $H(X)$ measures the amount of randomness in $X$ *on average* (in bits). The *min-entropy* of $X$ is $H_\infty(X) = \min_x[\log(1/\Pr[X = x])]$; this is a "worst-case" measure of randomness. In general $H_\infty(X) \leq H(X)$, but if $X$ is flat (i.e. uniform on its support), then $H(X) = H_\infty(X) = \log|\mathrm{Supp}(X)|$.

A family of hash functions $\mathcal{H}_{a,b} = \{h : \{0,1\}^a \to \{0,1\}^b\}$ is *pairwise independent* if for any two $x \neq x' \in \{0,1\}^a$ and any two $y, y' \in \{0,1\}^b$, when we randomly choose $h \leftarrow \mathcal{H}_{a,b}$, we have: $\Pr[h(x) = y \wedge h(x') = y'] = \frac{1}{2^{2b}}$. We define $\ell(a, b)$ to be the number of bits required to describe an element of the hash function family $\mathcal{H}_{a,b}$; that is, $\ell(a, b) = \max\{a, b\} + b$. We will use the following strong extractor property of $\mathcal{H}_{a,b}$.

**Lemma 4.1** (Leftover Hash Lemma [BBR88, ILL89])**.** *Let $\mathcal{H}_{a,b}$ be a pairwise independent family of hash functions mapping $\{0,1\}^a$ to $\{0,1\}^b$. Let $Z$ be a random variable taking values in $\{0,1\}^a$ such that $\mathcal{H}_\infty(Z) \geq b + 2\log(1/\varepsilon)$. Then the following distribution has statistical difference at most $\varepsilon$ from the uniform distribution on $\mathcal{H}_{a,b} \times \{0,1\}^b$: Choose $h \leftarrow \mathcal{H}_{a,b}$ and $x \leftarrow Z$ and output $(h, h(x))$.*

## 4.2 Interactive Hashing

Ostrovsky, Venkatesan and Yung [OVY93] introduced a powerful tool known as *interactive hashing (IH)*, which is a protocol between a sender $S_{\mathrm{IH}}$ and receiver $R_{\mathrm{IH}}$. The sender begins with a private input $y$, and at the end both parties outputs $y_0$ and $y_1$ such that $y \in \{y_0, y_1\}$. Informally, the IH protocol has the following properties:

1. *(Hiding)* If the sender's input $y$ is uniformly random, then the receiver does not learn which of $y_0$ or $y_1$ equals to $y$.

2. *(Binding)* The sender can "control" the value of at most one of the two outputs.

Naor, Ostrovsky, Venkatesan and Yung [NOVY98] showed that interactive hashing can be used to construct statistically hiding commitment schemes from one-way permutations.

We extend the notion of interactive hashing to allow multiple outputs (instead of just two output strings). Since we allow the number of outputs to be possibly superpolynomial, we succinctly describe the set of outputs as the image of a polynomial-sized circuit $C\colon \{0,1\}^k \to \{0,1\}^q$, where $k$ and $q$ are polynomially related to the security parameter.

For a relation $W$, let $W_y = \{z : W(y,z) = 1\}$ and we refer to any $z \in W_y$ as a *valid witness* for $y$. In the definitions below, we use general relations, and hence do not require that relation $W$ be polynomial-time computable.

**Definition 4.2.** An *interactive hashing scheme with multiple outputs* is a polynomial-time protocol $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ where both parties receive common inputs $(1^q, 1^k)$, $S_{\mathrm{IH}}$ receives a private input $y \in \{0,1\}^q$, with the common output being a circuit $C\colon \{0,1\}^k \to \{0,1\}^q$, and the private output of $S_{\mathrm{IH}}$ being a string $z \in \{0,1\}^k$. We denote $q$ to be the input length and $k$ to be the output length. The protocol $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ has to satisfy the following security properties:

1. *(Correctness)* For all $R^*$ and all $y \in \{0,1\}^q$, letting $C = (S_{\mathrm{IH}}(y), R^*)(1^q, 1^k)$ and $z = \mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(y), R^*)$, we have that $C(z) = y$.

2. *(Perfect hiding)* For all $R^*$, $(V, Z)$ is distributed identically to $(V, U_k)$, where $V = \mathsf{view}_{R^*}(S_{\mathrm{IH}}(U_q), R^*)$ and $Z = \mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(U_q), R^*)$.

3. *("Computational" binding)* There exists an oracle PPT algorithm $A$ such that for every $S^*$ and any relation $W$, letting circuit $C = (S^*, R_{\mathrm{IH}})(1^q, 1^k)$ and $((x_0, z_0), (x_1, z_1)) = \mathsf{output}_{S^*}(S^*, R_{\mathrm{IH}})$, if it holds that
$$\Pr[x_0 \in W_{C(z_0)} \wedge x_1 \in W_{C(z_1)} \wedge z_0 \neq z_1] > \varepsilon,$$
where the above probability is over the coin tosses of $R_{\mathrm{IH}}$ and $S^*$. Then we have that
$$\Pr_{y \leftarrow \{0,1\}^q}[A^{S^*}(y, 1^q, 1^k, \varepsilon) \in W_y] > 2^{-k} \cdot (\varepsilon/q)^{O(1)}.$$

We make three remarks regarding the above definition.

1. The security requirements should hold for all, even computationally unbounded $R^*$ (for correctness and perfect hiding) and computationally unbounded $S^*$ (even though binding is "computational"). In addition, the relation $W$ need not be polynomial-time computable.

2. To simplify notation, we often write $A^{S^*}(y)$, or even $A(y)$, to denote $A^{S^*}(y, 1^q, 1^k, \varepsilon)$.

3. Although the output of the honest sender $S_{\mathrm{IH}}$ is always a string $z$, the output of the cheating sender $S^*$ is arbitrary; hence, we can assume without loss of generality that $S^*$ breaks binding by producing two pairs of strings $(x_0, z_0)$ and $(x_1, z_1)$.

We think of the string $z \in \{0,1\}^k$ as a $k$-bit string commitment associated to one of the $2^k$ outputs strings, namely $y = C(z)$, and a witness $x \in W_y = W_{C(z)}$ as a decommitment to $z$. Intuitively, the knowledge of $x$ gives the sender the ability to decommit to $z$. The "computational" binding property, read in its contrapositive, says that if it is hard to find a witness for a uniformly random string $y$, then it is hard for a sender to successfully decommit to two different values. Notice that this property holds even if the set of "hard" $y$'s is not fixed in advance, but depends on the algorithm trying to find a witness for $y$, namely an element in $W_y$. In several places, however, we will only need the special case of a static set of $y$'s as captured in the following lemma.

The computational binding property in Definition 4.2 can be extended to the case of static sets as in following lemma.

**Lemma 4.3** (binding for static sets)**.** *For any protocol* $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ *satisfying the computational binding condition of Definition 4.2, the following holds: For all* $S^*$ *and any set* $T \subseteq \{0,1\}^q$*, letting* $C = (S^*, R_{\mathrm{IH}})(1^q, 1^k)$*, we have*

$$\Pr[\exists z_0 \neq z_1 \ s.t. \ C(z_0), C(z_1) \in T] < (\mu(T) \cdot 2^k)^{\Omega(1)} \cdot \mathrm{poly}(q),$$

*where the above probability is taken over the coin tosses of* $S^*$ *and* $R_{\mathrm{IH}}$*.*

Compare the bound of the above lemma to the case where adversarial sender $S^*$ had control of only one output string. This means that the rest of the $2^k - 1$ outputs strings are distributed uniformly on $\{0,1\}^q$, and hence the bound would be $\mu(T) \cdot (2^k - 1)$. ($S^*$ will make the string that it controls lie in $T$, and the probability that at least one of the rest of the $2^k - 1$ strings lie in $T$ is at most $\mu(T) \cdot (2^k - 1)$, by a union bound argument.) The above bound is almost as good, and in particular if $\mu(T)$ is negligible and $k$ logarithmic, both probabilities are negligible.

*Proof of Lemma 4.3.* Define the relation $W = \{(a,b) : a \in T\}$, that is $W(a,b) = 1$ if $a \in T$ (for all values of $b$), and 0 if $a \notin T$ (no matter what the value of $b$ is). Hence, if we have $S^*$ that violates the lemma—by outputting two elements $z_0 \neq z_1$ with $C(z_0), C(z_1) \in T$—with probability $\varepsilon$, there will be a procedure over a random $y \leftarrow \{0,1\}^q$ makes $y \in T$ with probability $2^{-k} \cdot (\varepsilon/q)^{O(1)}$, by the computational binding condition of Definition 4.2. Since $T$ is a fixed set, it must be the case that $2^{-k} \cdot (\varepsilon/q)^{O(1)} \leq \mu(T)$. This implies that $\varepsilon < (\mu(T) \cdot 2^k)^c \cdot \mathrm{poly}(q)$, for some constant $c > 0$. □

We extend the proof in [NOVY98] to obtain the following theorem. The protocol is obtained by simply ending the NOVY protocol $k-1$ rounds earlier. The proof that it satisfies Definition 4.2 is given in Appendix B.

**Theorem 4.4.** *There exists an interactive hashing scheme with multiple outputs, namely Protocol 4.5.*

**Protocol 4.5.** Interactive Hashing Scheme with Multiple Outputs $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$.

**Inputs:**

1. Input length $1^q$ and output length $1^k$, both given as common input.
2. String $y \in \{0,1\}^q$, given as private input to sender $S_{\mathrm{IH}}$.

**Protocol:**

$R_{\mathrm{IH}}$: Select $h_0, h_1, \dots, h_{q-k-1}$ such that each $h_i$ is a random vector over GF[2] of the form $0^i 1\{0,1\}^{q-i-1}$ (i.e., $i$ number of 0's followed by a 1, and random choice for the last $q-i-1$ positions).

For $j = 0, \dots, q-k-1$, do the following:

$R_{\mathrm{IH}} \to S_{\mathrm{IH}}$: Send $h_j$.
$S_{\mathrm{IH}} \to R_{\mathrm{IH}}$: Send $c_j = \langle h_j, y \rangle$.

**Output:**

- Common output is a circuit $C\colon \{0,1\}^k \to \{0,1\}^q$. computing an affine transformation whose image is $\{y : \langle h_j, y\rangle = c_j \ \ \forall j = 0, \dots, q-k-1\}$.
- Output of $S_{\mathrm{IH}}$ is a string $z \in \{0,1\}^k$ such that $C(z) = y$. (In fact, $z$ can be taken to be the last $k$ bits of $y$.)

### 4.3 From Regular One-Way Function with Known Preimage Size

We first informally describe a (standard) commitment scheme from a regular one-way function with known preimage size and known hardness $s(n) = n^{\omega(1)}$, based loosely on Haitner et al. [HHK$^+$05] (who prove a stronger result, no needing to know the hardness).

Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be a regular one-way function such that the entropy $H(f(U_n)) = t$ is known (this is equivalent to knowing the preimage size of $f$). In the commitment scheme, the sender $S$ generates a random string $x \in \{0,1\}^n$ and sets $y = f(x)$. $S$ picks a random hash function $h\colon \{0,1\}^n \to \{0,1\}^{t-\Delta}$ where $\Delta = (\log s(n))/2$. $(S, R)$ then run the interactive hashing protocol (with $k = 1$) with $S$ having input $(h, h(y))$. Their common output is a pair $(w_0, w_1) = (C(0), C(1))$, and the sender receives $d \in \{0,1\}$ such that $w_d = w$. To commit to the bit $b$, $S$ sends $c = d \oplus b$. The commitment $z$ is defined as $(w_0, w_1, c)$. In the reveal phase, $S$ sends $b$, $d$, and the string $x \in \{0,1\}^n$ used to generate $y$. $R$ checks that $f(x) = y$, $c = d \oplus b$, and $w_d$ is of the form $(h, h(y))$.

Intuitively, the commitment scheme is hiding since there are $2^t$ possible values of $y$ hence $(h, h(y))$ is $(1/s(n))^{\Omega(1)}$-close to the uniform distribution by the Leftover Hash Lemma (Lemma 4.1), which implies that the commitment scheme is hiding by the hiding property of interactive hashing. As for the binding property, the one-wayness of $f$ intuitively guarantees that the set $T$ of $y$'s for which a sender $S^*$ can compute an element of $f^{-1}(y)$ is of density at most $2^{-s(n)}$ in Image($f$), i.e. of size at most $2^{H(f(U_n))-s(n)}$. Thus the set of pairs $(h, h(y))$ such that $y \in T$ has density at most $2^{H(f(U_n))-s(n)}/2^{t-\Delta} = s(n)^{9/10} = \mathrm{neg}(n)$. By the binding property of interactive hashing (Lemma 4.3), the probability that $S^*$ can force both $w_0, w_1 \in T$ is negligible and the scheme is computationally binding. (The complete argument to prove the binding property is actually more subtle because the set $T$ is not actually fixed in advance, and we need to use the computationally binding property of interactive hashing given in Definition 4.2)

## 4.4 From Regular One-Way Function with Unknown Preimage Size

We show that if regular one-way functions with known hardness exist, then on security parameter $1^n$, we can construct a collection of 2-phase commitment schemes $\mathrm{Com}_1, \cdots, \mathrm{Com}_n$ such that:

- There exists an index $i \in [n]$ such that scheme $\mathrm{Com}_i$ is statistically hiding.
- For every index $j \in [n]$, scheme $\mathrm{Com}_j$ is $\binom{2}{1}$-computationally binding.

To deal with the case where the preimage size is unknown, a first attempt would be to try all possible values of $t$ in the protocol sketched above in Section 4.3 and obtain a collection of standard commitments. However, the above commitment scheme only seems to be computationally binding when $t \gtrsim H(f(U_n))$ (and is hiding when $t \lesssim H(f(U_n))$) not matching the guarantees of the desired collection of commitments.

We will in fact use the above protocol as the first phase. However, we also introduce a *second* phase that will be binding when $t \lesssim H(f(U_n))$ and hiding when $t \gtrsim H(f(U_n))$. This will be obtained by the sender using (a hash of) the preimage $x$ as an input to another execution of interactive hashing. Note that given $y = f(x)$, $x$ is distributed uniformly over a set of size $|f^{-1}(y)| = 2^{n-H(f(U_n))}$ so hiding and binding follow from the properties of interactive hashing. In fact these schemes for regular one-way functions achieve a stronger property than $\binom{2}{1}$-binding. For each value of $t$, either the first phase is always binding or the second phase is always binding (i.e. the sender cannot choose which binding property to break). However, we will in fact show that $\binom{2}{1}$-binding in the sense of Definition 2.6 is achieved for *any* one-way function $f$, regardless of whether it is regular. We use this $\binom{2}{1}$-binding commitment for each possible value of $t$. This ensures that all are $\binom{2}{1}$-binding and at least one of the commitments in this collection is hiding.

## 4.5 The Protocol

Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be any function, not necessarily regular nor one-way—as we shall later see, the regularity condition and one-way security of the function give us the hiding and binding properties, respectively. Let $\mathcal{H}_{a,b} = \{h_{a,b}\colon \{0,1\}^a \to \{0,1\}^b\}$ be a family of pairwise hash functions. The description of each element in $\mathcal{H}_{a,b}$ takes $\ell(a,b) = \max\{a,b\} + b < 2(a+b)$ bits. For $a, b < \mathrm{poly}(n)$, it is convenient to make $\ell(a,b) = q(n) - b$, for some fixed polynomial $q(n)$, so that for every $y \in \{0,1\}^a$, $|h, h(y)| = q(n)$. This can be done by padding random bits to the description of $h$.

In addition, it will be convenient to work with protocols where the sender has no input $\sigma$ to be committed to, but rather privately receives an output $d \in \{0,1\}^k$ at the end of each phase of the commitment. If we can ensure that $d$ is (nearly) uniform given the receiver's view, such a protocol can be tuned into a standard commitment scheme, where the sender can commit to an $\sigma$ of its choice by sending $d \oplus \sigma$ at the end of the commit phase.

**Protocol 4.6.** 2-Phase Commitment Scheme $(S, R)$ based on $f\colon \{0,1\}^n \to \{0,1\}^n$.

**Parameters:** Integers $t \in \{1, 2, \ldots, n\}$, $k \in \{1, 2, \ldots, n\}$, $\Delta_1 \in \{0, 1, \ldots, t\}$, and $\Delta_2 \in \{0, 1, \ldots, n - t\}$.

**Sender's private input:** String $x \in \{0,1\}^n$. (Note that this is not the value to which the sender is committing, but is rather part of its coin tosses, which will be chosen uniformly at random by $S$ unless otherwise specified.)

**First phase commit:**

1. $S_c^1$ sets $y = f(x)$.
2. Let $\mathcal{H}_1 = \{h_1\colon \{0,1\}^n \to \{0,1\}^{t-\Delta_1}\}$ be a family of pairwise independent hash functions. $S_c^1$ chooses a random hash $h_1 \leftarrow \mathcal{H}_1$, and computes $v = (h_1, h_1(y)) \in \{0,1\}^q$.
3. $(S_c^1, R_c^1)$ run Interactive Hashing Scheme (Protocol 4.5) $(S_{\mathrm{IH}}(v), R_{\mathrm{IH}})(1^q, 1^k)$, with $S_c^1$ and $R_c^1$ acting as $S_{\mathrm{IH}}$ and $R_{\mathrm{IH}}$ respectively.

   Let circuit $C^{(1)}\colon \{0,1\}^k \to \{0,1\}^q$ be the common output and $d^{(1)} \in \{0,1\}^k$ be $S_{\mathrm{IH}}$'s private output in $(S_{\mathrm{IH}}(v), R_{\mathrm{IH}})(1^q, 1^k)$.

*First phase sender's private output:* String $d^{(1)} \in \{0,1\}^k$.

**First phase reveal:**

$S_r^1$ sends the tuple $\gamma^{(1)} = (d^{(1)}, y, h_1)$.

Receiver $R_r^1$ accepts if and only if $C^{(1)}(d^{(1)}) = (h_1, h_1(y))$.

**Second phase commit:**

*Second phase common input:* First-phase transcript $\tau = \mathsf{transcript}(S^1(x), R^1)$, which in particular includes the string $y$.

1. Let $\mathcal{H}_2 = \{h_2\colon \{0,1\}^n \to \{0,1\}^{n-t-\Delta_2}\}$ be a family of pairwise independent hash functions. $S_c^2$ chooses a random hash $h_2 \leftarrow \mathcal{H}_2$, and computes $w = (h_2, h_2(x)) \in \{0,1\}^q$.
2. $(S_c^2, R_c^2)$ run Interactive Hashing Scheme (Protocol 4.5) $(S_{\mathrm{IH}}(w), R_{\mathrm{IH}})(1^q, 1^k)$, with $S_c^2$ and $R_c^2$ acting as $S_{\mathrm{IH}}$ and $R_{\mathrm{IH}}$ respectively.

   Let circuit $C^{(2)}\colon \{0,1\}^k \to \{0,1\}^q$ be the common output and $d^{(2)} \in \{0,1\}^k$ be $S_{\mathrm{IH}}$'s private output in $(S_{\mathrm{IH}}(v), R_{\mathrm{IH}})(1^q, 1^k)$.

*Second phase sender's private output:* String $d^{(2)} \in \{0,1\}^k$.

**Second phase reveal:**

$S_r^2$ sends the tuple $\gamma^{(2)} = (d^{(2)}, x, h_2)$.

Receiver $R_r^2$ accepts if and only if $f(x) = y$ and $C^{(2)}(d^{(2)}) = (h_2, h_2(x))$.

**Lemma 4.7** (statistical hiding)**.** *If $f$ is a regular function with $H(f(U_n)) \in (t_0 - 1, t_0]$, then*

*Protocol 4.6, with setting of parameters $t = t_0$, $k \leq q(n)$, and $\Delta_1 = \Delta_2 = \omega(\log n)$, is statistically hiding in the sense of Definition 2.5.*

*Proof Sketch.* For every $y \in \text{Support}(f(U_n))$, we have $p(y) = \Pr[f(U_n) = y] \in [2^{-t_0}; 2^{-t_0+1})$

We denote the distribution $f(U_n)$ by $Y$. The flat source $Y$ has min-entropy at least $t_0 - 1$. By the Leftover Hash Lemma (Lemma 4.1), the distribution $Z = (H_1, H_1(Y))$ is $2^{-\Omega(\Delta_1)}$-close to the uniform distribution $(H_1, U_{t-\Delta_1})$. By the hiding property of interactive hashing, the first commitment phase is $2^{-\Omega(\Delta_1)}$-statistically hiding.

Let $\tau$ be the transcript of the first phase and $y$ the string sent in the first reveal phase. Conditioned on $\tau$, the string $x$ comes from the uniform distribution $X$ over $f^{-1}(y)$ and $X$ is a flat souce with min-entropy at least $n - t_0$. By the Leftover Hash Lemma (Lemma 4.1), the distribution $W = (H_2, H_2(X))$ is $2^{-\Omega(\Delta_2)}$-close to the uniform distribution $(H_2, U_{n-t-\Delta_2})$. By the hiding property of interactive hashing, the second commitment phase is $2^{-\Omega(\Delta_2)}$-statistically hiding. □

**Lemma 4.8.** *If $f$ is a $s(n)$-secure one-way function (not necessarily regular), then for any value of $t \in \{1, \cdots, n\}$, Protocol 4.6, with setting of parameters $k = O(\log n)$, $\Delta_1 = \Delta_2 \leq (\log(s(n)))/4$, is 1-out-of-2 computationally binding in the sense of Definition 2.6.*

The proposition will be proved in two steps. For every $t \in \{1, \cdots, n\}$, we define the set of "light" strings $L_t = \{y \in \{0,1\}^n : \Pr_{U_n}[f(U_n) = y] \leq 2^{-t-\Delta_3}\}$, for a parameter $\Delta_3$ that we will set at the end of the proof. We define $\mathcal{B}$ to be the set of transcripts where the sender reveals $y \in L_t$. We will first show that if the first commitment transcript is in $\mathcal{B}$, then the second phase will be statistically binding. We will then prove that the first phase is computationally binding, i.e. if there exists an adversary that can break the binding property for the first phase, then there exists an adversary that can invert $f$ with nonnegligible success probability.

**Claim 4.9.** *For the binding set $\mathcal{B}$ defined above, Condition 1 of Definition 2.6 is satisfied with $\varepsilon(n) = \text{poly}(n) \cdot 2^{-\Omega(\Delta_3-\Delta_2)}$.*

*Proof of Claim 4.9.* Let $y$ be the string sent in the first reveal phase. Let $T = \{(h_2, h_2(x)) : h_2 \in \mathcal{H}_{n,n-t-\Delta_2}, x \in f^{-1}(y)\}$ and $\mu(T)$ denote the density of the subset $T$. Since $h_2$ maps $\{0,1\}^n$ to $\{0,1\}^{n-t-\Delta_2}$, we have

$$\mu(T) \leq |f^{-1}(y)| \cdot \frac{1}{2^{n-t-\Delta_2}} \leq (2^n \cdot 2^{-t-\Delta_3}) \cdot \frac{1}{2^{n-t-\Delta_2}} = 2^{(\Delta_2-\Delta_3)}$$

By the binding property of the second execution of the interactive hashing protocol for static sets, we have

$$\Pr\left[(w_0, w_1) = \mathsf{output}(S^\star_{\text{IH}}, R_{\text{IH}}) \text{ satisfies } w_0 \in T \wedge w_1 \in T\right] < 2^{-\Omega(\Delta_3-\Delta_2)} \cdot \text{poly}(q).$$

□

**Claim 4.10.** *For the binding set $\mathcal{B}$ defined above, if there exists a PPT $S^\star$ that succeeds with nonnegligible success probability $\varepsilon$ in the game in Condition 2 of Definition 2.6, then there exists a PPT $T$ that can invert $f$ with success probability at least*

$$\varepsilon^{O(1)} \cdot 1/\text{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_3)}.$$

*Proof of Claim 4.10.* We define the relation $\mathcal{R}$:

$$\mathcal{R} = \{((h_1, w), (y, x)) : \ w = h_1(y), \ y = f(x), \ y \notin L_t\}$$

Let $\mathcal{R}_v = \{(y, x) : \mathcal{R}(v, (y, x)) = 1\}$. Suppose we have a PPT $S^\star$ with success probability greater than $\varepsilon$ in the game of Definition 2.6. Then we have a PPT $S^*_{\mathrm{IH}}$ in the interactive hashing protocol such that

$$\begin{aligned}\Pr[\mathsf{output}_{S^*_{\mathrm{IH}}}(S^*_{\mathrm{IH}}, R_{\mathrm{IH}}) = ((v_0, v_1), (y, x), (y', x')) \text{ such that} \\ (v_0, v_1) = \mathsf{output}(S^*_{\mathrm{IH}}, R_{\mathrm{IH}}), (y, x) \in \mathcal{R}_{v_0}, (y', x') \in \mathcal{R}_{v_1}] \quad \geq \quad \varepsilon\end{aligned}$$

By the binding property of the interactive hashing protocol, there exists a PPT $A$ such that

$$\Pr_{v \leftarrow H_1 \times U_{t-\Delta_1}}[A(v, 1^{\ell_1}, \varepsilon) \in \mathcal{R}_v] > 2^{-k} \cdot \left(\frac{\varepsilon}{\ell_1}\right)^c$$

Consider the PPT $T$ that on input $y$ picks a hash function $h_1$ uniformly from $\mathcal{H}_{n,t-\Delta_1}$, runs $A$ on input $v = (h_1, h_1(y))$ and outputs the second component of $A(v)$. Assume without loss of generality that $A$ is deterministic. Then:

$$\begin{aligned}
&\Pr_{U_n, r_B}[T(f(U_n)) \in f^{-1}(f(U_n))] \\
= \quad & \Pr_{H_1, U_n}[A(H_1, H_1(f(U_n)))_2 \in f^{-1}(f(U_n))] \\
\geq \quad & \sum_{(h_1,w) \in \mathcal{H}_{n,t-\Delta_1} \times \{0,1\}^{t-\Delta_1}} \Pr_{U_n, H_1, r_A}[(H_1, H_1(f(U_n))) = (h_1, w) \wedge A(h_1, w) \in \mathcal{R}_{(h_1,w)}] \\
= \quad & \frac{1}{|\mathcal{H}_{n,t-\Delta_1}|} \sum_{(h_1,w) \ s.t. \ A(h_1,w) \in \mathcal{R}_{(h_1,w)}} \Pr[h_1(f(U_n)) = w] \\
\geq \quad & \frac{1}{|\mathcal{H}_{n,t-\Delta_1}|} \sum_{(h_1,w) \ s.t. \ A(h_1,w) \in \mathcal{R}_{(h_1,w)}} \Pr[f(U_n) = A(h_1, w)_1] \\
\geq \quad & \frac{1}{|\mathcal{H}_{n,t-\Delta_1}|} \cdot \left(|\mathcal{H}_{n,t-\Delta_1}| \cdot 2^{t-\Delta_1} \cdot 2^{-k} \cdot \left(\frac{\varepsilon}{\ell_1}\right)^c\right) \cdot 2^{-t-\Delta_3} \\
= \quad & \left(\frac{\varepsilon}{\ell_1}\right)^c \cdot 2^{-(k+\Delta_1+\Delta_3)}
\end{aligned}$$

The first inequality comes from considering fixed values of $h_1$ and $w$ and restricting the success probability of $A$ to the case where $y \notin L_t$. The third inequality comes from considering only values of $(h_1, w)$ such that $w = h_1(y)$ for some $y \notin L_t$. Such strings $y$ have mass at least $2^{-t-\Delta_3}$. □

The lemma follows from the above two claims by setting $\Delta_3 = \Delta_2 + (\log s(n))/4 \leq (\log s(n))/4$. With this, Claim 4.9 shows that Condition 1 in Definition 2.6 is satisfied with $\varepsilon(n) = \mathrm{poly}(n) \cdot 2^{-\Omega(\log s(n))} = \mathrm{neg}(n)$ because $s(n) = n^{\omega(1)}$. Condition 2 of Definition 2.6 is satisfied with negligible probability $\varepsilon(n)$ because otherwise $f$ can be inverted with probability

$$\begin{aligned}\varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_3)} &\geq \varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot 2^{-(O(\log n)+(3/4)\cdot(\log s(n)))} \\ &= \varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot s(n)^{-3/4},\end{aligned}$$

which is greater than $1/s(n)$ if $\varepsilon$ is nonnegligible.

# 5 Overview of Construction for General One-Way Functions

We now present an overview of how we generalize our construction for regular one-way functions with unknown preimage size (Protocol 4.6) to arbitrary one-way functions. As shown in Lemma 4.8, this protocol already achieves $\binom{2}{1}$-binding when $f$ is any one-way function (for every value of $t$). Thus the challenge is the hiding property. (Another issue is that Protocol 4.6 requires a one-way function with known security. It turns out that our method for handling the hiding property also eliminates the need to know the security.)

As discussed in Section 4, for regular one-way functions, Protocol 4.6 has a hiding first phase when the parameter $t$ satisfies $t \lesssim H(f(U_n))$ and has a hiding second phase when $t$ satisfies $t \gtrsim H(f(U_n))$. Intuitively, when $f$ is not regular, we should replace the fixed value $H(f(U_n))$ with the 'dynamic' value $\mathcal{H}_f(y) \stackrel{\mathrm{def}}{=} \log(1/\Pr[f(U_n) = y])$, where $y = f(x)$ is the value chosen by the sender in the pre-processing step, because $\mathcal{H}_f(y)$ can be viewed as measuring the amount of "entropy" in $y$. The "approximable preimage-size one-way functions" studied by Haitner et al. [HHK$^+$05] come equipped with an algorithm that estimates $\mathcal{H}_f(y)$, but for general one-way functions, this quantity may be infeasible to compute.

**A weakly hiding scheme (details in Section 6).** One natural approach is to have the sender choose $t$ at random and "hope" that it is close to $\mathcal{H}_f(y)$. When we choose $t$ too small, only the first phase will be hiding, and when we choose $t$ too large, only the second phase will be hiding. But we have a nonnegligible probability $\delta$ (specifically, $\delta = 1/n$) that $t \approx \mathcal{H}_f(y)$, and thus both phases will be hiding. Thus we have a $\binom{2}{1}$-binding commitment scheme satisfying a "weak hiding" property, where with probability $\delta$, both phases are hiding, and it is always the case that at least one phase is hiding. Actually, in order to simplify our analysis, we will include $t$ as a parameter to the protocol. Then there exists a choice of $t$ such that the protocol is weakly hiding in the sense above, and for all choices of $t$ the protocol is $\binom{2}{1}$-binding. At the end, we will enumerate over all values of $t$, resulting in a *collection* of commitment schemes as claimed in Theorem 3.1, albeit with a weak hiding property.

Unfortunately, we do not know how to directly construct zero-knowledge arguments from a weakly hiding $\binom{2}{1}$-binding commitment scheme. Thus instead, much of the effort in this paper is devoted to amplifying the weak hiding property ($\delta = 1/n$) into a strong hiding property ($\delta = 1 - \mathrm{neg}(n)$), while maintaining the $\binom{2}{1}$-binding property.

**Amplifying the hiding property (details in Section 7).** Inspired by the breakthrough results of Reingold [Rei05] and Dinur [Din06] on different topics, we do not amplify the hiding probability from $\delta = 1/n$ to $\delta = 1 - \mathrm{neg}(n)$ in "one shot," but instead perform a sequence of $\log n$ iterations, each one of which increases $\delta$ by a roughly factor of 2. This results in $\delta = \Omega(1)$, and then we are able to get $\delta = 1 - \mathrm{neg}(n)$ in just one more amplification step.

How do we double $\delta$? A natural idea is to consider several, executions of the previous weakly hiding scheme. Specifically, if we take $m = O(1)$ executions, the probability that at least one of the executions has both phases hiding is roughly $m \cdot \delta$. Moreover, each of the remaining $m-1$ executions have either the first phase hiding or the second phase hiding. Thus for some value of $\beta$, there are $\beta + 1$ first phases that are hiding and $m - \beta$ second phases that are hiding. It turns out that we can choose $\beta$ so that this exact $(\beta+1, m-\beta)$ breakdown given that one execution has both phases hiding occurs with probability $\Omega(1/\sqrt{m})$. Thus we are in the situation described with probability $m \cdot \delta \cdot \Omega(1/\sqrt{m}) > 2\delta$, for a large enough constant $m$.

Now our aim is to combine the outcomes of the weakly hiding schemes in such a way that when the above-described situation occurs, which happens with probability at least $2\delta$, both phases are hiding. Notice that the secret values $\sigma_1, \ldots, \sigma_m \in \{0,1\}^k$ to which the sender commits in the first commit phases have entropy (even min-entropy) at least $(\beta+1)\cdot k$ conditioned on the receiver's view (because $(\beta+1)$ of them are hiding), and similarly the sender's secrets in the second commit phases have entropy at least $(m-\beta)\cdot k$ conditioned on the receiver's view. Let us compare this to the situation with binding. Since each execution is $\binom{2}{1}$-binding, a cheating polynomial-time sender can break the binding property for either at most $\beta$ of the first phases or at most $m-\beta-1$ of the second phases. Thus the number of possible values to which the sender can open in each case is at most $2^m \cdot 2^{k\cdot\beta}$ in the first phase or at most $2^{k\cdot(m-\beta-1)}$, where the $2^m$ factor in the first bound comes from the sender's ability to choose which subset of executions to break (and it is this factor that limits us to taking $m$ to be a constant). We can view these as strong forms of "entropy" upper bounds $m + k\beta$ and $k\cdot(m-\beta-1)$. In at least one phase, there will be an entropy gap of at least $k-m$.

How can we exploit these entropy gaps? It turns out that interactive hashing, again, is a useful tool. Specifically, in the first phase we have the sender choose a random pairwise independent hash function $h_1$ mapping to approximately $(\beta+1)\cdot k$ bits and use $(h_1, h_1(\sigma_1,\ldots,\sigma_m))$ as the input to an Interactive Hashing protocol, and analogously for the second phase. By the Leftover Hash Lemma, this pairwise independent hashing converts the min-entropy lower bound described above to an almost-uniform distribution, so the Interactive Hashing hiding property applies. As for the binding property, the bound on the number of the sender's choices gets translated to saying that the sender's input (in the first phase) comes from a set $T$ of density $2^{-(k-m)}$, so the Interactive Hashing binding property applies. The analyses for the second phase are similar. Formalizing these ideas, we get a new $\binom{2}{1}$-binding commitment scheme in which both phases are hiding with probability at least $2\delta$.

When we try to iterate this amplification step $O(\log n)$ times, we run into a new difficulty. Specifically, the above sketch hides the fact that we pay entropy losses of $\omega(\log n)$ in both the hiding and binding analyses. The entropy loss of $\omega(\log n)$ in the hiding property comes from the Leftover Hash Lemma, in order to ensure that $(h_1, h_1(\sigma_1,\ldots,\sigma_m))$ has negligible statistical distance from uniform. The entropy loss of $\omega(\log n)$ in the binding property comes from needing the $\mu(T)\cdot 2^k$ factor to be negligible when applying Lemma 4.3. This forces us to go, in one step of amplification, from a commitment scheme for secrets of length $k$ to a scheme for secrets of length $k-m-\omega(\log n)$. As in Lemma 4.8, we can take the initial secret length to be $k = \Theta(\log s(n)) = \omega(\log(n))$ if the one-way function has known security $s(n) = n^{\omega(1)}$. But to tolerate $\log n$ losses of $\omega(\log n)$, we would need $s(n) = n^{\omega(\log n)}$ (i.e., at least quasipolynomial security).

To get around this difficulty, in the amplification, we work with more relaxed, "average-case" measures of "entropy" for both the hiding and binding analyses. Specifically, for hiding, we keep track of the expected collision probability of the sender's secret, conditioned on the receiver's view. (Actually, we use the expected square root of the collision probability.) For binding, we work with the expected number of values to which the sender can open. In both cases, we only require these expectations to be within a constant factor of the ideal values ($2^{-k}$ and 1 respectively). With these measures, it turns out that we need only lose $O(m) = O(1)$ bits in the entropy gap with each amplification step. Moreover, once we amplify $\delta$ to a constant, we can afford to take the number of executions $m$ to equal the security parameter $n$ and get an $\Omega(n)$-bit "entropy gap" in the final amplification step. This allows us to achieve exponentially strong statistical hiding even when we do not know the security and start with secret length of $k = O(\log n)$.

The hiding analysis of the above construction works only for certain values of $t$ in the weakly hiding scheme, and for certain values of the $\beta$'s in the amplification steps. We try out all possible values of $t$ and $\beta$'s, thus obtaining a collection poly$(n)$ schemes, at least one of which is strongly hiding and all of which are $\binom{2}{1}$-binding. Notice that the number of possible choices of $t$ and the $\beta$'s are polynomial in $n$ since $t \in \{1, 2, \ldots, n\}$, the $\beta$'s in the each step except for the last is in the range $\{0, 1, \ldots, m-1\}$, for some constant $m$, and the last $\beta$ is in the range $\{0, 1, \ldots, n\}$.

# 6 Weakly Hiding 2-Phase Commitments from One-Way Functions

As discussed in Section 4, for regular one-way functions, Protocol 4.6 has a hiding first phase when the parameter $t$ satisfies $t \lesssim H(f(U_n))$ and has a hiding second phase when $t$ satisfies $t \gtrsim H(f(U_n))$. When $f$ is not regular, then there will be one value of $t \in \{1, 2, \ldots, n\}$ such that $H(f(U_n)) \approx t$ with probability $1/n$. This is the case because there are only $n$ possible choices for the value of $t$.

With this observation in mind, our 2-phase commitment scheme from general one-way functions will be the same as the scheme in Protocol 4.6, with setting of parameters $t = t_0$, $k = O(\log n)$, and $\Delta_1 = \Delta_2 = 2 \log n$, for some $t_0 \in \{1, 2, \ldots, n\}$. In other words, the same scheme—with slightly different setting of parameters—used in the case of regular one-way functions is also applicable to general one-way functions.

This commitment scheme (using general one-way functions), as we will show, is statistically hiding in both phases with probability at least $1/n$ (hence, called *weakly hiding*), and computationally $\binom{2}{1}$-binding. In order to obtain a tighter analysis when we amplify this scheme, we depart from the standard measures of hiding and binding used in Section 4. Instead, we measure the statistical hiding property of our 2-phase commitments using the *expected square root of the collision probability* of the sender's secret, denoted as $\mathrm{CP}^{1/2}$, and defined in Section 6.1 below. We measure the binding property by analyzing the *expected* number of values to which an adversarial sender can open.

Later in Section 7, we show how to boost the statistical hiding probability to $1 - 2^{-\Omega(n)}$ while maintaining the computational $\binom{2}{1}$-binding property.

## 6.1 Properties of Collision Probability

We first define the *collision probability* of a random variable, denoted as CP, and then define the expected square root of the collision probability, denoted as $\mathrm{CP}^{1/2}$. In addition, we state several lemmas about the $\mathrm{CP}^{1/2}$ measure.

**Definition 6.1** (collision probability)**.** For any random variable $A$, we define its *collision probability* as the probability that two independent samples from $A$ are equal. Equivalently,

$$\mathrm{CP}(A) \overset{\mathrm{def}}{=} \sum_{a \in \mathrm{Supp}(A)} (\Pr[A = a])^2 .$$

Measuring the collision probability of a random variable is equivalent to measuring its *Renyi entropy of order 2*, defined as

$$H_2(A) = \log \frac{1}{\mathrm{E}_{a \leftarrow A}\left[\Pr[A = a]\right]} = \log \frac{1}{\mathrm{CP}(A)}.$$

**Definition 6.2** ($\mathrm{CP}^{1/2}$ measure)**.** For any random variable $A$, we define

$$\mathrm{CP}^{1/2}(A) \stackrel{\mathrm{def}}{=} \sqrt{\mathrm{CP}(A)}.$$

For any two (possibly correlated) random variables $A$ and $B$, we define

$$\mathrm{CP}^{1/2}(A|B) \stackrel{\mathrm{def}}{=} \mathop{\mathrm{E}}_{b \leftarrow B}\left[\mathrm{CP}^{1/2}(A|_{B=b})\right].$$

We think of $\mathrm{CP}^{1/2}(A|B) \leq \sqrt{2^k}$ as saying that $A$ has "conditional Renyi entropy" of at least $k$ given $B$. The following lemmas show that $\mathrm{CP}^{1/2}$ behaves nicely as an entropy measure. Proofs are in Appendix A.

**Lemma 6.3.** *For independent pairs of random variables $(X_1, Y_1), \ldots, (X_m, Y_m)$,*

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) = \prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|Y_i).$$

*Note that $X_i$ and $Y_i$ can be correlated, it is only required that the pair $(X_i, Y_i)$ be independent from the other tuples.*

In the language of "conditional Renyi entropy," Lemma 6.3 states that the entropy is additive for independent random variables. We will actually need a generalization of Lemma 6.3 to deal with somewhat dependent random variables, as stated in the next lemma.

**Lemma 6.4.** *Suppose random variables $(X_1, Y_1), \ldots, (X_m, Y_m)$ satisfy the following conditions for some values of $\alpha_1, \ldots, \alpha_m \in \mathbb{R}^+$ and all $i = 1, 2, \ldots, m$:*

1. *For any given $(y_1, \ldots, y_{i-1}) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_{i-1})$,*

$$\mathrm{CP}^{1/2}(X_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}} \mid Y_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}}) \leq \alpha_j.$$

2. *For any given $(y_1, \ldots, y_i) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_i)$, even if we condition on $Y_1 = y_1, \ldots, Y_i = y_i$, the $i+1$ random variables $X_1, X_2, \ldots, X_i, Y_{i+1}$ are independent.*

*Then,*

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) \leq \prod_{i=1}^{m} \alpha_i.$$

**Lemma 6.5.** *Let $(X, Y)$ be any (possibly correlated) pair of random variables, and let $H \leftarrow \mathcal{H}$ be chosen randomly (and independently from $(X, Y)$) from a family of pairwise-independent hash functions with a range of $\{0,1\}^\alpha$. Then,*

$$\mathrm{CP}^{1/2}((H, H(X))|Y) \leq \mathrm{CP}^{1/2}(H) \cdot (\mathrm{CP}^{1/2}(X|Y) + \sqrt{2^{-\alpha}}).$$

We use Lemma 6.5 to show that doing pairwise independent extraction $(h, h(x))$ preserves the $\mathrm{CP}^{1/2}$ measure, as stated in the next lemma.

**Corollary 6.6.** *Let $(X, Y)$ be any (possibly correlated) pair of random variables, and let $H \leftarrow \mathcal{H}$ be chosen randomly (and independently from $(X, Y)$) from a family of pairwise-independent hash functions with a range of $\{0,1\}^\alpha$. Suppose the hash functions from $\mathcal{H}$ are represented by $(q-\alpha)$-bit strings and* $\mathrm{CP}^{1/2}(X|Y) \leq \sqrt{2^{-(\alpha+3)}}$. *Then,*

$$\mathrm{CP}^{1/2}((H, H(X))|Y) \leq \sqrt{2^{-(q-1)}}.$$

In other words, if $X$ has at least $\alpha + 3$ bits of "conditional Renyi entropy" given $Y$, then we can extract $\alpha$ bits from $X$ that have "conditional Renyi entropy" at least $\alpha - 1$. Notice that this entropy loss is only 4 bits, as compared to $2\log(1/\varepsilon)$ if we require that the output be $\varepsilon$-close to uniform as in the Leftover Hash (Lemma 4.1). This constant loss of "conditional Renyi entropy" allows us to do a tighter hiding analysis in Section 7.3.

**Lemma 6.7.** *For any triple of (possibly correlated) random variables $X$, $Y$ and $Z$,*

$$\mathrm{CP}^{1/2}(X|Y) \leq \mathrm{CP}^{1/2}(X|(Y,Z)) \leq \sqrt{|\mathrm{Supp}(Z)|} \cdot \mathrm{CP}^{1/2}(X|Y).$$

This says that conditioning on random variable $Z$ can only decrease the "conditional Renyi entropy," but does so by at most $\log(|\mathrm{Supp}(Z)|)$ bits. The final lemma is a stronger variant of the Leftover Hash Lemma (Lemma 4.1), with the hypothesis stated in terms of Collision Probability.

**Lemma 6.8** (Leftover Hash Lemma [BBR88, ILL89])**.** *Let $\mathcal{H} = \{h \colon \{0,1\}^n \to \{0,1\}^\alpha\}$ be a family of pairwise-independent hash functions, and let $q - \alpha$ be the description of length of each element in $\mathcal{H}$. If* $\mathrm{CP}(X) \leq \varepsilon^2 \cdot 2^{-\alpha}$*, then* $\Delta((H, H(X)), U_q) \leq \varepsilon$.

## 6.2 Average-Case Hiding and Binding Properties of Interactive Hashing

We now analyze the Interactive Hashing Scheme (Protocol 4.5) in terms of "average-case" measures. For hiding, we use the $\mathrm{CP}^{1/2}$ measure introduced in the previous section. For the binding property, we present an average-case version of Lemma 4.3, where we look at the *expected* number of outputs that lies in any set $T$ (rather than bound the probability that there is more than one output in $T$).

**Lemma 6.9** (hiding in $\mathrm{CP}^{1/2}$ measure)**.** *Let $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ be the Interactive Hashing Scheme (Protocol 4.5). If the sender $S_{\mathrm{IH}}$'s input comes from a distribution $Y$ over $\{0,1\}^q$ and $W$ is any (possibly correlated) distribution (representing the receiver's a priori information about $Y$), then for any receiver $R^*$,*

$$\mathrm{CP}^{1/2}(Z|(W,V)) \leq \sqrt{2^{q-k}} \cdot \mathrm{CP}^{1/2}(Y|W),$$

*where* $Z = \mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Y), R^*)(1^q, 1^k)$ *and* $V = \mathsf{view}_{R^*}(S_{\mathrm{IH}}(Y), R^*)(1^q, 1^k)$.

*Proof.* Without loss of generality, we may assume that $R^*$ is deterministic. (The randomized case then follows by taking expectation over $R^*$'s coin tosses.) Now that since $R^*$ is deterministic, the hash functions sent $h_0, \ldots, h_{q-k-1}$ are fully determined by $S_{\mathrm{IH}}$'s responses $c_0, \ldots, c_{q-k-1} \in \{0,1\}$ (refer to Protocol 4.5). Hence, the number of possible different receiver's view is bounded by $2^{q-k}$. This implies that $|\mathrm{Supp}(V)| \leq 2^{q-k}$, where $V = \mathsf{view}_{R^*}(S_{\mathrm{IH}}(Y), R^*)(1^q, 1^k)$. By Lemma 6.7,

$$\mathrm{CP}^{1/2}(Y|(W,V)) \leq \sqrt{|\mathrm{Supp}(V)|} \cdot \mathrm{CP}^{1/2}(Y|W) \leq \sqrt{2^{q-k}} \cdot \mathrm{CP}^{1/2}(Y|W).$$

Observe that given any particular instantiation of $W = w$ and $V = v$, the distributions $\mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Y), R_{\mathrm{IH}})(1^q, 1^k)|_{W=w,V=v}$ has the same collision probability with $Y|_{W=w,V=v}$ (indeed they are in bijective correspondence). Hence, $\mathrm{CP}^{1/2}(Z|(W,V)) = \mathrm{CP}^{1/2}(Y|(W,V)) \leq \sqrt{2^{q-k}} \cdot \mathrm{CP}^{1/2}(Y|W)$. □

**Lemma 6.10** (binding in expected measure)**.** *Let $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$ be the Interactive Hashing Scheme (Protocol 4.5). For any fixed subset $T \subseteq \{0,1\}^q$, and for any sender $S^*$, setting $C = \mathsf{output}((S^*, R_{\mathrm{IH}})(1^q, 1^k))$, we have*

$$\mathrm{E}\left[|\{z : C(z) \in T\}|\right] < \max\{24, 2^{k+1} \cdot \mu(T)\} \leq 24 + 2^{k+1} \cdot \mu(T)$$

*where the above expectation is taken over the coin tosses of $S^*$ and $R_{\mathrm{IH}}$.*

This lemma and its proof are inspired by the work of Goldriech, Goldwasser, and Linial [GGL98], who studied a protocol similar to interactive hashing for a different purpose (namely, random selection protocols).

*Proof.* Without loss of generality, we may assume that $R^*$ is deterministic. (Else, we can fix its coin tosses to maximize the expectation.) Note that for iteration $j = 0, \ldots, q-k-1$, $R_{\mathrm{IH}}$ will send a random $h_j$, partitioning the set of possible outputs into two sets $\{y : h_j(y) = 0\}$ and $\{y : h_j(y) = 1\}$, and $S^*$ chooses a side of the partition by sending a bit $c_j$. Let $T_0 = T$, and for all $j > 0$, $T_j = \{y \in T : h_i(y) = c_i \forall i < j\}$ denote the set of compatible elements at iteration $j$. Let $\mu_j = \mathrm{E}[|T_j| \cdot 2^{-(q-j)}]$, where the expectation is taken over random choices of $h_0, \ldots, h_{j-1}$. For convenience of notation, assume that the hash function $h_i$'s range is $\{\pm 1\}$, instead of $\{0,1\}$.

Consider a particular set $T_j$, and a particular hash function $h_j$. Observe that for every $y \neq y' \in T_j$, $\Pr_{h_j}[h_j(y) = h_j(y')] \leq 1/2$. Hence,

$$\mathop{\mathrm{E}}_{h_j}[h_j(y)h_j(y')] \leq 0. \tag{1}$$

Observe that the set $T_{j+1} = \{y \in T_j : h_j(y) = c_j\}$. Therefore,

$$\begin{aligned}
\mathop{\mathrm{E}}_{h_j}[\mu(T_{j+1})] &= \mu(T_j) + 2^{-(q-j)} \cdot \mathop{\mathrm{E}}_{h_j}\left[\left|\sum_{y \in T_j} h_j(y)\right|\right] \\
&\leq \mu(T_j) + 2^{-(q-j)} \cdot \sqrt{\mathop{\mathrm{E}}_{h_j}\left[\left(\sum_{y \in T_j} h_j(y)\right)^2\right]} && \text{(Cauchy-Schwartz/Jensen)} \\
&= \mu(T_j) + 2^{-(q-j)} \cdot \sqrt{|T_j| + \sum_{y \neq y'} \mathop{\mathrm{E}}_{h_j}[h_j(y)h_j(y')]} \\
&\leq \mu(T_j) + 2^{-(q-j)} \cdot \sqrt{|T_j|} && \text{(by (1))} \\
&= \mu(T_j) + \sqrt{2^{-(q-j)} \cdot \mu(T_j)}.
\end{aligned}$$

Consequently,

$$\begin{aligned}
\mu_{j+1} &= \mathop{\mathrm{E}}_{h_0,\ldots,h_j}[\mu(T_{j+1})] \\
&= \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}[\mathop{\mathrm{E}}_{h_j}[\mu(T_{j+1})]] \\
&\leq \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}\left[\mu(T_j) + \sqrt{2^{-(q-j)} \cdot \mu(T_j)}\right] \\
&\leq \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}[\mu(T_j)] + \sqrt{2^{-(q-j)} \cdot \mathop{\mathrm{E}}_{h_0,\ldots,h_{j-1}}[\mu(T_j)]} \qquad \text{(Cauchy-Schwartz/Jensen)} \\
&= \mu_j + \sqrt{2^{-(q-j)} \cdot \mu_j}.
\end{aligned}$$

Assume that the $\mu_j$'s are monotonically increasing (otherwise, we can make it so). This gives us

$$\begin{aligned}
\mu_{q-k} &\leq \mu_0 + \sum_{j=0}^{q-k-1} \sqrt{2^{-(q-j)} \cdot \mu_j} \\
&\leq \mu_0 + \sqrt{\mu_{q-k}} \cdot \sum_{j=0}^{q-k-1} \sqrt{2^{-(q-j)}} \qquad (\mu_j\text{'s are monotonically increasing}) \\
&< \mu_0 + \sqrt{\mu_{q-k}} \cdot \sqrt{6/2^k} \\
&\leq \mu_0 + \frac{\mu_{q-k}}{2} \qquad (\text{if } \mu_{q-k} \geq 24 \cdot 2^{-k}),
\end{aligned}$$

giving us $\mu_{q-k} < 2\mu_0 = 2\mu(T)$ if $\mu_{q-k} \geq 24 \cdot 2^{-k}$. Therefore, we can conclude that

$$\mathrm{E}[|\{z : C(z) \in T\}| : C = \mathsf{output}((S^*, R_{\mathrm{IH}})(1^q, 1^k))] = \mu_{q-k} \cdot 2^k < \max\{2 \cdot \mu(T) \cdot 2^k, 24\}.$$

□

### 6.3 Hiding Property

Recall that our 2-phase commitment scheme $(S, R)$ from general one-way functions is Protocol 4.6, with setting of parameters $t = t_0$, $k = O(\log n)$, and $\Delta_1 = \Delta_2 = 2\log n$, for some $t_0 \in \{1, 2, \ldots, n\}$. We wish to analyze the collision probability of the sender's private output, both in the first and second phases, when interacting with an adversarial receiver $R^*$. The collision probability measure will be $\mathrm{CP}^{1/2}$, as defined in Section 6.1.

When the sender's input is $x$, let random variable $\mathsf{view}_{R^*}(S_c^1(x), R^*)$ denote the view of receiver $R^*$ in the first commit phase, let random variable $\mathsf{output}_S(S^1(x), R^*)$ denote the sender's private output in the first phase, and let random variable $\mathsf{transcript}(S^1(x), R^*)$ denote the first (commit and reveal) phase transcript. Using similar notations, for transcript $\tau$ and sender's input $x$, let random variable $\mathsf{view}_{R^*}(S_c^2(x), R^*)(\tau)$ denote the view of receiver $R^*$ in the second commit phase, let random variable $\mathsf{output}_S(S^2(x), R^*)(\tau)$ denote the sender's private output in the second phase, and let random variable $\mathsf{transcript}(S^2(x), R^*)(\tau)$ denote the second (commit and reveal) phase transcript.

We prove that for a specific value of $t$, the above 2-phase commitment scheme is weakly hiding ($\delta = 1/n$) in both phases, then prove that is it also $\binom{2}{1}$-computationally binding (see Lemma 6.12).

**Lemma 6.11.** *Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be any function (not necessarily one-way). There exist an integer $t \in [1,n]$ and two sets $T_1, T_2 \subseteq \{0,1\}^n$ such that for every $k \le q(n)$, $\Delta_1 \ge \log n + 4$, and $\Delta_2 \ge 3$, the following properties hold for 2-phase commitment scheme $(S,R)$ in Protocol 4.6:*

*(H.1)* $T_1 \cup T_2 = \{0,1\}^n$ *and* $\mu(T_1 \cap T_2) \ge 1/n$.

*(H.2) When the sender's initial coin tosses $x$ are chosen uniformly from $T_1$, the sender's private output in the first phase has low collision probability. Formally, for any adversarial receiver $R^*$,*

$$\mathrm{CP}^{1/2}(A|V) \le \sqrt{2^{-(k-1)}},$$

*where $A = \mathsf{output}_S(S_c^1(T_1), R^*)$ and $V = \mathsf{view}_{R^*}(S_c^1(T_1), R^*)$, where by abuse of notation we write $T_1$ to denote the uniform distribution on $T_1$.*

*(H.3) When the sender's coin tosses are in $T_2$, the sender's private output in the second phase has low collision probability even when given the first phase transcript. Formally, for every adversarial receiver $R^*$ and every $\tau \in \mathrm{Supp}(\Lambda)$, where $\Lambda = \mathsf{transcript}(S^1(T_2), R^*)$, we have*

$$\mathrm{CP}^{1/2}(B_\tau|W_\tau) \le \sqrt{2^{-(k-1)}},$$

*where $(B_\tau, W_\tau) = (\mathsf{output}_S(S_c^2(T_2), R^*), \mathsf{view}_{R^*}(S_c^2(T_2), R^*))|_{\Lambda=\tau}$.*

*Proof.* Without loss of generality, we may assume that $R^*$ is deterministic since we can fix the coin tosses of $R^*$ that maximizes the collision probability. We prove that $(S,R)$ satisfies the above three properties as follows:

**Property (H.1).** Define $p(y) = \Pr_{x \leftarrow U_n}[f(x) = y]$, and for $t \in \{1,\ldots,n\}$, define $A_t = \{y \in \{0,1\}^n : 2^{-t} \le p(y) < 2^{-t+1}\}$. Since $\cup_t A_t = f(\{0,1\}^n)$, there exists a $\hat{t}$ such that $\Pr[f(U_n) \in A_{\hat{t}}] \ge 1/n$. For the rest of the proof, we set $t = \hat{t}$.

Define sets $T_1$ and $T_2$ as follows:

$$\begin{aligned} T_1 &= \{x : p(f(x)) < 2^{-t+1}\} \\ T_2 &= \{x : p(f(x)) \ge 2^{-t}\} \end{aligned}$$

By the definition of $T_1$ and $T_2$, we have that $\mu(T_1 \cap T_2) = \Pr[f(U_n) \in A_t] \ge 1/n$, and $T_1 \cup T_2 = \{0,1\}^n$.

**Property (H.2).** In the case when the sender's coin tosses is in $T_1$, we analyze the collision probability of the distribution of the first phase secret as follows.

$$\begin{aligned} \mathrm{CP}(f(T_1)) &= \frac{\sum_{y:p(y)<2^{-t+1}} p(y)^2}{\mu(T_1)^2} \\ &\le \left(\max_{y:p(y)<2^{-t+1}} p(y)\right) \cdot \left(\sum_{y:p(y)<2^{-t}} p(y)\right) \cdot \frac{1}{\mu(T_1)^2} \\ &< 2^{-t+1} \cdot \mu(T_1) \cdot \mu(T_1)^{-2} \\ &\le 2^{-(t-\log n-1)} && \text{(since } \mu(T_1) \ge 1/n) \end{aligned}$$

Observe that $\mathrm{CP}(f(T_1)) \leq 2^{-(t-\log n-1)} \leq 2^{-(t-\Delta_1+3)}$. Hence, by Corollary 6.6, letting $Q = (\mathcal{H}_1, \mathcal{H}_1(f(T_1)))$, we have that $\mathrm{CP}^{1/2}(Q) \leq \sqrt{2^{-(q-1)}}$. By Lemma 6.9, letting $A \overset{\mathrm{def}}{=} \mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*) = \mathsf{output}_S(S^1(T_1), R^*)$ and $V \overset{\mathrm{def}}{=} \mathsf{view}_{R^*}(S_{\mathrm{IH}}(Q), R^*) = \mathsf{view}_{R^*}(S^1_c(T_1), R^*)$, we have

$$\mathrm{CP}^{1/2}(A|V) \leq \sqrt{2^{q-k}} \cdot \sqrt{\mathrm{CP}(Q)} \leq \sqrt{2^{q-k}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k-1)}}.$$

**Property (H.3).** In the case when the sender's coin tosses is in $T_2$, we analyze the collision probability of the distribution of the second phase secret as follows. First we observe that for any $x, x' \in \{0,1\}^n$ such that $f(x) = f(x')$, the first phase transcripts for both coin tosses $x$ and $x'$ are identical, that is $\mathsf{transcript}(S^1(x), R^*) \equiv \mathsf{transcript}(S^1(x'), R^*)$.

Fix a first phase transcript $\tau \in \mathsf{transcript}(S^1(x), R^*)$ containing value $y$ in the reveal phase. Observe that the subset $T_{2,y} = f^{-1}(y) \subseteq T_2$ is such that any element in $T_{2,y}$ is equally likely to have generated $\tau$. Note that the $T_{2,y}$'s form a partition of $T_2$.

Note that $|T_{2,y}| \geq 2^{n-t}$ by the definition of $T_{2,y}$ and $T_2$, and hence $\mathrm{CP}(T_{2,y}) \leq 2^{-(n-t)} \leq 2^{-(n-t-\Delta_2+3)}$. By Corollary 6.6, letting $Q' = (\mathcal{H}_2, \mathcal{H}_2(T_{2,y}))$, we have $\mathrm{CP}(Q') \leq 2^{-(q-1)}$. Observing that $B_\tau = \mathsf{output}_S(S^2(T_{2,y}), R^*)(\tau) = \mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q'), R^*)$ and $W_\tau = \mathsf{view}_{R^*}(S^2_c(T_{2,y}), R^*)(\tau) = \mathsf{view}_{R^*}(S_{\mathrm{IH}}(Q), R^*)$, we can apply Lemma 6.9 to deduce that

$$\mathrm{CP}^{1/2}(B_\tau|W_\tau) \leq \sqrt{2^{q-k}} \cdot \sqrt{\mathrm{CP}(Q')} \leq \sqrt{2^{q-k}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k-1)}}.$$

Our proof is complete. □

## 6.4 Binding Property

The definition of $\binom{2}{1}$-binding in Definition 2.6 considers the first phase (resp., second phase) to be broken if the sender $S^*$ produces valid decommitments to *two* different values after the first commit stage (resp., second commit stage). In this section and the next one, we will work with a relaxed notion where we simply bound the *expected* number of values to which the sender can open. To this end, we define $\mathsf{openings}(S^*, R^i)$ to be a random variable denoting the number of values to which the sender successfully opens in phase $i$, where 'successfully' opens is defined for each phase analogously to Definition 2.6. More formally, for a two-phase commitment scheme $(S, R)$ and a 'binding' set $\mathcal{B}$, we define $\mathsf{openings}(S^*, R^1)(\mathcal{B})$ as follows:

- $S^*$ and $R^1_c$ interact to get first phase commitment $z^{(1)}$.
- After the interaction, $S^*$ outputs a sequence of values $d_1, \ldots, d_\ell$ and corresponding full transcripts $\tau_1, \ldots, \tau_\ell$ of both phases.
- We let $\mathsf{openings}(S^*, R^1)(\mathcal{B})$ be the set of distinct values $d_i$ whose opening $\tau_i$ is valid, where by valid we mean that $\tau_i$ begins with prefix $z^{(1)}$, $\tau_i$ contains an opening of $z^{(1)}$ to the value $d_1$ with a first-phase transcript not in $\mathcal{B}$, and both $R^1_r$ and $R^2_r$ accept in $\tau_i$.

Analogously, we define $\mathsf{openings}(S^*, R^2)(\tau^{(1)})$, where $\tau^{(1)}$ is a first phase transcript, as follows:

- $S^*$ and $R^2_c$ interact to get second phase commitment $z^{(2)}$.
- After the interaction, $S^*$ outputs a sequence of values $d_1, \ldots, d_\ell$ and corresponding second-phase transcripts $\tau^{(2)}_1, \ldots, \tau^{(2)}_\ell$.

- We let $\mathsf{openings}(S^*, R^2)(\tau^{(1)})$ be the set of distinct values $d_i$ whose opening $\tau_i^{(2)}$ is valid, where by valid we mean that $\tau_i^{(2)}$ starts with prefix $z^{(2)}$, $\tau_i^{(2)}$ contains an opening of $z^{(2)}$ to the value $d_i$, and $R_r^2$ accepts in $\tau_i^{(2)}$.

Now, we can describe the binding property of Protocol 4.6 in this language (even when the underlying one-way function has unknown hardness).

**Lemma 6.12.** *Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be any one-way function. For every integers $t \in [1, n]$, $k = O(\log n)$, $\Delta_1 = O(\log n)$, and $\Delta_2 = O(\log n)$, the following properties hold for the 2-phase commitment scheme $(S, R)$ in Protocol 4.6 using $f$:*

*There exists a binding set $\mathcal{B}$ for $(S, R)$ where:*

*(B.1) No nonuniform PPT adversary $S^*$ can break the first phase binding with nonnegligible probability in the sense of Definition 2.6. That is, for any nonuniform PPT $S^*$, we have $|\mathsf{openings}(S^*, R^1)(\mathcal{B})| \leq 1$ with probability $1 - \mathrm{neg}(n)$ over the coins of $S^*$ and $R_c^1$.*

*(B.2) For all $\tau \in \mathcal{B}$ and any adversarial sender $S^*$,*

$$\mathrm{E}[|\mathsf{openings}(S^*, R^2)(\tau)|] < 32,$$

*where the above expectation is taken over the coin tosses of $S^*$ and $R^2$.*

*Proof.* We follow the proof of the binding property in Lemma 4.8, using both Claims 4.9 and 4.10 from that proof. Let $\mathcal{B} = \{y \in \{0,1\}^n : \Pr_{U_n}[f(U_n) = y] \leq 2^{-t-\Delta_3}\}$ be the same binding set as defined in both claims. We set $\Delta_3 = \Delta_2 + O(\log n)$ to be large enough so that the binding parameter $\mathrm{poly}(n) \cdot 2^{-\Omega(\Delta_3 - \Delta_2)}$ in Claim 4.9 is at most $2^{-k}$. (This can be done since $k = O(\log n)$.) Now, Claim 4.9 states that if $\tau \in \mathcal{B}$, then the second commitment phase is *not* binding (i.e., $|\mathsf{output}(S^*, R^2)(\tau)| \geq 2$) with probability at most $2^{-k}$. Since $|\mathsf{output}(S^*, R^2)(\tau)| \leq 2^k$ (the commitment is to a $k$-bit string), taking expectations we have:

$$\mathrm{E}[|\mathsf{output}(S^*, R^2)(\tau)|] \leq 2^{-k} \cdot 2^k + (1 - 2^{-k}) \cdot 1 < 2 < 32.$$

To see why property (B.1) holds, observe that the inversion success probability in Claim 4.10 is

$$\varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_3)} = \varepsilon^{O(1)} \cdot 1/\mathrm{poly}(n) \cdot 2^{-(k+\Delta_1+\Delta_2+O(\log n))} = \frac{\varepsilon^{O(1)}}{\mathrm{poly}(n)},$$

since all $k, \Delta_1, \Delta_2 = O(\log n)$. That probability is nonnegligible if $\varepsilon$ is nonnegligible. □

# 7 Converting Weakly Hiding 2-Phase Commitment Schemes into Strongly Hiding Schemes

In the previous section, we constructed a 2-phase commitment scheme that is weakly statistically hiding ($\delta = 1/n$) and $\binom{2}{1}$-computationally binding. In this section, we show how to boost the hiding probability to $\delta = 1 - \mathrm{neg}(n)$ (*strongly hiding*), while maintaining the $\binom{2}{1}$-computationally binding property.

We first show how to double the hiding probability by combining a constant number of schemes to obtain a new scheme. We repeat this doubling amplification process $O(\log n)$ times to get the hiding probability from $1/n$ to a constant $c > 0$. Now when the hiding probability is a constant, we can boost it all the way to $1 - \mathrm{neg}(n)$ by combining $\mathrm{poly}(n)$ number of schemes (that have constant hiding probability).

## 7.1 One Step Amplification Procedure

In Protocol 7.1, we present a hiding amplification procedure $\mathsf{Amplify}$ for 2-phase commitments that takes as input scheme $(S, R)$ and outputs a new scheme $(\mathcal{S}, \mathcal{R})$. The parameters for $\mathsf{Amplify}$, all given in unary, are listed below:

1. Security parameter $n$.
2. Number $m$ of schemes $(S, R)$ to be combined.
3. Integer $r$ denoting $S$'s private input length.
4. Integer $k$ denoting $S$'s private output length.
5. Integer $k'$ denoting $\mathcal{S}$'s private output length.
6. Integral thresholds $\alpha_1$ and $\alpha_2$, for the first and second commit phases respectively.

Hence, we write $(\mathcal{S}, \mathcal{R}) = \mathsf{Amplify}((S, R); 1^n, 1^m, 1^r, 1^k, 1^{k'}, 1^{\alpha_1}, 1^{\alpha_2})$, whose protocol is presented in the next page. To simplify notation, we also write $(\mathcal{S}, \mathcal{R}) = \mathsf{Amplify}((S, R))$ when the parameters are clear from context.

**Protocol 7.1.** Scheme $(\mathcal{S}, \mathcal{R})$ from Hiding Amplification of $(S, R)$.

**Sender's private input:** $x = (x_1, \ldots, x_m) \in \{0,1\}^{mr}$.

**First phase commit:**

1. $(\mathcal{S}_c^1, \mathcal{R}_c^1)$ does $m$ sequential executions of $(S_c^1, R_c^1)$, using $x_i$ for $S_c^1$'s secret in the $i$-th execution. Let $(S_{c,i}^1(x_i), R_{c,i}^1)$ denote the $i$-th execution of $(S_c^1, R_c^1)$. Define $a_i = \mathsf{output}_S(S_{c,i}^1(x_i), R_{c,i}^1) \in \{0,1\}^k$, and let $a = (a_1, \ldots, a_m)$.
2. Let $\mathcal{H}_1 = \{h_1 \colon \{0,1\}^{mk} \to \{0,1\}^{\alpha_1}\}$ be a family of pairwise independent hash functions. $\mathcal{S}^1$ chooses a random hash $h_1 \leftarrow \mathcal{H}_1$, and computes $y^{(1)} = (h_1, h_1(a)) \in \{0,1\}^q$.
3. $(\mathcal{S}_c^1, \mathcal{R}_c^1)$ runs Interactive Hashing Scheme (Protocol 4.5) $(S_{\mathrm{IH}}^1(y^{(1)}), R_{\mathrm{IH}}^1)(1^q, 1^k)$, with $\mathcal{S}^1$ and $\mathcal{R}^1$ acting as $S_{\mathrm{IH}}^1$ and $R_{\mathrm{IH}}^1$, respectively.

   Let circuit $C \colon \{0,1\}^{k'} \to \{0,1\}^q$ be the common output, and $d^{(1)} \in \{0,1\}^{k'}$ be $S_{\mathrm{IH}}^1$'s private output in $(S_{\mathrm{IH}}^1(y^{(1)}), R_{\mathrm{IH}}^1)(1^q, 1^k)$.

*First phase sender's private output:* String $d^{(1)} \in \{0,1\}^{k'}$.

**First phase reveal:**

$S_r^1$ sends tuple $\gamma^{(1)} = (d^{(1)}, a, h_1) \circ (\gamma_1^{(1)}, \ldots, \gamma_m^{(1)})$, where $\gamma_i^{(1)}$ is the first phase revelation string of $S_{r,i}^1$ in the above execution of $(S_{r,i}^1(x_i), R_{r,i}^1)$.

Receiver $\mathcal{R}_r^1$ accepts if only if $C(d^{(1)}) = (h_1, h_1(a))$ and $R_{r,i}^1$ accepts $(\gamma_i^{(1)}, a_i)$ for all $i$.

**Second phase commit:**

*Second phase common input:* Transcript $\tau = (\tau_1, \ldots, \tau_m)$, where $\tau_i = \mathsf{transcript}(S_i^1(x_i), R_i^1)$.

1. $(\mathcal{S}_c^2, \mathcal{R}_c^2)$ does $m$ sequential executions of $(S_c^2, R_c^2)$, using $x_i$ for $S^2$'s secret and transcript $\tau_i$ in the $i$-th execution. Let $(S_{c,i}^2(x_i), R_{c,i}^2)(\tau_i)$ denote the $i$-th execution of $(S^2, R^2)$. Define $b_i = \mathsf{output}_S(S_{c,i}^2(x_i), R_{c,i}^2)(\tau_i) \in \{0,1\}^k$, and let $b = (b_1, \ldots, b_m)$.
2. Let $\mathcal{H}_2 = \{h_2 \colon \{0,1\}^{mk} \to \{0,1\}^{\alpha_2}\}$ be a family of pairwise independent hash functions. $\mathcal{S}^2$ chooses a random hash $h_2 \leftarrow \mathcal{H}_2$, and computes $y^{(2)} = (h_2, h_2(b)) \in \{0,1\}^q$.
3. $(\mathcal{S}_c^2, \mathcal{R}_c^2)$ runs Interactive Hashing Scheme (Protocol 4.5) $(S_{\mathrm{IH}}^2(y^{(2)}), R_{\mathrm{IH}}^2)(1^q, 1^k)$, with $\mathcal{S}_c^2$ and $\mathcal{R}_c^2$ acting as $S_{\mathrm{IH}}^2$ and $R_{\mathrm{IH}}^2$, respectively.

   Let circuit $C \colon \{0,1\}^{k'} \to \{0,1\}^q$ be the common output, and $d^{(2)} \in \{0,1\}^{k'}$ be $S_{\mathrm{IH}}^2$'s private output in $(S_{\mathrm{IH}}^2(y^{(2)}), R_{\mathrm{IH}}^2)(1^q, 1^k)$.

*Second phase sender's private output:* String $d^{(2)} \in \{0,1\}^{k'}$.

**Second phase reveal:**

$S_r^2$ sends tuple $\gamma^{(2)} = (d^{(2)}, b, h_2) \circ (\gamma_1^{(2)}, \ldots, \gamma_m^{(2)})$, where $\gamma_i^{(2)}$ is the second phase revelation string of $S_{r,i}^2$ in the above execution of $(S_{r,i}^2(x_i), R_{r,i}^2)$.

Receiver $\mathcal{R}^2$ accepts if only if $C^{(2)}(d^{(2)}) = (h_2, h_2(b))$ and $R_{r,i}^2$ accepts $(\gamma_i^{(2)}, b_i)$ for all $i$.

## 7.2 Iterative Amplification Procedure

We start off with a weakly-hiding 2-phase commitment scheme based on one-way function (cf., Section 6), denoted by $(S_0, R_0)$. We get a new scheme $(\mathsf{S}, \mathsf{R})$ by iteratively applying the amplification process $\mathsf{Amplify}$, as described in Algorithm 7.2 below. Let $D > 1$ denote a large enough integer constant. (We set $m = D$ in all but the last iteration.)

**Algorithm 7.2.** Iterative Amplification Procedure.

**Input:** Security parameter $n$, constant integer $D > 1$, and thresholds $t \in \{1, 2, \ldots, n\}$, $\beta_1, \ldots, \beta_\ell \in \{0, 1, \ldots, D-1\}$, $\beta_{\ell+1} \in \{0, 1, \ldots, n\}$.

1. Let $k_0 = (16D) \cdot \log n$, $\ell = \log n$, and $(S_0, R_0)$ be the 2-phase commitment scheme based on one-way function $f \colon \{0,1\}^n \to \{0,1\}^n$ from Protocol 4.6 using parameters $t$, $k = k_0$, and $\Delta_1 = \Delta_2 = 2 \log n$.
2. For $j = 1, 2, \ldots, \ell$, repeat the following:
    (a) Set $k_j = k_{j-1} - 8D - 8$.
    (b) Set $(S_j, R_j) = \mathsf{Amplify}((S_{j-1}, R_{j-1}))$ for settings of parameters $m = D$, $r = n \cdot D^{j-1}$, $k = k_{j-1}$, $k' = k_j$, $\alpha_1 = (\beta_j + 1)(k_{j-1} - 1) - 3$ and $\alpha_2 = (D - \beta_j)(k_{j-1} - 1) - 3$.
3. Set $(\mathsf{S}, \mathsf{R}) = \mathsf{Amplify}((S_\ell, R_\ell))$ for settings of parameters $m = n$, $r = n \cdot D^\ell$, $k = k_\ell$, $k' = 1$, $\alpha_1 = \lfloor (\beta_{\ell+1} + \frac{1}{3}\delta n)k \rfloor$ and $\alpha_2 = \lfloor (n - \beta_{\ell+1} + \frac{1}{3}\delta n)k \rfloor$, where $\delta = 1/(2D)$.

**Output:** 2-phase commitment scheme $(\mathsf{S}, \mathsf{R})$.

**Lemma 7.3.** *If scheme $(S_0, R_0)$ used by Algorithm 7.2 runs in probabilistic polynomial time, then scheme $(\mathsf{S}, \mathsf{R})$, the output of Algorithm 7.2, also runs in probabilistic polynomial time.*

*Proof.* Scheme $(\mathsf{S}, \mathsf{R})$ consists of $n \cdot D^\ell = n \cdot D^{O(\log n)} = \mathrm{poly}(n)$ executions of $(S_0, R_0)$. In addition, each amplification procedure $\mathsf{Amplify}$ adds an overhead time of $\mathrm{poly}(n)$ since both the sender and receiver are doing Interactive Hashing. Since there are only $1 + n + nD + nD^2 + \cdots + D^{\ell-1} = \mathrm{poly}(n)$ amplifications steps, the overhead time is polynomial. Hence, scheme $(\mathsf{S}, \mathsf{R})$ runs in probabilistic polynomial time if $(S_0, R_0)$ does too. □

## 7.3 Hiding Amplification

The following two lemmas, Lemma 7.4 and 7.5, provide us a way to understand the hiding property (in the $\mathrm{CP}^{1/2}$ measure) of $(\mathcal{S}, \mathcal{R})$, the amplified hiding scheme as presented in Protocol 7.1, in terms of $(S, R)$.

**Lemma 7.4** (intermediate step hiding amplification)**.** *For a sufficiently large constant $D \in \mathbb{Z}$,*

*If there exists a 2-phase commitment scheme $(S, R)$ having two associated subsets $T_1, T_2 \subseteq \{0,1\}^r$ such that the following holds for every adversarial receiver $R^*$:*

*(H.1)* $\mu(T_1 \cap T_2) \overset{\text{def}}{=} \delta$ *and* $T_1 \cup T_2 = \{0,1\}^r$.

*(H.2)* $\mathrm{CP}^{1/2}(A|V) \leq \sqrt{2^{-(k-1)}}$, *where* $A = \mathsf{output}_S(S_c^1(T_1), R^*)$ *and* $V = \mathsf{view}_{R^*}(S_c^1(T_1), R^*)$.

*(H.3)* $\mathrm{CP}^{1/2}(B_\tau|W_\tau) \leq \sqrt{2^{-(k-1)}}$, *for every* $\tau \in \mathrm{Supp}(\Lambda)$, *where* $\Lambda = \mathsf{transcript}(S^1(T_2), R^*)$ *and* $(B_\tau, W_\tau) = (\mathsf{output}_S(S_c^2(T_2), R^*), \mathsf{view}_{R^*}(S_c^2(T_2), R^*))|_{\Lambda=\tau}$.

*Then there exist an integer* $\beta \in [0, D-1]$ *such that scheme* $(\mathcal{S}, \mathcal{R}) = \mathsf{Amplify}((S, R))$, *with parameters* $m = D$, $k' = k - 8D - 8$, $\alpha_1 = (\beta+1)(k-1) - 3$, *and* $\alpha_2 = (D - \beta)(k-1) - 3$, *has two associated sets* $T_1', T_2' \subseteq \{0,1\}^{Dr}$ *such that the following holds for every adversarial receiver* $R^*$*:*

*(H'.1)* $\mu(T_1' \cap T_2') \geq \min\{2\delta, 1/D\}$ *and* $T_1' \cup T_2' = \{0,1\}^{Dr}$.

*(H'.2)* $\mathrm{CP}^{1/2}(A'|V') \leq \sqrt{2^{-(k'-1)}}$, *where* $A' = \mathsf{output}_{\mathcal{S}}(\mathcal{S}^1(T_1'), R^*)$ *and* $V' = \mathsf{view}_{R^*}(\mathcal{S}^1(T_1'), R^*)$.

*(H'.3)* $\mathrm{CP}^{1/2}(B'_{\tau'}|W'_{\tau'}) \leq \sqrt{2^{-(k'-1)}}$, *for every* $\tau' \in \mathrm{Supp}(\Lambda')$, *where* $\Lambda' = \mathsf{transcript}(\mathcal{S}^1(T_2'), R^*)$ *and* $(B'_{\tau'}, W'_{\tau'}) = (\mathsf{output}_{\mathcal{S}}(\mathcal{S}_c^2(T_2'), R^*), \mathsf{view}_{R^*}(\mathcal{S}_c^2(T_2'), R^*))|_{\Lambda'=\tau'}$.

*Proof.* Without loss of generality, we may assume that $R^*$ is deterministic since we can fix the coin tosses of $R^*$ that maximizes the collision probability. Throughout this proof, the value of $m$ will be fixed to $D$, although we will keep writing $m$.

**Property (H.1) implies (H'.1).** Define the sets $T_1'$ and $T_2'$ as follows (the value of $\beta$ will be determined later).

$$\begin{aligned} T_1' &= \{(x_1, \ldots, x_m) : \exists\, i_1, \ldots, i_{\beta+1} \text{ such that } x_{i_1}, \ldots, x_{i_{\beta+1}} \in T_1\}, \\ T_2' &= \{(x_1, \ldots, x_m) : \exists\, i_1, \ldots, i_{m-\beta} \text{ such that } x_{i_1}, \ldots, x_{i_{m-\beta}} \in T_2\}. \end{aligned}$$

By the way we defined $T_1'$ and $T_2'$ together with the fact that $T_1 \cup T_2 = \{0,1\}^r$, we can conclude that $T_1' \cup T_2' = \{0,1\}^{mr}$. In addition, we know that $\mu(T_1 \cap T_2) = \delta$. Choose any subset $S \subseteq T_1 \cap T_2$ such that $\mu(S) = \min\{\delta, 1/(2m)\} \stackrel{\text{def}}{=} \delta'$. Hence, we have

$$\Pr_{x_1,\ldots,x_m \leftarrow \{0,1\}^r}[\text{exactly one } x_i \in S] = m\delta'(1-\delta')^{m-1} \geq m\delta'(1 - 1/(m-1))^{m-1} = \Omega(m\delta').$$

Given that exactly one $x_i \in S$ and wlog assume that $x_m \in S$. Let $p_t$ denote the conditional probability that exactly $t$ of the rest of the $m-1$ $x_i$'s are in $T_1 \setminus T_2$. Choose $\beta \in [0, m-1]$ to maximize $p_t$, i.e., $\beta = \mathrm{argmax}_t\, p_t$. Let $I_i$, for $i = 1, 2, \ldots, m-1$, be a binary random variable indicating whether $x_i \in T_1$ or not; note that these are independent random variables conditioned on the fact that $x_m \in S$. Let the $\mu$ the mean of the $I_i$'s. By a Chernoff bound,

$$\Pr\left[\left|\sum_i I_i - \mu \cdot (m-1)\right| > 3\sqrt{m-1}\right] \leq 2e^{((m-1)/3)\cdot(3/\sqrt{m-1})^2} < 1/2.$$

This means that greater $1/2$ of the weight is centered around $\mu \cdot (m-1) \pm 3\sqrt{m-1}$. Since we chose $\beta = \mathrm{argmax}_t\, p_t$ in a maximal way, we have

$$\Pr_{x_1,\ldots,x_m \leftarrow \{0,1\}^r}[\text{exactly } \beta \text{ of } x_i\text{'s are in } T_1 \setminus S \mid \text{exactly one } x_i \in S] = \Omega\left(\frac{1}{\sqrt{m}}\right).$$

Knowing that $T_1 \cup T_2 = \{0,1\}^r$, if exactly $\beta$ of $x_i$'s in $T_1 \setminus S$ and exactly one $x_i \in S$, then there must be at least $m - 1 - \beta$ of $x_i$'s in $T_2 \setminus S$. Consequently,

$$\Pr_{x_1,\ldots,x_m \leftarrow \{0,1\}^r}[(x_1, \ldots, x_m) \in T_1' \cap T_2'] = \Omega(m\delta') \cdot \Omega\left(\frac{1}{\sqrt{m}}\right) = \Omega(\sqrt{m}\delta') > 2\delta' = \min\{2\delta, 1/m\},$$

for a large enough constant $m = D$.

**Property (H.2) implies (H'.2).** In the commit phase $(\mathcal{S}_c^1, R^*)$, the cheating receiver $R^*$ interacts with $m$ sequential executions of $S_c^1$. Here we must analyze the case that the coin tosses for $S_c^1$ in these $m$ executions are given by $x = (x_1, \ldots, x_m)$ distributed uniformly in $T_1'$. We let $A_i = A_i(x)$ denote the private output of the sender and $V_i = V_i(x)$ the view of the receiver in the $i$'th execution. That is, for $i = 1, \ldots, m$,

$$\begin{aligned} A_i &= \mathsf{output}_S(S_c^1(x_i), R^*(V_1, \ldots, V_{i-1}); \\ V_i &= \mathsf{view}_{R^*}(S_c^1(x_i), R^*(V_1, \ldots, V_{i-1})). \end{aligned}$$

Note that while the sender's behavior in the $i$'th execution is independent of the previous executions, the cheating receiver may base its strategy on its previous views. We want to bound $\mathrm{CP}^{1/2}(A''(X)|V''(X))$, where $A''(X) = (A_1(X), \ldots, A_m(X))$, $V''(X) = (V_1(X), \ldots, V_m(X))$, and $X$ is distributed uniformly in $T_1'$. To do this, we consider, for each $I \subseteq [m]$ of size at least $\beta + 1$, the distribution $X_I$ on coin tosses for the sender, where we choose $x_i$ uniformly in $T_1$ for $i \in I$, and uniformly in $\overline{T_1}$ for $i \notin I$. To get a bound on $\mathrm{CP}^{1/2}(A''(X_I)|V''(X_I))$, we will have to refer to Lemma 6.4 and see why the $(A_i, V_i)$'s satisfy the two conditions of the lemma.

Conditioned on the any previous view, i.e., $V_1(X_I) = v_1, \ldots, V_{i-1}(X_I) = v_{i-1}$ for any $v_1, \ldots, v_{i-1}$, it is the case that $\mathrm{CP}^{1/2}(A_i(X_I)|V_i(X_I)) \leq \sqrt{2^{-(k-1)}}$ if $i \in I$. This follows from Property (H.2) because the (unbounded) receiver $R^*$ can incorporate the previous view $v_1, \ldots, v_{i-1}$ as nonuniform advice, and then the only randomness in the definition of $A_i$ and $V_i$ is the sender's coin tosses $x_i \leftarrow (X_I)_i$, which are uniform in $T_1$ (even conditioned on $v_1, \ldots, v_{i-1}$). This shows that the first condition of Lemma 6.4 is satisfied.

For the second condition, we need to show that conditioned on $V_1(X_I) = v_1, \ldots, V_i(X_I) = v_i$, the random variables $A_1(X_I), \ldots, A_i(X_I), V_{i+1}(X_I)$ are independent. This can be seen by induction on $i$ as follows. It is vacuously true for $i = 0$. Assuming it is true for $i = j - 1$, we prove it for $i = j$ as follows. First condition on $v_1, \ldots, v_{j-1}$. By inductive hypothesis, $A_1, \ldots, A_{j-1}, V_j$ are independent (omitting $X_I$ from the notation for readability). Moreover, since we have conditioned on $v_1, \ldots, v_{j-1}$, $A_j$ and $V_j$ are functions of only $(X_I)_j$, the sender's coin tosses in the $j$'th execution, which is independent of $A_1, \ldots, A_{j-1}$ (because we have only used $(X_I)_1, \ldots, (X_I)_{j-1}$ so far). Thus, if we condition on $V_j = v_j$, $A_j$ remains independent of $A_1, \ldots, A_{j-1}$. $V_{j+1}$ is independent of $A_1, \ldots, A_j$ because now it is only a function of $(X_I)_{j+1}$, which has not been used yet.

Applying Lemma 6.4, we have

$$\mathrm{CP}^{1/2}(A''(X_I)|V''(X_I)) \leq \sqrt{2^{-(\beta+1)(k-1)}}, \tag{2}$$

since from property (H.2), it is the case that for all $i \in I$, $\mathrm{CP}^{1/2}(A_i|V_i) \leq \sqrt{2^{-(k-1)}}$ (even conditioned on the previous views), and $|I| = \beta + 1$.

Now, to bound $\mathrm{CP}^{1/2}(A''(X)|V''(X))$ where $X$ is uniform in $T_1'$, we observe that $X = X_{\mathcal{I}}$, where $\mathcal{I}$ is the distribution on subsets $I$ of size at least $\beta + 1$ given by

$$\Pr[\mathcal{I} = I] = \Pr_{(x_1, \ldots, x_m) \leftarrow T_1'}[\{i : x_i \in T_1\} = I].$$

That is, sampling from $T_1'$ can be broken into two steps; first sampling an $I \leftarrow \mathcal{I}$, and second sampling $x_i \leftarrow T_1$ for $i \in I$ and $x_i \leftarrow \overline{T_1}$ for $i \notin I$. Therefore:

$$\begin{aligned}
\mathrm{CP}^{1/2}(A''(X_{\mathcal{I}})|V''(X_{\mathcal{I}})) &\le \mathrm{CP}^{1/2}(A''(X_{\mathcal{I}})|(V''(X_{\mathcal{I}}),\mathcal{I})) && \text{(by Lemma 6.7)}\\
&= \mathop{\mathrm{E}}_{I \leftarrow \mathcal{I}}\left[\mathrm{CP}^{1/2}(A''(X_I)|V''(X_I)\right]\\
&\le \sqrt{2^{-(\beta+1)(k-1)}} && \text{(by (2))}\\
&= \sqrt{2^{-(\alpha_1+3)}}.
\end{aligned}$$

And by Corollary 6.6, $\mathrm{CP}^{1/2}(H_1, H_1(A''(X))|V''(X)) \le \sqrt{2^{-(q-1)}}$.

Let $Q = (H_1, H_1(A''(X)))$. By Lemma 6.9, letting $A' = \mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*_{\mathrm{IH}}) = \mathsf{output}_{\mathcal{S}}(\mathcal{S}^1(T_1'), R^*)$ and $V' = (\mathsf{view}_{R^*_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*_{\mathrm{IH}}), V'') = \mathsf{view}_{R^*}(\mathcal{S}^1(T_1'), R^*)$, we have

$$\mathrm{CP}^{1/2}(A'|V') \le \sqrt{2^{q-k'}} \cdot \mathrm{CP}^{1/2}(Q|V'') \le \sqrt{2^{q-k'}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k'-1)}},$$

as required.

**Property (H.3) implies (H'.3).** Fix a transcript $\tau' \in \mathrm{Supp}(\Lambda')$, where $\Lambda = \mathsf{transcript}(\mathcal{S}^1(T_2'), R^*)$ $\tau'$ contains first-phase transcripts $(\tau_1, \ldots, \tau_m)$ for the $m$ executions of $(S, R)$. Similarly to the above proof of Property (H'.2), we define random variables

$$\begin{aligned}
B_i(x) &= \mathsf{output}_S(S_c^2(x_i), R^*(W_1, \ldots, W_{i-1})(\tau_i);\\
W_i(x) &= \mathsf{view}_{R^*}(S_c^2(x_i), R^*(W_1, \ldots, W_{i-1})(\tau_i),
\end{aligned}$$

where $x_i$ are the coin tosses of the sender in the $i$'th execution of the the $(S, R)$. For notational simplicity, we omit the sender's coin-tosses from the first-phase interactive hashing (they can be considered fixed for the analysis below). As above, we want to bound $\mathrm{CP}^{1/2}(B''(X_{\tau'})|W''(X_{\tau'}))$, where $B''(X_{\tau'}) = (B_1(X_{\tau'}), \ldots, B_m(X_{\tau'}))$, $W''(X_{\tau'}) = (W_1(X_{\tau'}), \ldots, W_m(X_{\tau'}))$, $X_{\tau'} = X|_{\Lambda'(X)=\tau'}$, and $X$ is distributed uniformly in $T_2'$. To do this, we consider, for each $J \subseteq [m]$ of size at least $m - \beta$, the distribution $X_J$ on coin tosses for the sender, where we choose $x_i$ uniformly in $T_2$ for $i \in J$, and uniformly in $\overline{T_2}$ for $i \notin J$.

It is important to note that even when we condition on $\Lambda'(X) = \tau'$, the components $(X_1, \ldots, X_m)$of $X_J$ remain independent, and the distribution of $X_i|_{\Lambda'(X_J)=\tau'}$ is equivalent to $X_i|_{\Lambda(X_i)=\tau_i}$, where only condition on the transcript of the $i$'th execution. (Similarly to the inductive proof above, it can be shown that $(X_1, \ldots, X_m)$ are independent given the receiver's view $V_m$ of the $m$ executions of $S_c^1$. The only additional information revealed about the $X_i$'s in the first phase is $(A_1, \ldots, A_m)$, where $A_i$ is a function only of $X_i$ once we condition on $V_m$.)

Thus from property (H.3), we have for all $i \in J$, $\mathrm{CP}^{1/2}(B_i(X_{J,\tau'})|W_i(X_{J,\tau'})) \le \sqrt{2^{-(k-1)}}$, where $X_{J,\tau'} = X_J|_{\Lambda'(X_J)=\tau'}$, and this holds even conditioned on the previous views. Similar to the first phase, we apply Lemma 6.4 to show that

$$\mathrm{CP}^{1/2}(B''(X_{J,\tau'})|W''(X_{J,\tau'})) \le \sqrt{2^{-(m-\beta)(k-1)}}. \tag{3}$$

Similarly to above, we observe that $X_{\tau'} = X_{\mathcal{J},\tau'}$ for an appropriate distribution $\mathcal{J}$ on sets of size at least $m - \beta$, and thus

$$\mathrm{CP}^{1/2}(B''(X_{\tau'})|W''(X_{\tau'})) \le \sqrt{2^{-(m-\beta)(k-1)}} = \sqrt{2^{-(\alpha_2+3)}}.$$

By Corollary 6.6, we have $\mathrm{CP}^{1/2}(H_2, H_2(B''(X_{\tau'}))|W''(X_{\tau'})) \leq \sqrt{2^{-(q-1)}}$, which by Lemma 6.9 implies that

$$\mathrm{CP}^{1/2}(B'_{\tau'}|W'_{\tau'}) \leq \sqrt{2^{q-k'}} \cdot \sqrt{2^{-(q-1)}} = \sqrt{2^{-(k'-1)}},$$

as required. □

**Lemma 7.5** (final step hiding amplification)**.** *For every constant $\delta > 0$ and every $k \geq 100/s$, the following holds:*

*If there exists a 2-phase commitment scheme $(S, R)$ having two associated subsets $T_1, T_2 \subseteq \{0,1\}^r$ such that the following holds for every adversarial receiver $R^*$:*

*(H.1)* $\mu(T_1 \cap T_2) = \delta$ *and* $T_1 \cup T_2 = \{0,1\}^r$.

*(H.2)* $\mathrm{CP}^{1/2}(A|V) \leq \sqrt{2^{-(k-1)}}$, *where* $A = \mathsf{output}_S(S_c^1(T_1), R^*)$ *and* $V = \mathsf{view}_{R^*}(S_c^1(T_1), R^*)$.

*(H.3)* $\mathrm{CP}^{1/2}(B_\tau|W_\tau) \leq \sqrt{2^{-(k-1)}}$, *for every* $\tau \in \mathrm{Supp}(\Lambda)$, *where* $\Lambda = \mathsf{transcript}(S^1(T_2), R^*)$ *and* $(B_\tau, W_\tau) = (\mathsf{output}_S(S_c^2(T_2), R^*), \mathsf{view}_{R^*}(S_c^2(T_2), R^*))|_{\Lambda=\tau}$.

*Then there exist an integer $\beta \in [0, n]$ such that scheme $(\mathcal{S}, \mathcal{R}) = \mathsf{Amplify}((S, R))$, with parameters $m = n$, $k' = 1$, $\alpha_1 = \lfloor(\beta + \frac{1}{3}\delta n)k\rfloor$ and $\alpha_2 = \lfloor(n - \beta + \frac{1}{3}\delta n)k\rfloor$, has two associated sets $T'_1, T'_2 \subseteq \{0,1\}^{nr}$ such that the following holds for every adversarial receiver $R^*$:*

*(H'.1) Both* $\mu(T'_1), \mu(T'_2) \geq 1 - 2^{-\Omega(n)}$.

*(H'.2)* $(A', V')$ *is* $2^{-\Omega(n)}$*-close to* $(U_1, V')$, *where* $A' = \mathsf{output}_{\mathcal{S}}(\mathcal{S}_c^1(T'_1), R^*)$ *and* $V' = \mathsf{view}_{R^*}(\mathcal{S}_c^1(T'_1), R^*)$.

*(H'.3)* $(B', W', \Lambda')$ *is* $2^{-\Omega(n)}$*-close to* $(U_1, W', \Lambda')$, *where* $B' = \mathsf{output}_{\mathcal{S}}(\mathcal{S}_c^2(T'_2), R^*)(\Lambda')$ *and* $W' = \mathsf{view}_{R^*}(\mathcal{S}_c^2(T'_2), R^*)(\Lambda')$, *and* $\Lambda' = \mathsf{transcript}(\mathcal{S}^1(T_2), R^*)$.

*Proof.* Throughout this proof, the value of $m$ will be fixed to $n$, although we will keep writing $m$.

**Property (H.1) implies (H'.1).** Let $p = \mu(T_1)$. Set $\beta = \lfloor pn - \frac{1}{2}\delta n\rfloor$, $\gamma_1 = \lfloor pn - \frac{1}{12}\delta n\rfloor$ and $\gamma_2 = \lfloor(1 - p + \delta)n - \frac{1}{12}\delta n\rfloor$. Note that $\beta \in [0, n]$ since $p \in [\delta, 1]$.

Define the sets $T'_1$ and $T'_2$ as follows:

$$\begin{aligned} T'_1 &= \{(x_1, \ldots, x_n) : \exists\, i_1, \ldots, i_{\gamma_1} \text{ such that } x_{i_1}, \ldots, x_{i_{\gamma_1}} \in T_1\}, \\ T'_2 &= \{(x_1, \ldots, x_n) : \exists\, i_1, \ldots, i_{\gamma_2} \text{ such that } x_{i_1}, \ldots, x_{i_{\gamma_2}} \in T_2\}. \end{aligned}$$

To lower bound $\mu(T'_1)$, note that $\mu(T_1) - \gamma_1/n = p - \lfloor pn - \frac{1}{12}\delta n\rfloor/n \geq \frac{1}{12}\delta = \Omega(1)$ since $\delta = \Omega(1)$. Using a Chernoff bound, we get

$$\begin{aligned} \mu(T'_1) &= 1 - \Pr_{(x_1,\ldots,x_n)}[\text{less than } \gamma_1 \text{ of the } x_i\text{'s are in } T_1] \\ &= 1 - 2^{-\Omega(n)}. \end{aligned}$$

To analyze $\mu(T'_2)$, we note that $\mu(T_2) - \gamma_2/n = (1-p+\delta) - \lfloor(1-p+\delta)n - \frac{1}{12}\delta n\rfloor/n \geq \frac{1}{12}\delta = \Omega(1)$. Using a similar analysis as above, we get $\mu(T'_2) = 1 - 2^{-\Omega(n)}$.

**Property (H.2) implies (H'.2).** Using the same analysis as in the proof of Lemma 7.4, we get

$$\mathrm{CP}^{1/2}(A''|V'') \leq \sqrt{2^{-\gamma_1 \cdot (k-1)}},$$

where $A'' = (\mathsf{output}_S(S_1^1(x_1), R^*), \ldots, \mathsf{output}_S(S_n^1(x_n), R^*))$ with $(x_1, \ldots, x_n) \leftarrow T_1'$, and analogously for $V''$. By Markov, we know that with probability greater than $1 - 2^{-n}$ over $v'' \leftarrow V''$, we have

$$\mathrm{CP}(A''|_{V''=v''}) \leq 2^{-\gamma_1(k-1)} \cdot 2^{2n} \leq 2^{-\alpha_1 - (1/24)\delta kn + 3n} \leq 2^{-(\alpha_1 + n)}, \tag{4}$$

since $k \geq 100/\delta$.

Consider $v'' \in V''$ such that the above (4) holds. Let $Q = (H_1, H_1(A''))$ and hence, $Q|_{V''=v''} = (H_1, H_1(A''|_{V''=v''}))$. By Lemma 6.8, we have that $\Delta(Q|_{V''=v''}, U_q) \leq 2^{-\Omega(n)}$. Therefore, by the perfect hiding property of Interactive Hashing (Theorem 4.4 following Definition 4.2), $(V_{\mathrm{IH}}|_{V''=v''}, A'|_{V''=v''})$ is $2^{-\Omega(n)}$-close to $(V_{\mathrm{IH}}|_{V''=v''}, U_1)$, where $V_{\mathrm{IH}} \stackrel{\mathrm{def}}{=} (\mathsf{view}_{R^*_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*_{\mathrm{IH}})$ and $A' \stackrel{\mathrm{def}}{=} \mathsf{output}_{S_{\mathrm{IH}}}(S_{\mathrm{IH}}(Q), R^*_{\mathrm{IH}}) = \mathsf{output}_{\mathcal{S}}(\mathcal{S}^1(T_1'), R^*)$.

Let $V' \stackrel{\mathrm{def}}{=} \mathsf{view}_{R^*}(\mathcal{S}_c^1(T_1'), R^*) = (V'', V_{\mathrm{IH}})$. Since $(V_{\mathrm{IH}}|_{V''=v''}, A'|_{V''=v''})$ is $2^{-\Omega(n)}$-close to $(V_{\mathrm{IH}}|_{V''=v''}, U_1)$ for all but a $2^{-n}$ fraction of $v'' \leftarrow V''$, it follows that $(V', A')$ is $2^{-\Omega(n)}$-close to $(V', U_1)$, as required.

**Property (H.3) implies (H'.3).** Using similar ideas in the proof of Lemma 7.4, we can proceed as above and obtain that Property (H'.3) holds assuming (H.3). □

## 7.4 Binding Preservation

In the execution of Algorithm 7.2, we obtain intermediate commitment schemes $[(S_j, R_j)]_{1 \leq j \leq \ell}$, and final commitment scheme $(\mathsf{S}, \mathsf{R})$. Our goal is to prove that the final scheme $(\mathsf{S}, \mathsf{R})$ satisfies the $\binom{2}{1}$-binding property of Definition 2.6. To achieve our goal, we inductively show that the *expected* number of openings a sender can produce in the intermediate schemes is bounded by some constant, namely 32. (This is captured by Lemma 7.7 below.) Then in the final step, for scheme $(\mathsf{S}, \mathsf{R})$, we show how to shrink this expectation to value that is very close to 1, effectively proving that scheme $(\mathsf{S}, \mathsf{R})$ is satisfies the $\binom{2}{1}$-binding property. (This in turn is captured by Lemma 7.9.)

In the definition of the computational $\binom{2}{1}$-binding property (Definition 2.6), we stipulated that the adversarial sender in the second phase can be computationally unbounded, whereas the adversarial sender in the first phase must be probabilistic polynomial time (PPT). It will be rather messy to work with PPT senders, hence we will first abstract away the PPT requirement by showing, in the next section, how to convert any *polynomial-time* sender violating the $\binom{2}{1}$-binding property in the first phase into a computationally unbounded sender with a special *"unique binding"* property. A sender with the unique binding property, intuitively, will not break the (first-phase) binding property of any execution of the initial schemes $(S_0, R_0)$, but might still break the binding property of the intermediate schemes $(S_j, R_j)$ (or final scheme $(\mathsf{S}, \mathsf{R})$). Intuitively, we can restrict to such senders because of the computational $\binom{2}{1}$-binding property of the initial scheme $(S_0, R_0)$. Once we have a sender with the unique binding property, the analysis of the amplification steps is entirely information theoretic.

To formally define the unique binding property for senders, we observe that schemes $[(S_j, R_j)]_{1 \leq j \leq \ell}$ and $(\mathsf{S}, \mathsf{R})$ each contain multiple executions of initial scheme $(S_0, R_0)$. Hence, when a cheating sender $S^*$ interacts with $R_j$, it is actually also interacting with the $i$-th execution of $R_0$, for each $i = 1, 2, \ldots$, which we will denote by $R_0[i]$. Formally, we obtain a (computationally unbounded) cheating sender strategy $S^*[i]$ that interacts with this single execution of $R_0[i]$ (more precisely, the first commit stage $R^1_{0,c}[i]$), by simulating all of the other messages of $R_j$ on its own until the end of the first commit stage of $R_0[i]$. Then it enumerates over all choices for the subsequent messages of $R_j$ and outputs all of the resulting transcripts of $S^*$'s interactions with $R_0[i]$ together with the corresponding first-phase decommitment values.

**Definition 7.6** (unique binding sender)**.** For intermediate schemes $[(S_j, R_j)]_{1\leq j\leq \ell}$ and final scheme $(\mathsf{S}, \mathsf{R})$, we say that a (deterministic) sender $S^*$ has the *unique binding* property if for all $i$, we have $|\,\mathsf{openings}(S^*[i], R_0[i])| \leq 1$ with probability 1 (over the coin tosses of $S^*[i]$ and $R_0[i]$[8]) where $\mathsf{openings}(\cdot)$ is defined as in Section 6.4.

The following two lemmas, Lemma 7.7 and 7.9, provide us a way to understand the binding property (in an average case sense) of $(\mathcal{S}, \mathcal{R})$, the amplified hiding scheme as presented in Protocol 7.1, in terms of $(S, R)$. We might occasionally drop the superscript notations (1) and (2) from the notations if it is clear which phase we are referring to.

**Lemma 7.7** (intermediate step binding preservation)**.** *For some constant $D \in \mathbb{N}$ and any integers $t \in [1, n]$, $\beta_1, \ldots, \beta_\ell \in [0, D-1]$, and $\beta_{\ell+1} \in [0, n]$, letting $[(S_j, R_j)]_{1\leq j\leq \ell}$ be the intermediate commitment schemes obtained in the execution of Algorithm 7.2 with parameters $D$, $t$, and $(\beta_1, \ldots, \beta_{\ell+1})$, there exists a binding set $\mathcal{B}$ such that the following two conditions hold for each $j = 1, 2, \ldots, \ell$:*

*(B.1) For every deterministic sender $S^*$ with the* unique binding property*,*

$$\mathrm{E}\left[\left|\mathsf{openings}(S^*, R^1_j)(\mathcal{B})\right|\right] < 32,$$

*where the expectation is taken over the coins tosses of $R^1_j$.*

*(B.2) For every $\tau \in \mathcal{B}$ and for every deterministic sender $S^*$,*

$$\mathrm{E}\left[\left|\mathsf{openings}(S^*, R^2_j)(\tau)\right|\right] < 32,$$

*where the expectation is taken over the coins tosses of $R^2_j$.*

*Proof.* We proceed to prove by induction on $j$. In fact, we will start with a base case of $j = 0$, i.e., consider the scheme $(S_0, R_0)$ from Section 6. By Lemma 6.12, we know that Scheme $(S_0, R_0)$ satisfies both conditions (B.1) and (B.2). (Although Lemma 6.12 guarantees that $(S_0, R_0)$ satisfies condition (B.1) only for PPT $S^*$, it is also trivially satisfied for computationally unbounded $S^*$ with the unique binding property.)

For the inductive step, we assume $(S_j, R_j)$ satisfy both (B.1) and (B.2), and show that so does $(S_{j+1}, R_{j+1})$. Note that $(S_{j+1}, R_{j+1})$ is obtained by the amplification procedure (Protocol 7.1) that combines $m$ sequential executions of $(S_j, R_j)$, i.e., $(S_{j+1}, R_{j+1}) = \mathsf{Amplify}(S_j, R_j)$. Hence, for convenience of notation we will denote $(S_j, R_j)$ and $(S_{j+1}, R_{j+1})$ as $(S, R)$ and $(\mathcal{S}, \mathcal{R})$ respectively. The $i$-th execution of $(S, R)$ in $(\mathcal{S}, \mathcal{R})$ is denoted as $(S[i], R[i])$, not to be confused with the subscript indexing notation of $(S_j, R_j)$.

[8] Note that $S^*[i]$ is probabilistic even if $S^*$ is deterministic, because it simulates all of the random choices of $R_j$ other than those of $R_0[i]$.

Also throughout this proof, the value of $m$ will be fixed to $D$, although we will keep writing $m$. Let $\mathcal{B}$ be the binding set for $(S, R)$. We define our new binding set $\mathcal{B}'$ for $(\mathcal{S}, \mathcal{R})$ in terms of $\mathcal{B}$ as follows:

$$\mathcal{B}' = \{(\tau_1, \ldots, \tau_m) : \exists\, j_1, \ldots, j_{\beta+1} \text{ such that } \tau_{j_1}, \ldots, \tau_{j_{\beta+1}} \in \mathcal{B}\}.$$

That is, a transcript $\tau' = (\tau_1, \ldots, \tau_m) \in \mathcal{B}'$ iff at least $\beta + 1$ of $\tau_j$'s are in $\mathcal{B}$. Conversely, $\tau' \notin \mathcal{B}'$ iff at least $m - \beta$ of the $\tau_j$'s are not in $\mathcal{B}$.

**Property (B.1).** Consider a deterministic $S^*$ with the unique binding property interacting with $\mathcal{R}^1$. The random coins of $\mathcal{R}^1$ can be broken up into independent random coins of $R^1[1], \ldots, R^1[m]$ and $R^1_{\text{IH}}$, the receiver in the Interactive Hashing Scheme.

Recall that the $m$ executions of $(S, R)$ in $(\mathcal{S}, \mathcal{R})$ are sequential. We want to focus on the interaction of $S^*$ with (the commit phase of) $R^1[i]$. To do so, define $S^*[i]$, the sender interacting with $R^1[i]$, as follows: $S^*[i]$ simulates $S^*$ using fixed coin tosses $r_j$ for all the previous $R^1[j]$'s (for all $j < i$) and after the interaction with $R^1[i]$, $S^*[i]$ outputs all the valid openings that occur in some continuation of $S^*$'s interaction with $R[i]$ (by enumerating over all coin tosses of the future $R[j]$'s, $j > i$, the coin tosses of $R^1_{\text{IH}}$, and the coin tosses of $\mathcal{R}^2$). Observe that $S^*[i]$ inherits the unique binding property from $S^*$. We will write $S^*[i](r_1, \ldots, r_{i-1})$ to indicate the fixed coin tosses $r_j$ that are used by $S^*[i]$ in simulating $R^1[j]$.

Let $X_i(r_1, \ldots, r_i) = \left|\mathsf{openings}(S^*[i](r_1, \ldots, r_{i-1}, R^1[i](r_i))(\mathcal{B})\right|$, i.e., a count of the number of valid decommitment in $i$-th execution, when the sender uses simulated coin tosses $r_1, \ldots, r_{i-1}$ and $R^1[i]$ uses coin tosses $r_i$. Let $U = (U_1, \ldots, U_m)$, where $U_i$ denotes the uniform distribution on coin tosses $r_i$ for $R[i]$; note that these are independent because the honest receiver tosses independent coins for each execution. We now consider the random variables $X_i(U) = X_i(U_1, \ldots, U_i)$.

By our induction hypothesis, for all fixed $(r_1, \ldots, r_{i-1})$, we have

$$\mathrm{E}\left[X_i(U) | U_1 = r_1, \ldots, U_{i-1} = r_{i-1}\right] = \mathrm{E}\left[X_i(r_1, \ldots, r_{i-1}, U_i)\right] < 32.$$

Because the previous $X_j(U)$'s, for $j < i$, only depend on $U_1, \ldots, U_j$, we have that the expected value of $X_i$ is less than 32 even given any previous values of $X_j$'s. That is, $\mathrm{E}\left[X_i|_{X_1 = x_1, \ldots, X_{i-1} = x_{i-1}}\right] < 32$ for any $(x_1, \ldots, x_{i-1}) \in \mathrm{Supp}(X_1, \ldots, X_{i-1})$. The following claim allows us to bound the expectation of the product of these random variables.

**Claim 7.8.** *Let $Y_1, \ldots, Y_\ell$ be nonnegative real-valued random variables such that for all $i = 1, 2, \ldots, \ell$,*

$$\mathrm{E}[Y_i|_{Y_1 = y_1, \ldots, Y_{i-1} = y_{i-1}}] < \alpha_i \in \mathbb{R}^+,$$

*for any $(y_1, \ldots, y_{i-1}) \in \mathrm{Supp}(Y_1, \ldots, Y_{i-1})$. Then,*

$$\mathrm{E}\left[\prod_{i=1}^{\ell} Y_i\right] < \prod_{i=1}^{\ell} \alpha_i.$$

*Proof of claim.* Note that

$$\begin{aligned}\mathrm{E}[Y_1 \cdots Y_\ell] &= \mathrm{E}[Y_1 \cdots Y_{\ell-1}] \cdot \mathop{\mathrm{E}}_{(y_1, \ldots, y_{\ell-1}) \leftarrow (Y_1, \ldots, Y_{\ell-1})} \left[\mathrm{E}[Y_\ell|_{Y_1 = y_1, \ldots, Y_{\ell-1} = y_{\ell-1}}]\right] \\ &< \mathrm{E}[Y_1 \cdots Y_{\ell-1}] \cdot \alpha_\ell,\end{aligned}$$

and the claim follows by induction on $\ell$. □

As noted above, it is always the case that $\mathrm{E}[X_i] < 32$, regardless of the instantiation of previous $X_j$'s, for $j < i$. Note that Claim 7.8 also applies to computing the expectation of $\prod_{i \in J} X_i$, for any subset $J \subset [m]$, since any subset of the $X_i$'s (preserving the right order) satisfy the condition of claim.

Once the $m$ commitments $R^1[i]$ are complete, we can define a random variable $A = A(U)$ that denotes the set of values $a = (a_1, \ldots, a_m)$'s for which the sender $S^*$ produces a valid opening with respect to $\mathcal{B}'$ in some continuation of the protocol. By the definition of $\mathcal{B}'$, this means that $a = (a_1, \ldots, a_m)$ is valid if at least $m - \beta$ of those are $a_i$'s correspond to decommitments that are in $\mathcal{B}$. For those $a_i$'s corresponding to decommitments that are in $\mathcal{B}$, the number of possible values that $a_i$ can take on is $X_i(U)$. And for those $a_i$'s correspond to decommitments that are not in $\mathcal{B}$, we can only bound the number of possible values that $a_i$ can take on by $2^k$ (since $a_i$ is a $k$-bit string).

$$\begin{aligned}
\mathop{\mathrm{E}}_{U}[|A(U)|] &\leq \mathop{\mathrm{E}}_{U}\left[\sum_{J \subseteq [m], |J| \geq m-\beta} \prod_{i \in J} X_i(U) \prod_{i \notin J} 2^k\right] \\
&= \sum_{J \subseteq [m], |J| \geq m-\beta} \mathop{\mathrm{E}}_{U}\left[\prod_{i \in J} X_i(U) \prod_{i \notin J} 2^k\right] \\
&< \sum_{J \subseteq [m], |J| \geq m-\beta} \prod_{i \in J} 32 \cdot \prod_{i \notin J} 2^k && \text{(by Claim 7.8))} \\
&\leq 2^m \cdot 32^{m-\beta} \cdot (2^k)^\beta && \text{(because } 32 < 2^k\text{)} \\
&\leq 2^{(\beta+1)(k-1)+6m-k+1} \\
&= 2^{\alpha_1 - (k-6m-4)}.
\end{aligned}$$

Let random variable $T = \{(h_1, h_1(a)) : h_1 \in \mathcal{H}_1 \wedge a \in A\}$. Since $\mathrm{E}[|A|] \leq 2^{\alpha_1 - (k-6m-4)}$ and the range of $h_1 \in \mathcal{H}_1$ is $\alpha_1$, the expected density of $T$ satisfies $\mathrm{E}[\mu(T)] \leq \mathrm{E}[|A|] \cdot 2^{-\alpha_1} \leq 2^{-(k-6m-4)}$, where the expectation is taken over the coins tosses $U = (U_1, \ldots, U_m)$. Note that $T$ is independent of the coin tosses of $R^1_{\mathrm{IH}}$ in the first phase interactive hashing (though not independent of the coin tosses of $\mathcal{R}^1$).

Finally, we have

$$\mathop{\mathrm{E}}_{\text{coins } \mathcal{R}^1}\left[\left|\mathsf{openings}(S^*, \mathcal{R}^1)(\mathcal{B}')\right|\right] \leq \mathop{\mathrm{E}}_{\text{coins } R^1_{\mathrm{IH}}, T}\left[\left|\{d^{(1)} : C^{(1)}(d^{(1)}) \in T\}\right|\right],$$

where in the second expectation, $C = \mathsf{output}(S^*, R^1_{\mathrm{IH}})$. By Lemma 6.10,

$$\mathop{\mathrm{E}}_{\text{coins } R^1_{\mathrm{IH}}, T}\left[\left|\{d^{(1)} : C^{(1)}(d^{(1)}) \in T\}\right|\right] < 24 + 2^{k'+1} \cdot \mathrm{E}[\mu(T)] < 32,$$

with the last inequality following from $k' < k - 8m - 8$.

**Property (B.2).** We use the same approach as above, except this time, we consider all deterministic $S^*$, as opposed to only those with the unique binding property. Also we need to fix a binding transcript $\tau = (\tau_1, \ldots, \tau_m) \in \mathcal{B}'$. Let $J$ be the set of indices such that $\tau_i \in \mathcal{B}$.

As done previously, we define $S^*[i]$ and set $X_i = \left|\mathsf{openings}(S^*[i], R^2[i])(\tau_i)\right|$, where $S^*[i]$. By our induction hypothesis, for all $i \in J$, we have

$$\mathrm{E}\left[X_i|_{X_1=x_1,\ldots,X_{i-1}=x_{i-1}}\right] < 32,$$

for any $(x_1, \ldots, x_{i-1}) \in \mathrm{Supp}(X_1, \ldots, X_{i-1})$.

Let random variable $B$ denote the denotes the set of values $b = (b_1, \ldots, b_m)$ for which the sender $S^*$ produces a valid opening in some continuation of the protocol. Noting that $X_i$ can be as large as $2^k$ for indices $i \notin J$, we have

$$\begin{aligned}
\mathrm{E}\left[|B|\right] &\leq \mathop{\mathrm{E}}_{\text{coins } R^2[1],\ldots,R^2[m]}\left[\prod_{i\in J} X_i \prod_{i\notin J} 2^k\right] \\
&< \prod_{i\in J} 32 \cdot \prod_{i\notin J} 2^k && \text{(by Claim 7.8))} \\
&\leq 32^{\beta+1} \cdot (2^k)^{m-\beta-1} && \text{(because } 32 < 2^k\text{)} \\
&\leq 2^{(m-\beta)(k-1)-(k-6m)} && \text{(because } m > 5\text{)} \\
&= 2^{\alpha_2-(k-6m-3)}.
\end{aligned}$$

Let random variable $T = \{(h_2, h_2(b)) : h_2 \in \mathcal{H}_1 \wedge b \in B\}$. Since $\mathrm{E}[|B|] \leq 2^{\alpha_2-(k-6m-3)}$ and the range of $h_2 \in \mathcal{H}_2$ is $\alpha_2$, the expected density of $T$ satisfies $\mathrm{E}[\mu(T)] \leq \mathrm{E}[|B|] \cdot 2^{-\alpha_2} \leq 2^{-(k-6m-3)}$, where the expectation is taken over the coins tosses of $R_1^2, \ldots, R_m^2$. Note that $T$ is independent of the coin tosses of $R_{\mathrm{IH}}^2$ in the second phase interactive hashing (though not independent of the coin tosses of $\mathcal{R}^2$). Finally, we have

$$\mathop{\mathrm{E}}_{\text{coins } \mathcal{R}^2}\left[\left|\mathsf{openings}(S^*, \mathcal{R}^2)(\tau')\right|\right] \leq \mathop{\mathrm{E}}_{\text{coins } R_{\mathrm{IH}}^2, T}\left[\left|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T\}\right|\right],$$

where in the second expectation, $C = \mathsf{openings}(S^*(T), R_{\mathrm{IH}})$. By Lemma 6.10,

$$\mathop{\mathrm{E}}_{\text{coins } R_{\mathrm{IH}}^2, T}\left[\left|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T\}\right|\right] < 24 + 2^{k'+1} \cdot \mathrm{E}[\mu(T)] < 32,$$

with the last inequality following from $k' < k - 8m - 8$. □

**Lemma 7.9** (final step binding preservation). *For some constant $D \in \mathbb{N}$ and any integers $t \in [1, n]$, $\beta_1, \ldots, \beta_\ell \in [0, D-1]$, and $\beta_{\ell+1} \in [0, n]$, letting $(\mathsf{S}, \mathsf{R})$ be the final output of Algorithm 7.2 with parameters $D$, $t$, and $(\beta_1, \ldots, \beta_{\ell+1})$, there exists a binding set $\mathcal{B}'$ such that the following two conditions hold:*

*(B.1) For every deterministic sender $S^*$ with the* unique binding property*, with probability $1 - 2^{-\Omega(n)}$ over the coin tosses of $\mathsf{R}^1$,*

$$\left|\mathsf{openings}(S^*, \mathsf{R}^1)(\mathcal{B}')\right| \leq 1.$$

*(B.2) For every $\tau \in \mathcal{B}'$ and for every deterministic sender $S^*$, with probability $1 - 2^{-\Omega(n)}$ over the coin tosses of $\mathsf{R}^2$,*

$$\left|\mathsf{openings}(S^*, \mathsf{R}^2)(\tau)\right| \leq 1.$$

*Proof.* From Lemma 7.7, we have scheme $(S_\ell, R_\ell)$ with an associated binding set $\mathcal{B}$ satisfying both conditions (B.1) and (B.2) in Lemma 7.7. Scheme $(\mathsf{S}, \mathsf{R}) = \mathsf{Amplify}(S_\ell, R_\ell)$, and hence we will need to show that the amplification boosts the binding by making sure both $\left|\mathsf{openings}(S^*, \mathsf{R}^1)(\mathcal{B})\right| \leq 1$ and $\left|\mathsf{openings}(S^*, \mathsf{R}^2)(\tau)\right| \leq 1$ with probability $1 - 2^{-\Omega(n)}$.

Throughout this proof, the value of $m$ will be fixed to $n$ (as in Step 3 of Algorithm 7.2), although we will keep writing $m$. We define our new binding set $\mathcal{B}'$ for $(\mathsf{S}, \mathsf{R})$ in terms of $\mathcal{B}$ as follows:

$$\mathcal{B}' = \{(\tau_1, \ldots, \tau_m) : \exists\, j_1, \ldots, j_{\beta+1} \text{ such that } \tau_{j_1}, \ldots, \tau_{j_{\beta+1}} \in \mathcal{B}\}.$$

That is, a transcript $\tau' = (\tau_1, \ldots, \tau_m) \in \mathcal{B}'$ iff at least $\beta + 1$ of $\tau_j$'s are in $\mathcal{B}$. Conversely, $\tau' \notin \mathcal{B}'$ iff at least $m - \beta$ of the $\tau_j$'s are not in $\mathcal{B}$.

**Property (B.1).** Using the same analysis and notations as in the proof of Lemma 7.7, we have that

$$\mathop{\mathrm{E}}_{\text{coins } R^1[1], \cdots, R^1[m]} [|A|] \leq 2^m \cdot 32^{m-\beta} \cdot (2^k)^\beta \leq 2^{\beta k + 6m},$$

where $A$ is the random variable denoting the set of values $a = (a_1, \ldots, a_m)$'s for which the sender $S^*$ produces a valid opening with respect to $\mathcal{B}'$ in some continuation of the protocol.

Since $\delta = \Omega(1)$ and $k_\ell \geq \log n$, observe that $\alpha_1 = \lfloor(\beta + \frac{1}{3}\delta n)k\rfloor > \beta k + 8n$, for large enough values of $n$. Let random variable $T = \{(h_1, h_1(a)) : h_1 \in \mathcal{H}_1 \wedge a \in A\}$. Since the range of $h_1 \in \mathcal{H}_1$ is $\{0,1\}^{\alpha_1}$, the density of $T$ satisfies

$$\mathop{\mathrm{E}}_{\text{coins } R^1[1], \cdots, R^1[m]} [\mu(T)] \leq \mathrm{E}[|A|] \cdot 2^{-\alpha_1} < 2^{\beta k + 6m} \cdot 2^{-(\beta k + 8n)} = 2^{-2n},$$

since $m = n$. Hence, with probability at least $1 - 2^{-n}$ over the coins tosses of $R^1[1], \ldots, R^1[m]$, we have that

$$\mu(T) \leq 2^{-2n} \cdot 2^n = 2^{-n}.$$

By Lemma 4.3, we can conclude that for such a $T$ (with $\mu(T) \leq 2^{-n}$),

$$\Pr_{\text{coins } R^1_{\mathrm{IH}}} \left[ \left| \{d^{(1)} : C^{(1)}(d^{(1)}) \in T\} \right| > 1 \right] = 2^{-\Omega(n)}.$$

Finally, we have

$$\begin{aligned}
&\Pr_{\text{coins } \mathcal{R}^1} \left[ \left|\mathsf{openings}(S^*, \mathcal{R}^1)\right| > 1 \right] \\
&\leq \Pr_{\text{coins } R^1_1, \cdots, R^1_m} \left[\mu(T) > 2^{-n}\right] + \Pr_{\text{coins } R^1_{\mathrm{IH}}} \left[ |\{d^{(1)} : C^{(1)}(d^{(1)}) \in T\}| > 1 \;\middle|\; \mu(T) \leq 2^{-n} \right] \\
&= 2^{-\Omega(n)}.
\end{aligned}$$

**Property (B.2).** Fix any $\tau' \in \mathcal{B}'$. Again, we use the same analysis and notations as in the proof of Lemma 7.7 to get:

$$\mathop{\mathrm{E}}_{\text{coins } R^2[1], \cdots, R^2[m]} [|B|] \leq 32^{\beta+1} \cdot (2^k)^{m-\beta-1} \leq 2^{(m-\beta)k + 5m},$$

where $B$ is the random variable denoting the set of values $b = (b_1, \ldots, b_m)$'s for which the sender $S^*$ produces a valid opening in some continuation of the protocol

Since $\delta = \Omega(1)$ and $k \geq \log n$, observe that $\alpha_2 = \lfloor (n - \beta + \frac{1}{3}\delta n)k \rfloor > (n-\beta)k + 7n$, for large enough values of $n$. Let random variable $T = \{(h_2, h_2(b)) : h_2 \in \mathcal{H}_2 \wedge b \in B\}$. Since the range of $h_2 \in \mathcal{H}_2$ is $\{0,1\}^{\alpha_2}$, the density of $T$ satisfies

$$\mathop{\mathrm{E}}_{\text{coins } R^2[1], \cdots, R^2[m]}[\mu(T)] \leq \mathrm{E}[|B|] \cdot 2^{-\alpha_2} < 2^{(m-\beta)k+5m} \cdot 2^{-((n-\beta)k+7n)} = 2^{-2n},$$

since $m = n$. Hence, with probability at least $1 - 2^{-n}$ over the coins tosses of $R^2[1], \ldots, R^2[m]$, we have that

$$\mu(T) \leq 2^{-2n} \cdot 2^n = 2^{-n}.$$

By Lemma 4.3, we can conclude that for such a $T$ (with $\mu(T) \leq 2^{-n}$),

$$\Pr_{\text{coins } R^2_{\mathrm{IH}}}\left[\left|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T\}\right| > 1\right] = 2^{-\Omega(n)}.$$

Finally, we have

$$\begin{aligned}
&\Pr_{\text{coins } \mathcal{R}^2}\left[\left|\mathsf{openings}(S^*, \mathcal{R}^2)(\tau')\right| > 1\right] \\
&\leq \Pr_{\text{coins } R^2_1, \cdots, R^2_n}\left[\mu(T) > 2^{-n}\right] + \Pr_{\text{coins } R^2_{\mathrm{IH}}}\left[|\{d^{(2)} : C^{(2)}(d^{(2)}) \in T\}| \mid \mu(T) \leq 2^{-n}\right] \\
&= 2^{-\Omega(n)}.
\end{aligned}$$

Our proof is complete. □

## 7.5 One-Way Functions implies Collection of Commitments

In this section, we prove Theorem 3.1, restated in the next theorem.

**Theorem 7.10** (Restatement of Theorem 3.1). *If one-way functions exist, then on security parameter $n$, we can construct in time* $\mathrm{poly}(n)$ *a collection of public-coin 2-phase commitment schemes* $\mathrm{Com}_1, \cdots, \mathrm{Com}_m$ *for* $m = \mathrm{poly}(n)$ *such that:*

- *There exists an index $i$ such that scheme* $\mathrm{Com}_i$ *is statistically hiding. (This property holds regardless of whether the function for which the scheme is based on is one-way or not.)*
- *For every index $j$, scheme* $\mathrm{Com}_j$ *is* $\binom{2}{1}$*-computationally binding.*

*Proof of Theorem 3.1.* We apply Algorithm 7.2 to the scheme $(S_0, R_0)$ based on one-way function. In doing so, we obtain a collection of commitments by enumerating over all the polynomially many choices of the integers $t \in [1, n]$, $\beta_1, \ldots, \beta_\ell \in [0, D-1]$, and $\beta_{\ell+1} \in [0, n]$. (Note that the number of choices is $n \cdot D^\ell \cdot (n+1) = \mathrm{poly}(n)$, as $D$ is a constant and $\ell = \log n$.) By Lemma 7.3, the resulting commitment schemes $\mathrm{Com}_1, \cdots, \mathrm{Com}_m$ all run in probabilistic polynomial time. The hiding and binding properties are given by Lemmas 7.11 and 7.12 below. □

**Lemma 7.11** (statistically hiding). *There exists a constant $D \in \mathbb{N}$, integers $t \in [1, n]$, $\beta_1, \ldots, \beta_\ell \in [0, D-1]$, and $\beta_{\ell+1} \in [0, n]$ such that the 2-phase commitment scheme* $(\mathsf{S}, \mathsf{R})$ *produced by Algorithm 7.2 with parameters $D$, $t$, and $(\beta_1, \ldots, \beta_{\ell+1})$ is statistically hiding in the sense Definition 2.5 (regardless of whether the function $f$ on which the scheme is based on is one-way or not).*

*Proof.* We prove by induction on the properties of $(S_j, R_j)$ for $j = 0, 1, \ldots, \ell$. The induction hypothesis is that $(S_j, R_j)$ has two associated sets $T_{1,j}, T_{2,j} \subseteq \{0,1\}^{nm^j}$ such that for all $R^*$, the following holds:

1. $T_{1,j} \cup T_{2,j} = \{0,1\}^{nm^j}$ and $\mu(T_{1,j} \cap T_{2,j}) \geq \min\{2^j/n, 1/2D\}$.
2. $\mathrm{CP}^{1/2}(A|V) \leq \sqrt{2^{-(k_j-1)}}$, where $A = \mathsf{output}_S(S^1_{c,j}(T_{1,j}), R^*)$ and $V = \mathsf{view}_{R^*}(S^1_{c,j}(T_{1,j}), R^*)$.
3. $\mathrm{CP}^{1/2}(B|W,\Lambda) \leq \sqrt{2^{-(k_j-1)}}$, where $B = \mathsf{output}_S(S^2_{c,j}(T_{2,j}), R^*)(\Lambda)$ and $W = \mathsf{view}_{R^*}(S^2_{c,j}(T_{2,j}), R^*)(\Lambda)$, and $\Lambda = \mathsf{transcript}(S^1_j(T_{2,j}), R^*)$,

where $k_j$ is defined as in Algorithm 7.2.

The base case of $j = 0$ follows from Lemma 6.11, and Lemma 7.4 proves the induction step. Finally, observe that $\mu(T_{1,\ell} \cap T_{2,\ell}) \geq \min\{2^\ell/n, 1/(2D)\} = \Omega(1)$ since $\ell = \log n$. By Lemma 7.5, there exists two sets $T_{1,\ell+1}$ and $T_{2,\ell+1}$ such that for all $R^*$, the following holds:

1. $\mu(T_{1,\ell+1}), \mu(T_{2,\ell+1}) > 1 - 2^{-\Omega(n)}$.
2. $(A, V)$ is $2^{-\Omega(n)}$-close to $(U_1, V)$, where $A = \mathsf{output}_{\mathsf{S}}(\mathsf{S}^1_c(T_{1,\ell+1}), R^*)$ and $V = \mathsf{view}_{R^*}(\mathsf{S}^1_c(T_{1,\ell+1}), R^*)$.
3. $(B, W, \Lambda)$ is $2^{-\Omega(n)}$-close to $(U_1, W, \Lambda)$, where $B = \mathsf{output}_{\mathsf{S}}(\mathsf{S}^2_c(T_{2,\ell+1}), R^*)(\Lambda)$ and $W = \mathsf{view}_{R^*}(\mathsf{S}^2_c(T_{2,\ell+1}), R^*)(\Lambda)$, and $\Lambda = \mathsf{transcript}(\mathsf{S}^1(T_{2,\ell+1}), R^*)$.

Our proof is complete. □

**Lemma 7.12** (1-out-of-2-computationally binding)**.** *There exists a constant $D \in \mathbb{N}$ such that for all integers $t \in [1, n]$, $\beta_1, \ldots, \beta_\ell \in [0, D-1]$, and $\beta_{\ell+1} \in [0, n]$, the 2-phase commitment scheme $(\mathsf{S}, \mathsf{R})$ produced by Algorithm 7.2 with parameters $D$, $t$, and $(\beta_1, \ldots, \beta_{\ell+1})$ is computationally $\binom{2}{1}$-binding in the sense of Definition 2.6. (Here the function $f$ for which the scheme is based on needs to be hard to invert.)*

*Proof.* By Lemma 7.9, we have established that the 2-phase commitment scheme $(\mathsf{S}, \mathsf{R})$ produced by Algorithm 7.2 satisfies the first condition of Definition 2.6. In addition, it also satisfies the second condition for all $S^*$ with the *unique binding* property. Stated formally, for every deterministic (and computationally unbounded) $S^*$ with the unique binding property,

$$\Pr\left[\left|\mathsf{openings}(S^*, \mathsf{R}^1)\right| \leq 1\right] = 1 - 2^{-\Omega(n)}, \tag{5}$$

where the probability is taken over the coins of $\mathsf{R}^1$.

Thus, it suffices to prove is that any PPT $S^*$ breaking the second condition of Definition 2.6 with probability $\varepsilon$ will either (i) yield a *PPT* $\hat{S}$ that violates the computational $\binom{2}{1}$-binding property of $(S_0, R_0)$ with probability at least $\varepsilon^{O(1)}/\mathrm{poly}(n)$, or (ii) yield a computationally unbounded $\hat{S}$ that has the unique binding property and succeeds with probability greater than $\varepsilon/2$. In both cases, $\varepsilon$ needs to be negligibly small in order to avoid a contradiction. Without loss of generality, we may assume adversarial PPT sender $S^*$ to be deterministic since we can set its coins to maximizes its success probability.

From now on, let $\varepsilon$ be the probability that the deterministic $S^*$ breaks the second condition of Definition 2.6 with respect to scheme $(\mathsf{S}, \mathsf{R})$. By the way we defined $(\mathsf{S}, \mathsf{R})$, it contains polynomially many executions of $(S_0, R_0)$. Let $N = n \cdot D^\ell$ denote such number.

Let $\mathbf{z}$ denote the transcript of $(S^*, \mathsf{R})$. Contained in $\mathbf{z}$ is also a first-phase commitment $z[i]$ for the $i$-th execution of $R_0$, denoted $R_0[i]$ (for all $i = 1, 2, \ldots, N$). Let $\hat{z}[i]$ be the partial transcript of $\mathbf{z}$ up to and including the first commit stage of $R_0[i]$. Note that $z[i]$ is a suffix of $\hat{z}[i]$, and $\hat{z}[i]$ is a prefix of $\mathbf{z}$.

For all index $i \in [N]$, partial transcripts $\hat{z}[i]$ ending with the first commit stage of $R_0[i]$ and $d \in \{0,1\}^{k_0}$, define

$$p_{i,\hat{z}[i],d} = \Pr_{\mathbf{z} \leftarrow (S^*, \mathsf{R}^1)} [\mathbf{z} \text{ contains a valid opening of } z[i] \text{ to value } d \,|\, \mathbf{z} \text{ begins with } \hat{z}[i]]\,,$$

where as usual by a valid opening, we mean that the transcript $\tau[i]$ of $S^*$'s interaction with $R_0[i]$ contains an opening of $z[i]$ to the value $d$, the first phase of $\tau[i]$ is not in the binding set $\mathcal{B}_0$, and $R_0[i]$ accepts in both phases of $\tau[i]$.

Let $K = 2^{k_0}$, where $k_0$ is the message length in $(S_0, R_0)$. We have two cases to consider.

**Case 1:** There exists an $i \in [N]$ such that with probability at least $\frac{\varepsilon}{4NK}$ over $\hat{z}[i]$, there exists $d \neq d'$ with both $p_{i,\hat{z}[i],d}, p_{i,\hat{z}[i],d'} > \frac{\varepsilon}{4NK}$.

In this case, we violate the computational $\binom{2}{1}$-binding property of $(S_0, R_0)$ by considering the following sender $\hat{S}$ interacting with $R_0[i]$.

1. Select a random $i \leftarrow [N]$.
2. Run $S^*$ with $\mathsf{R}^1$, simulating all of the messages of $\mathsf{R}^1$ internally except for those of $R_0[i]$. Halting after the first commit stage of $R_0[i]$, we obtain a partial transcript $\hat{z}[i]$. From $\hat{z}[i]$, we get $z[i]$, the first-phase commitment of $R_0[i]$.
3. Record the current state $\psi$ of $S^*$ and $\mathsf{R}^1$.
4. Continue the execution of $S^*$ with $\mathsf{R}^1$ from $\psi$ to obtain a decommitment to a value $d$ in the interaction with $R_0[i]$.
5. Repeat Step 4 with independent randomness in continuing the execution of $S^*$ with $\mathsf{R}^1$ to obtain a decommitment to a value $d'$. (This can be done since $\mathsf{R}$ is public coin, i.e., just sends independent random coins at each round, and $S^*$ is deterministic.)

Because our goal was to violate the computational $\binom{2}{1}$-binding property of $(S_0, R_0)$, we succeed in the above algorithm if $d \neq d'$ and decommitments produced are valid. We calculate our success probability as follows: We guess correct index $i \in [N]$ with probability $1/N$. Given that we guess the correct $i$, we get the desired $\hat{z}[i]$ with probability at least $\frac{\varepsilon}{4NK}$. Now, when we do two independent continuations of $\hat{z}[i]$ we arrive at two different decommitted values with probability greater than $(\frac{\varepsilon}{4NK})^2$. Consequently, we violate the computational $\binom{2}{1}$-binding property of $(S_0, R_0)$ (i.e., win the game in Condition 2 of Definition 2.6) with probability greater than

$$\frac{1}{N} \cdot \frac{\varepsilon}{4NK} \cdot \left(\frac{\varepsilon}{4NK}\right)^2 = \frac{1}{N} \cdot \left(\frac{\varepsilon}{4NK}\right)^3 = \left(\frac{\varepsilon}{n}\right)^{O(1)},$$

since $K = 2^{k_0} = 2^{O(\log n)} = \mathrm{poly}(n)$ and $N = n \cdot D^{\ell} = n \cdot O(1)^{O(\log n)} = \mathrm{poly}(n)$. This forces $\varepsilon$ to be a negligible function.

**Case 2.** For all $i \in [N]$, with probability greater than $1 - \frac{\varepsilon}{4NK}$ over $\hat{z}[i]$, there is at most one $d$ such that $p_{i,\hat{z}[i],d} > \frac{\varepsilon}{4NK}$.

Define $d^*(\hat{z}[i])$ to be the value of $d$ that maximizes $p_{i,\hat{z}[i],d}$. Taking a union bound over all the rest of the $p_{i,\hat{z}[i],d'} < \frac{\varepsilon}{4NK}$, we have that

$$\begin{aligned}
&\Pr_{\mathbf{z} \leftarrow (S^*,\mathsf{R})} [S^* \text{ opens some } z[i] \text{ to a value other than } d^*(\hat{z}[i])] \\
&\leq \sum_{i=1}^{N} \left( \frac{\varepsilon}{4NK} \cdot K + \Pr_{\hat{z}[i]} \left[ \text{exists more than one } d \text{ such that } p_{i,\hat{z}[i],d} > \frac{\varepsilon}{4NK} \right] \right) \\
&< N \cdot \left( \frac{\varepsilon}{4NK} \cdot K + \frac{\varepsilon}{4NK} \right) \\
&< \frac{\varepsilon}{2}.
\end{aligned}$$

Let $\hat{S}$ be the adversary that mimics $S^*$ except that it halts and fails if $S^*$ attempts to open some $z[i]$ to a value other than $d^*(\hat{z}[i])$, for all $i \in [N]$ and all $\hat{z}[i]$. By the way we defined $\hat{S}$, the final outcome of $(\hat{S}, \mathsf{R}^1)$ will only differ with the original final outcome of $(S^*, \mathsf{R}^1)$ with probability at most $\varepsilon/2$ over the coin tosses of $\mathsf{R}^1$. In addition, $\hat{S}$ has the unique binding property. By Equation (5) above, $\left|\mathsf{openings}(\hat{S}, \mathsf{R}^1)\right| > 1$ occurs with at most negligible probability over the coin tosses of $\mathsf{R}^1$. Hence, $\left|\mathsf{openings}(S^*, \mathsf{R}^1)\right| > 1$ occurs with probability at most $\text{neg}(n) + \varepsilon/2$. We started off assuming that $S^*$ breaks property (B.1) of scheme $(\mathsf{S}, \mathsf{R})$ with probability at least $\varepsilon$, i.e. $\left|\mathsf{openings}(S^*, \mathsf{R}^1)\right| > 1$ with probability at least $\varepsilon$. Thus $\varepsilon \leq \text{neg}(n) + \varepsilon/2$, which implies. that $\varepsilon = \text{neg}(n)$. □

# 8 Statistical Zero-Knowledge Arguments from 1-out-of-2-Binding Commitments

In order to prove Theorem 3.2, we provide an overview of our construction of statistical zero-knowledge arguments for all of **NP** from 1-out-of-2-binding commitment schemes. Our construction is identical to that of Nguyen and Vadhan [NV06]. However, the analysis of the soundness property is more involved since we are considering 1-out-of-2 *computationally* binding commitments rather than 1-out-of-2 statistically binding commitments.

## 8.1 Zero-Knowledge Protocol for Hamiltonicity

It will be convenient to present our protocols based on an abstraction of standard zero-knowledge proofs for **NP**-complete problems [GMW91, Blu87]. By repeating the standard zero-knowledge proof for Hamiltonicity [Blu87] a total of $q = O(\log n)$ times in parallel (for $n = |x|$), we may assume that every language $L \in \mathbf{NP}$ has a public-coin zero-knowledge proof $(P, V)(x)$ of the form:

1. $P$ commits to $\ell$ bits $(b_1, b_2, \cdots, b_\ell)$, and sends the commitments to $V$. (In Hamiltonicity, this is a commitment to the adjacency matrix of permuted graphs)

2. $V$ sends a challenge $c \leftarrow \{0,1\}^q$. (This tells the prover whether to reveal the permutation or a cycle in the permuted graph in each of the executions)

3. $P$ sends a sequence of indices $U \in [\ell]^q$, where $U$ is determined by the challenge, the **NP**-witness and the prover's coin tosses. $P$ opens the commitments to $b_i$ for $i \in U$. ($U$ consists of openings to the entire graph or cycles. By using appropriate "dummy" commitments, we can ensure that the subsets of indices are of fixed size $u = u(n)$.)

4. $V$ checks that "$U$ and $(b_i)_{i \in U}$ are valid with respect to the challenge $c$" and that the opened commitments are valid. (The verifier will check that either these values correspond to the adjacency matrix of a permuted graph or that they correspond to a Hamiltonian cycle.)

This proof system has perfect completeness and soundness $2^{-q} = 1/\mathrm{poly}(n)$ if the commitment scheme used is perfectly binding. More generally, we can say that if $x \in \Pi_N$, then with probability $1 - 1/\mathrm{poly}(n)$, either the verifier rejects or the prover breaks one of the commitments. If the commitment scheme used is statistically hiding, the protocol is statistical zero knowledge.

Let us abstract the properties of the generic protocol $(P, V)$ once the commitments have been removed.

**Lemma 8.1** (cf. [NV06]). *For every language $L \in$ **NP** and every $s(n) = 1/\mathrm{poly}(n)$, there are four polynomial-time algorithms $P, V$, $U$ and* Sim *and functions $\ell(n) = \mathrm{poly}(n), q(n) = O(\log(n)), u(n) = \mathrm{poly}(n)$ such that*

- *$P$ takes as input an instance $x$, an **NP**-witness $w$, and a sequence of coin tosses $r_p$ and outputs a $\ell$-tuple $(b_1, \cdots, b_\ell)$.*
- *$U$ takes as input an instance $x$, an $NP$-witness $w$, a sequence of coin tosses $r_p$, and a challenge $c \in \{0,1\}^q$ and outputs a sequence of indices $U \in [\ell]^q$.*
- *$V$ takes as input an instance $x$, a challenge $c \in \{0,1\}^q$, a sequence of indices $U \in [\ell]^q$, and a sequence of bits $(b_i)_{i \in U}$ and outputs a decision $\in \{\texttt{accept}, \texttt{reject}\}$.*
- Sim *takes as input an instance $x$, a challenge $c \in \{0,1\}^q$ and a sequence of coin tosses $r_S$ and outputs a sequence of indices $U \in [\ell]^q$ and a sequence of bits $(b_i)_{i \in U}$.*

**Perfect completeness** *If $x \in L$, then with probability 1 over $r_p$ and $c \overset{R}{\leftarrow} \{0,1\}^q$ it holds that $V$ accepts $\big(x, c, U(x, w, r_p, c), P(x, w; r_p)|_{U(x, r_p, c)}\big)$.*

**Soundness** *If $x \notin L$, then for every $(b_1, \cdots, b_\ell)$, with probability at most $s(n)$ over $c \overset{R}{\leftarrow} \{0,1\}^q$, there exists a sequence $U$ such that $V$ accepts $(x, c, U, (b_i)_{i \in U})$.*

**Zero-knowledge** *There exists a PPT* Sim *such that for every $x \in L$, and every $c \in \{0,1\}^q$ the distributions* $\mathrm{Sim}(x, c)$ *and $(U(x, r_p, c), P(x, w; r_p)|_U)$ (taken over the choice of $r_p$) are identical.*

## 8.2 Zero-Knowledge Arguments from a Single of 1-out-of-2-Binding Commitment

As a warm-up to the construction of zero-knowledge arguments based on the collection of commitments given by Theorem 3.1, we will give the construction based on a single $\binom{2}{1}$-binding commitment scheme.

**Theorem 8.2.** *Let $L \in \mathbf{NP}$ and $(S, R)$ be a 2-phase commitment scheme on security parameter $1^n$. There exists an interactive protocol $(P', V')$ such that:*

- *If $x \in L$ and $(S, R)$ is statistically hiding, then $(P', V')$ is statistical zero-knowledge*
- *If $x \notin L$ and $(S, R)$ is 1-out-of-2 computationally binding, then $(P', V')$ is computationally sound with soundness error $s'(|x|) = 1/\text{poly}(|x|)$.*

The new protocol $(P', V')$ will consist of two sequential executions of the generic protocol $(P, V)$. The prover will use the first phase of $(S, R)$ in the first execution and the second phase of $(S, R)$ in the second execution. The soundness property will rely on the fact that for each commitment, at least one phase is binding (though it might be a different phase for each commitment). Intuitively, this $\binom{2}{1}$-binding property should ensure that the prover cannot cheat in both executions.

However two difficulties arise at this point. First, the prover only opens $u$ first phase commitments in the first execution, whereas we need $\ell$ second phase commitments in the second execution. Secondly, the prover only needs to break one first phase commitment to ruin the soundness property in the first execution and we cannot guarantee that the corresponding second phase commitment (known to be binding) will be opened by the prover in the second execution.

In order to manage these difficulties, in the first execution, we make the prover commit to each bit $b_i$ a total of $\ell^2$ times using the first phase of $(S, R)$. We denote these first phase commitments $z_{i,j}$ for $i \in [\ell]$ and $j \in [\ell^2]$. Hence the prover opens more than enough first phase commitments in the first execution so that the corresponding second phase commitments can be used in the second execution.

The protocol $(P', V')$ is zero knowledge when $x \in L$ because both phases of the commitment scheme $(S, R)$ are statistically hiding and the generic protocol is zero knowledge when the commitment scheme used is hiding.

Let $x \notin L$. Suppose that in the first execution of $(P, V)$, there exists a first-phase commitment $z_{i^*,j^*}$ such that $z_{i^*,j^*}$ can be opened successfully as both 0 and 1 with first-phase transcripts not in $\mathcal{B}$ (hence the corresponding second phase commitments are not guaranteed to be binding). If this is the case, we can build an adversary breaking the computational binding property of $(S, R)$ by guessing which first phase commitment $z_{i,j}$ satisfies this property.

Hence, we can assume that each first-phase commitment $z_{i,j}$ has at most one "proper" decommitment value $b^*_{i,j}$. Let us consider the soundness property for the first execution of $(P, V)$. Consider the sequence $(b^*_1, \cdots, b^*_\ell)$ where $b^*_i$ is the majority of $b^*_{i,j}$ (over $j \in [\ell^2]$). By soundness of the generic protocol, the verifier would reject if the prover opens consistently with $(b^*_1, \cdots, b^*_\ell)$. Thus there must be an index $i^*$ such that the prover opens inconsistently with $b_{i^*}$, i.e. at least half of the commitments $\{z_{i^*,1}, \cdots, z_{i^*,\ell^2}\}$ are not opened properly and the second phases of these commitments will be statistically binding.

Before the second execution starts, the verifier chooses a random correspondence between the first phase commitments opened in the first execution and the second phase commitments to be used in the second execution. This random "shuffling" guarantees that if the prover cheats in the first execution by opening at least half of the commitments $\{z_{i^*,1}, \cdots, z_{i^*,\ell^2}\}$ improperly, then with high probability *every* bit $b'_i$ committed to in the second execution of $(P, V)$ will have at least one binding second phase commitment. Then, by soundness of the generic protocol, the verifier rejects in the second execution.

The details of the construction and proof of Theorem 8.2 follow.

**Zero-knowledge protocol from a single 1-out-of-2 computationally binding commitment**

**First execution of** $(P, V)$**:** $P'$ and $V'$ simulate a first execution of the generic protocol $(P, V)(x)$ with soundness parameter $s(n) = 1/4n$ using the first commitment phase and first reveal phase of $(S, R)$.

1. The prover generates a sequence $(b_1, \cdots, b_\ell)$ as in the generic protocol.
2. For each $i \in \{1, \cdots, \ell\}$, the prover commits to $b_i$ a total of $\ell^2$ times by running the first commitment phase. We refer to these commitments to $b_i$ as a "block" $B_i = \{z_{i,1}, \cdots, z_{i,\ell^2}\}$, that is $z_{i,j} = (S_c^1(b_i; r_{i,j}), R_c^1$ where the $r_{i,j}$ are uniform and independent coin tosses for $S_c^1$.
3. $V'$ sends a first challenge $c \in \{0, 1\}^q$ as in the generic protocol.
4. $P'$ computes a sequence of indices $U \in [\ell]^q$ corresponding to the challenge $c$ as in the generic protocol.
5. For each index $i \in U$, $P'$ sends $b_i$ and opens the $\ell^2$ commitments to $b_i$ in the block $B_i$ by running the first reveal phase $(S_r^1(r_{i,j}), R_r^1)(z_{i,j}, b_i)$ for each $j \in [\ell^2]$.
6. $V'$ rejects if the verifier $V$ rejects in the generic protocol or if the receiver $R_r^1$ rejects in the reveal phase.

**Second execution of** $(P, V)$**:** $P'$ and $V'$ simulate a second execution of the original protocol $(P, V)(x)$ using the second commitment phase and second reveal phase of $(S, R)$. To do so, the verifier will choose a random correspondence between the first phase commitments opened in the first execution and the second phase commitments to be used in the second execution.

1. For each $i \in U$, the verifier sends a random bijection $\phi_i : \{1, \cdots, \ell^2\} \mapsto [\ell] \times [\ell], j \to (p, q)$ (This is the correspondence between the first phase commitments already opened and the second phase commitments to be used).
2. The prover generates a sequence $(b'_1, \cdots, b'_\ell)$ as in the generic protocol.
3. For each $p \in \{1, \cdots, \ell\}$, the prover commits to $b'_p$ a total of $u\ell$ times by running the second commitment phase. More specifically the commitments to $b'_p$ are composed of $u$ blocks of size $\ell$ denoted by $B_p^i = \{z_{p,1}^i, \cdots, z_{p,q}^i, \cdots, z_{p,\ell}^i\}$ (for $i \in U$) where
$$z_{p,q}^i = (S_c^2(b'_p; r_{i,(\phi_i)^{-1}(p,q)}), R_c^2)(z_{i,(\phi_i)^{-1}(p,q)})$$
4. $V'$ sends a second challenge $c' \in \{0, 1\}^q$ as in the generic protocol.
5. $P'$ computes a sequence of indices $U' \subseteq [\ell]^q$ corresponding to the challenge $c'$ as in the generic protocol.
6. For each $p \in U'$, $P'$ sends $b'_p$ and opens the $\ell$ commitments to $b'_p$ in each block $B_p^i$ (for each $i \in U$) by running $(S_r^2(r_{i,(\phi_i)^{-1}(p,q)}), R_r^2)(z_{p,q}^i, b'_p)$ for $q \in [\ell]$.
7. $V'$ rejects if the verifier $V$ rejects in the generic protocol or if the receiver $R_r^2$ rejects in the reveal phase.

**Lemma 8.3** (Soundness property). *If $(S, R)$ is computationally 1-out-of-2 binding, then $(P', V')$ has soundness error $s'(n) = 1/n$.*

*Proof.* We write $|x| = n$. Recall that $s(n) = 1/4n$ is the soundness error of the generic protocol and we set $\delta = 1/\text{poly}(n)$ for some polynomial to be determined below.

For each prefix $\tau$ of a protocol transcript and each first-phase commitment $z_{i,j}$ in $\tau$ that has completed but has not been opened yet, we define (for $b \in \{0, 1\}$):

$$\begin{aligned} p_{\tau,z_{i,j},b} \quad = \quad & \Pr_T[z_{i,j} \text{ is opened to } b \text{ successfully with a transcript } T \notin \mathcal{B} \\ & \quad \wedge \ \text{corresponding second-phase opened successfully} | T \text{ begins with } \tau] \end{aligned}$$

where the probability is over transcripts $T$ of $(P^\star, V')$.

**Case 1**: with probability at least $\delta$ over $T$, there exists a prefix $\tau$ containing a first-phase commitment $z_{i*,j*}$ $(i* \in [\ell], j* \in [\ell^2])$ such that $p_{\tau,z_{i*,j*},0} \geq \delta$ and $p_{\tau,z_{i*,j*},1} \geq \delta$.

If this is the case, then we will build an adversary $S^\star$ breaking the computational binding property of $(S, R)$ by guessing which first phase commitment $z_{i,j}$ satisfies this property and running two parallel executions of $(P^\star, V')$ to break the binding property. The PPT adversary $S^\star$ does as follows:

1. $S^\star$ guesses which first-phase commitment $z_{i,j}$ corresponds to the commitment $z_{i*,j*}$ of Case 1. Note that with probability at least $\delta$ over transcripts $T \leftarrow \langle P^\star, V' \rangle$, $S^\star$'s guess is successful with probability at least $1/\ell^3$.

2. $S^\star$ executes the protocol $(P^\star, V')$ by simulating $P^\star$ and $V'$ on its own for all but one of the $\ell^3$ commitment phases and interacts with $R$ in the guessed first phase commitment $z_{i*,j*}$. Hence $S^\star$ generates blocks of first phase commitments $B_i = \{z_{i,1}, \cdots, z_{i,\ell^2}$ for $i \in [\ell], j \in [\ell^2]$. This constitutes a prefix $\tau$ of a protocol transcript.

3. $S^\star$ generates two valid transcripts $T, T' \leftarrow \langle P^\star, V' \rangle$ starting with the prefix $\tau$ (this is possible because $V'$ is public-coin). If $S^\star$'s guess was successful, then by definition of Case 1, we have:
   - the probability that $z_{i*,j*}$ is opened to 0 successfully with $T \notin \mathcal{B}$ and the corresponding second phase commitment is opened successfully is $\geq \delta$.
   - the probability that $z_{i*,j*}$ is opened to 1 successfully with $T' \notin \mathcal{B}$ and the corresponding second phase commitment is opened successfully is $\geq \delta$.

If we take $\delta = 1/\text{poly}(n)$, then the adversary $S^\star$ breaks the computational binding property of $(S, R)$ with probability at least $\left(\frac{\delta}{\ell}\right)^3 = 1/\text{poly}(n)$ which is nonnegligible. We have reached a contradiction hence Case 1 does not occur.

**Case 2**: with probability at least $1 - \delta$ over $T$, it holds that for every prefix $\tau$ of $T$ and for every first phase commitment $z_{i,j}$ in $\tau$, there is at most one value $b^\star_{i,j} \in \{0, 1\}$ such that $p_{\tau,z_{i,j},b^\star_{i,j}} \geq \delta$.

We say that $z_{i,j}$ *is opened properly* if $z_{i,j}$ is opened to the value $b^\star_{i,j}$ specified above.

Let $x \notin L$ and assume that the verifier $V'$ accepts in the interaction $(P^\star, V')(x)$ with probability $\eta$. By definition of Case 2, with probability at least $1 - \delta$ over the transcripts of Steps 1 and 2 of the first execution of $(P, V)$, each $z_{i,j}$ has at most one proper decommitment value $b^\star_{i,j}$. Consider the sequence $(b^\star_1, \cdots, b^\star_\ell)$ where $b^\star_i$ is the majority of $b^\star_{i,j}$ (over $j \in [\ell^2]$). By soundness of the generic protocol, the verifier accepts with probability at most $s(n)$ if the prover opens consistently with $(b^\star_1, \cdots, b^\star_\ell)$. Thus, except with probability $s(n)$, there must be an index $i^\star$ such that the prover opens inconsistently with $b^\star_{i^\star}$, i.e. at least half of the first phase commitments in the block $B_{i^\star}$ are not opened properly. Since $p_{\tau, z_{i,j}, \overline{b^\star_{i,j}}} < \delta$, with probability at least $1-\delta$, each corresponding second phase will be statistically binding or it will not be opened successfully.

Let $p \in [\ell]$. We will upper bound the probability that the block $B^{i^\star}_p$ has no binding second phase commitment. Recall that the block $B^{i^\star}_p$ contains $\ell$ second phase commitments and these are a random $\ell$-subset of $B_{i^\star}$ over the verifier's choice of $\phi_{i^\star}$. The probability that all of the corresponding first phase commitments were opened properly is at most $\binom{\ell^2/2}{\ell}/\binom{\ell^2}{\ell} \leq 2^{-\ell} = \text{neg}(n)$. Hence the probability that the block $B^{i^\star}_p$ contains no binding second phase commitment is at most $\text{neg}(n)+\delta\cdot\ell$ (without loss of generality, we will ignore the probability that the block $B^{i^\star}_p$ contains no binding second phase but the prover fails to complete the second reveal phase since it would only increase the soundness error). By a union bound, the probability that there exists an index $p \in [\ell]$ such that the block $B^{i^\star}_p$ has no binding second phase commitment is at most $\delta \cdot \ell^2 + \text{neg}(n)$.

In case every bit $b'_p$ in the second execution is statistically binding, then by soundness of the generic protocol $V'$ accepts in the second execution with probability at most $s(n)$.

$$\begin{aligned}
&\Pr[V' \text{ accepts in both executions}]\\
&\quad\leq \Pr_T[\exists i,j,\ z_{i,j} \text{ has more than one proper decommitment value}]\\
&\qquad + \Pr[V' \text{ accepts in 1st execution AND every } z_{i,j} \text{ is opened properly}]\\
&\qquad + \Pr[\text{some } z_{i,j} \text{ opened improperly AND } \exists p \in [\ell] \text{ such that } b'_p \text{ is not binding}]\\
&\qquad + \Pr[V' \text{ accepts in 2nd execution AND } \forall p \in [\ell],\, b'_p \text{ is binding}]\\
&\quad\leq \delta + s(n) + (\delta\ell^2 + \text{neg}(n)) + s(n) = \delta + 2s(n) + \delta\ell^2 + \text{neg}(n) < \frac{1}{n}
\end{aligned}$$

for $\delta = \frac{1}{4n(\ell^2+1)}$. □

**Lemma 8.4** (Zero knowledge property)**.** *If $x \in L$ and $(S, R)$ is statistically hiding, then Protocol 8.2 is statistical zero knowledge.*

*Proof.* Let $x \in L$ and $w$ be a corresponding **NP**-witness. The interaction $(P', V')$ consists of two sequential executions of the generic protocol $(P, V)$ such that the two executions are related by a collection of bijections $\{\phi_i\}_{i \in U}$.

The interaction $(P'(w), V')(x)$ produces a distribution of the form

$$((S^1_c(P^1(x, w; r^1_p)), R^1_c), c, P^1(x, w; r^1_p)|_{U^1(x,w,r_p,c)},$$

$$\{\phi_i\}, (S^2_c(P^2(x, w; r^2_p)), R^2_c), c', P^2(x, w; r^2_p)|_{U^2(x,w,r^2_p,c')})$$

where:

- $P^i(x, w; r^i_p)$ corresponds to the $\ell$-tuple $(b_1, \cdots, b^\ell)$ output by the prover P in the $i$th execution of the generic protocol (for $i \in \{1, 2\}$).

- $(S_c^i(P^i(x, w; r_p^i)), R_c^i)$ corresponds to $i$th phase commitments to the values $P^i(x, w; r_p)$
- $c, c'$ denote the challenges sent by the possibly cheating verifier $V^\star$ (that depend on the previous messages of the generic protocol).

To simulate the verifier's view in the protocol $(P', V^\star)$ (even if $V^\star$ does not follow the prescribed protocol), the simulator will randomly guess which first challenge $c$ the cheating verifier will select and later check that the guess was successful by running $V^\star$; if the guess was not successful, the simulator will try again. Intuitively, with polynomially many trials, the simulator will succeed in guessing the verifier's first challenge $c$ and in simulating the verifier's view of the first execution. The simulator will then proceed to the second execution by randomly guessing which second challenge $c'$ the cheating verifier will select.

**Simulator** Sim **for** $(P', V')$

**Inputs:** an instance $x$ and a cheating verifier algorithm $V^\star$ (deterministic wlog)

**First execution of the generic protocol**

1. Uniformly select a challenge $c \leftarrow \{0,1\}^q$
2. Run the simulator Sim for the generic protocol on input $(x, c)$ to obtain a sequence of indices $U \in [\ell]^q$ and commit to a sequence of bits $(b_1, \cdots, b_\ell)$ where $b_i$ is determined by the challenge $c$ if $i \in U$, 0 if $i \notin U$.
3. Run the first commitment phase of $(S, R)$ to obtain the blocks $\{B_i\}$ that are first phase commitments to the values $(b_i)_{i \in [\ell]}$
4. Run $V^\star(\{B_i\})$ to determine which challenge $c^\star$ would be sent if it had received the above first-phase commitments
5. If $c^\star \neq c$, go back to the beginning of the first execution (for up to $n \cdot 2^q = \text{poly}(n)$ trials). Otherwise, set $\tau = (\{B_i\}_{i \in [\ell]}, c, (b_i)_{i \in U})$.

**Second execution of the generic protocol**

1. Run $V^\star(\tau)$ to determine which bijections $\{\phi_i\}$ to use for the second phase commitments.
2. Uniformly select a challenge $c' \leftarrow \{0,1\}^q$
3. Run the simulator Sim for the generic protocol on input $(x, c')$ to obtain a sequence of indices $U' \in [\ell]^q$ and commit to a sequence of bits $(b'_1, \cdots, b'_\ell)$ where $b'_i$ is determined by the challenge $c'$ if $p \in U'$, 0 if $p \notin U'$.
4. Run the second commitment phase of $(S, R)$ to obtain the blocks $\{B_p^i\}$ that are second phase commitments to the values $(b'_p)_{p \in [\ell]}$
5. Run $V^\star(\tau, \{B_p^i\})$ to determine which challenge $c^\star$ would be sent if it had received the above second-phase commitments
6. If $c^\star \neq c'$, go back to the beginning of the second execution (for up to $n \cdot 2^q$ trials). Otherwise, output $(\tau, \{\phi_i\}, \{B_p^i\}_{i \in U, p \in [\ell]}, c', (b'_p)_{p \in U'})$.

Since the challenges $c, c'$ are taken from $\{0,1\}^q$ where $q = O(\log |x|)$ and the only information the verifier has about $c, c'$ when computing its challenges are the statistically hiding commitments in $\{B_i\}, \{B_p^i\}$, the simulator will guess each challenge successfully with noticeable probability $1/2^q - \text{neg}(n) = 1/\text{poly}(n)$. Thus, polynomially many trials will yield successful guesses $c$ and $c'$ with all but negligible probability.

By a hybrid argument, the distribution output by the simulator conditioned on successful guesses $c$ and $c'$ is statistically indistinguishable from the distribution output in a real interaction $(P', V')(x)$ since the commitments are statistically hiding in both phases.

□

## 8.3 Zero-Knowledge Arguments from a Collection of 1-out-of-2-Binding Commitments

We will now show how to construct a zero-knowledge argument based on a collection of commitments.

**Theorem 8.5.** *Let $L \in \mathbf{NP}$ and $\mathrm{Com}_1, \cdots, \mathrm{Com}_t$ be 2-phase commitment schemes (where $\mathrm{Com}_j = (S_j, R_j)$) on security parameter $1^n$. There exists an interactive protocol $(P', V')$ such that:*

- *If $x \in L$ and one of the commitments $\mathrm{Com}_1, \cdots, \mathrm{Com}_t$ is statistically hiding, then $(P', V')$ is statistical zero-knowledge*
- *If $x \notin L$ and all commitments $\mathrm{Com}_1, \cdots, \mathrm{Com}_t$ are 1-out-of-2 computationally binding, then $(P', V')$ is computationally sound with soundness error $s'(|x|) = 1/\mathrm{poly}(|x|)$.*

The new protocol $(P', V')$ will consist of $(t+1)$ sequential executions of the generic protocol $(P, V)$. In order to preserve the zero-knowledge property of the generic protocol, we need the prover's commitments in each execution to be statistically hiding. Since we are only guaranteed to have *at least one* statistically hiding commitment among $\mathrm{Com}_1, \cdots, \mathrm{Com}_t$ (when $x \in L$), we will use a secret sharing scheme for each bit that the prover must commit to in the generic protocol. Each bit $b_i$ will be shared using $t$ random values and the prover will commit to the $j$th share of $b_i$ using $\mathrm{Com}_j$. This will ensure that each unopened bit $b_i$ is hidden from the verifier and thus that the protocol is zero-knowledge.

The soundness property will be proven by showing that the prover's commitments are binding in *at least one* of the executions. Similarly to the warm-up case, in each execution of $(P, V)$, for every $j \in [t]$, the prover commits to the $j$th share *multiple* times using both the first and the second phases of $\mathrm{Com}_j$. For every $j \in [t]$, the verifier chooses a random correspondence between the first phase commitments using $\mathrm{Com}_j$ opened in the $r$th execution ($r \in \{1, \cdots, t+1\}$) and the second phase commitments using $\mathrm{Com}_j$ in the remaining $(t-r+1)$ executions. This random "shuffling" of the commitments using $\mathrm{Com}_j$ guarantees that if in the $r$th execution, the prover cheats by opening inconsistently and breaking some first phase commitments using $\mathrm{Com}_j$, then with high probability, for each of the remaining $(t-r+1)$ executions, for every $i \in [\ell]$, the $j$th share of $b_i$ will have at least one binding second-phase commitment. Hence *every bit $b_i$* committed to in the $(t+1)$st execution will be binding and by soundness of the generic protocol the verifier rejects in the $(t+1)$st execution (if it hasn't rejected in an earlier execution).

The details of the construction and proof of Theorem 8.5 follow. Similarly to the warm-up case of a single 1-out-of-2 computationally binding commitment, we will establish the soundness property by analyzing all first phase commitments (of the first $t$ executions of the generic protocol) at once.

**Zero-knowledge protocol from a collection of 1-out-of-2 computationally binding commitments**

**First execution of the generic protocol $(P, V)$:** $P'$ and $V'$ simulate a first execution of the generic protocol $(P, V)(x)$ with soundness parameter $s(n) = \frac{1}{3n(t+1)}$ using the first commitment phase and first reveal phase of each commitment scheme.

1. The prover generates a sequence $(b_1^1, \cdots, b_\ell^1)$ as in the original protocol.
2. For each $i^1 \in [\ell]$, the prover first computes *shares* of $b_{i^1}^1$, i.e. chooses random bits $b_{i^1,1}^1, \cdots, b_{i^1,t}^1$ such that $b_{i^1}^1 = b_{i^1,1}^1 \oplus \cdots \oplus b_{i^1,j}^1 \oplus \cdots \oplus b_{i^1,t}^1$.
3. For every $i^1 \in [\ell], j \in [t]$, the prover commits to the share $b_{i^1,j}^1$ a total of $(t\ell^2)$ times by running the first commitment phase of $\mathrm{Com}_j$. We refer to these commitments to $b_{i^1,j}^1$ as a "block" $B_{i^1,j}^1 = \{z_{i^1,j,1}^1, \cdots, z_{i^1,j,t\ell^2}^1\}$, that is $z_{i^1,j,k}^1 = ((S_j)_c^1(b_{i^1,j}^1; r_{i^1,j,k}^1), R_{j_c}^1)$ where the $r_{i^1,j,k}^1$ are uniform and independent coin tosses for $(S_j)_c^1$.
4. $V'$ sends a first challenge $c^1 \in \{0,1\}^q$ as in the generic protocol.
5. $P'$ computes a sequence of indices $U^1 \in [\ell]^q$ corresponding to the challenge $c^1$ as in the generic protocol.
6. For each index $i^1 \in U^1$, for each commitment scheme $\mathrm{Com}_j$, $P'$ sends $b_{i^1,j}^1$ and opens the $t\ell^2$ commitments in the block $B_{i^1,j}^1$ by running the first reveal phase of $\mathrm{Com}_j$, i.e. $((S_j)_r^1(r_{i^1,j,k}^1), (R_j)_r^1)(z_{i^1,j,k}^1, b_{i^1,j}^1)$ for $k \in [t\ell^2]$.
7. For every $i^1 \in U^1$, $V'$ computes the bit $b_{i^1}^1 = b_{i^1,1}^1 \oplus \cdots \oplus b_{i^1,j}^1 \oplus \cdots \oplus b_{i^1,t}^1$ and rejects if the verifier rejects in the generic protocol.

**Second execution of the generic protocol $(P, V)$:**

1. The prover generates a sequence $(b_1^2, \cdots, b_\ell^2)$ as in the original protocol.
2. For each $i^2 \in [\ell]$, the prover first computes *shares* of $b_{i^2}^2$, i.e. chooses random bits $b_{i^2,1}^2, \cdots, b_{i^2,t}^2$ such that $b_{i^2}^2 = b_{i^2,1}^2 \oplus \cdots \oplus b_{i^2,j}^2 \oplus \cdots \oplus b_{i^2,t}^2$.
3. For every $i^1 \in U^1$, every $j \in [t]$, the verifier sends a random bijection $\phi_{i^1,j}^1 : \{1, \cdots, t\ell^2\} \mapsto \{2, \cdots, t+1\} \times [\ell] \times [\ell]$. In other words, the verifier gives a correspondence between the first phase commitments that were opened in the first execution and the second phase commitments to be used in the subsequent executions $2, \cdots, t+1$. The block $B_{i^1,j}^1$ of size $t\ell^2$ is divided into subblocks of size $\ell$, one subblock for each subsequent $p$th execution (for $p \in \{2, \cdots, t+1\}$) and for each shared bit $b_{i^p}^p$ (for $i^p \in [\ell]$).
4. For every $i^2 \in [\ell]$, every $j \in [t]$, the prover commits to the share $b_{i^2,j}^2$ a total of $(u\ell + (t-1)\ell^2)$ times:

- $u\ell$ commitments are obtained as follows: for each $i^1 \in U^1$, we take the first phase commitments in $B^1_{i^1,j}$ corresponding to $(\phi^1_{i^1,j})^{-1}(\{2\} \times \{i^2\} \times [\ell])$ and run the second commitment phase of $\mathrm{Com}_j$. We refer to these second phase commitments to $b^2_{i^2,j}$ as a block $C^2_{i^2,j}$.
- $(t-1)\ell^2$ commitments are obtained by running the first commitment phase of $\mathrm{Com}_j$. We refer to these first phase commitments as a block $B^2_{i^2,j}$.

5. $V'$ sends a challenge $c^2 \in \{0,1\}^q$ as in the generic protocol.
6. The prover computes a sequence of indices $U^2 \in [\ell]^q$ corresponding to the challenge $c^2$ as in the generic protocol.
7. For each index $i^2 \in U^2$, for each commitment scheme $\mathrm{Com}_j$, $P'$ sends $b^2_{i^2,j}$ and opens the commitments to the share $b^2_{i^2,j}$ as follows:
   - the prover opens the $u\ell$ commitments in the block $C^2_{i^2,j}$ by running the second reveal phase of $\mathrm{Com}_j$
   - the prover opens the $(t-1)\ell^2$ commitments in the block $B^2_{i^2,j}$ by running the first reveal phase of $\mathrm{Com}_j$
8. For every $i^2 \in U^2$, $V'$ computes the bit $b^2_{i^2} = b^2_{i^2,1} \oplus \cdots \oplus b^2_{i^2,j} \oplus \cdots \oplus b^2_{i^2,t}$ and rejects if the verifier rejects in the generic protocol.

**$r$th execution of $(P,V)$ for $r \in \{3,\cdots,t+1\}$:**

1. The prover generates a sequence $(b^r_1,\cdots,b^r_\ell)$ as in the original protocol.
2. For each $i^r \in [\ell]$, the prover first computes *shares* of $b^r_{i^r}$, i.e. chooses random bits $b^r_{i^r,1},\cdots,b^r_{i^r,t}$ such that $b^r_{i^r} = b^r_{i^r,1} \oplus \cdots \oplus b^r_{i^r,j} \oplus \cdots \oplus b^r_{i^r,t}$.
3. For every $i^{r-1} \in U^{r-1}$, every $j \in [t]$, the verifier sends a random bijection $\phi^{r-1}_{i^{r-1},j} : \{1,\cdots,(t+1-r)\ell^2\} \mapsto \{r,\cdots,t+1\} \times [\ell] \times [\ell]$. In other words, the verifier gives a correspondence between the first phase commitments that were opened in the $(r-1)$th execution and the second phase commitments to be used in the subsequent executions $r,\cdots,t+1$. The block $B^{r-1}_{i^{r-1},j}$ of size $(t-(r-1))\ell^2$ is divided into subblocks of size $\ell$, one block for each $p$th execution (for $p \in \{r,\cdots,t+1\}$) and for each shared bit $b^p_{i^p}$ (for $i^p \in [\ell]$).
4. For every $i^r \in [\ell]$, every $j \in [t]$, the prover commits to the share $b^r_{i^r,j}$ a total of $((r-1)u\ell + (t-r+1)\ell^2)$ times:
   - $(r-1)u\ell$ commitments are obtained as follows. For every $m \in \{1,\cdots,r-1\}$, every $i^m \in U^m$, we take the commitments in $B^m_{i^m,j}$ corresponding to $(\phi^m_{i^m,j})^{-1}(\{r\} \times \{i^r\} \times [\ell])$ and run the second commitment phase of $\mathrm{Com}_j$. We refer to these second phase commitments to $b^r_{i^r,j}$ as a block $C^r_{i^r,j}$.
   - $(t-r+1)\ell^2$ commitments are obtained by running the first commitment phase of $\mathrm{Com}_j$. We refer to these first phase commitments as a block $B^r_{i^r,j}$.

5. $V'$ sends a challenge $c^r \in \{0,1\}^q$ as in the generic protocol.

6. The prover computes a sequence of indices $U^r \in [\ell]^q$ corresponding to the challenge $c^r$ as in the generic protocol.

7. For each index $i^r \in U^r$, for each commitment scheme $\mathrm{Com}_j$, $P'$ sends $b^r_{i^r,j}$ and opens the commitments to the share $b^r_{i^r,j}$ as follows:

    - the prover opens the $(r-1)u\ell$ commitments in the block $C^r_{i^r,j}$ by running the second reveal phase of $\mathrm{Com}_j$
    - the prover opens the $(t-r+1)\ell^2$ commitments in the block $B^r_{i^r,j}$ by running the first reveal phase of $\mathrm{Com}_j$

8. For every $i^r \in U^r$, $V'$ computes the bit $b^r_{i^r} = d^r_{i^r,1} \oplus \cdots \oplus b^r_{i^r,j} \oplus \cdots \oplus b^r_{i^r,t}$ and rejects if the verifier rejects in the generic protocol.

$V'$ accepts in the execution $(P',V')(x)$ if and only if $V'$ accepts in all $(t+1)$ executions of $(P,V)$.

We will analyze all first phase commitments (of the $t$ first executions of the generic protocol) at once and consider two cases:

**Case 1:** If there exists a first phase commitment $z^r_{i,j,k}$ in round $r$ using $\mathrm{Com}_j$ that the cheating prover $P^\star$ can open in two different ways, then we will build an adversary breaking the computational binding property of $\mathrm{Com}_j$ by guessing which first phase commitment $z^r_{i,j,k}$ satisfies this property.

**Case 2:** If for every first phase commitment $z^r_{i,j,k}$, the cheating prover $P^\star$ has low success probability in opening $z^r_{i,j,k}$ in two different ways then there exists a "proper" opening value $b^r_{i,j,k}$ that $z^r_{i,j,k}$ should be opened to. Then the analysis proceeds similarly to the case of 1-out-of-2 statistically binding commitments.

**Lemma 8.6** (Soundness property)**.** *If all commitments* $\mathrm{Com}_1, \cdots, \mathrm{Com}_t$ *are 1-out-of-2 computationally binding, then* $(P',V')$ *is sound with soundness error* $s'(n) = 1/n$.

*Proof Sketch.* We write $|x| = n$. Recall that $s(n) = \frac{1}{3n(t+1)}$ is the soundness error of the generic protocol and we set $\delta = 1/\mathrm{poly}(n)$ (or some polynomial to be determined below. Each first phase commitment used in the $(P',V')$ protocol will be denoted $z^r_{i,j,k}$ where $r \in [t+1]$ denotes the round, $i$ denotes that the commitment is to the share of bit $b_i$ in the generic protocol, $j$ is the commitment scheme used and $k$ is the index of the commitment within the block $B^r_{i,k}$.

For each prefix $\tau$ of a protocol transcript and each first-phase commitment $z^r_{i,j,k}$ in $\tau$ that has completed but has not been opened yet, we can define (for $b \in \{0,1\}$):

$$\begin{aligned} p_{\tau, z^r_{i,j,k}, b} \quad = \quad & \Pr_T[z^r_{i,j,k} \text{ is opened successfully to } b \text{ with } T \notin \mathcal{B}_j \\ & \quad \wedge \text{ corresponding second-phase opened successfully} | T \text{ begins with } \tau] \end{aligned}$$

where the probability is taken over transcripts $T$ of $(P^\star, V')$.

**Case 1**: with probability at least $\delta$ over $T$, there exists a prefix $\tau$ containing a first-phase commitment $z^r{*}_{i*,j*,k*}$ such that $p_{\tau,z^r{*}_{i*,j*,k*},0} \geq \delta$ and $p_{\tau,z^r{*}_{i*,j*,k*},1} \geq \delta$.

If this is the case, then we will build an adversary $S^\star$ breaking the computational binding property of $\mathrm{Com}_{j*}$ by guessing which first phase commitment $z^r{*}_{i*,j*,k*}$ satisfies this property and running two executions of $(P^\star, V')$ to break the binding property. The PPT adversary $S^\star$ does as follows:

1. $S^\star$ will guess which round $r$, which commitment scheme $\mathrm{Com}_j$ and which first-phase commitment $z^r_{i,j,k}$ corresponds to the commitment $z^{r*}_{i*,j*,k*}$ of Case 1. Note that with probability at least $\delta$ over transcripts $T \leftarrow \langle P^\star, V' \rangle$, $S^\star$'s guess is successful with probability at least $1/(t \cdot \ell \cdot t \cdot t\ell^2) = (1/t\ell)^3$.
2. $S^\star$ executes the protocol $(P^\star, V')$ by simulating $P^\star$ and $V'$ on its own for all but one of the first commitment phases and interacts with $R$ in the guessed first phase commitment $z^{r*}_{i*,j*,k*}$. This constitutes a prefix $\tau$ of a protocol transcript.
3. $S^\star$ generates two valid transcripts $T, T' \leftarrow \langle P^\star, V' \rangle$ starting with the prefix $\tau$. If $S^\star$'s guess was successful, then by definition of Case 1, we have:
   - the probability that $z^{r*}_{i*,j*,k*}$ is opened to 0 successfully with $T \notin \mathcal{B}_{j*}$ and the corresponding second phase commitment is opened successfully is at least $\delta$.
   - the probability that $z^{r*}_{i*,j*,k*}$ is opened to 1 successfully with $T \notin \mathcal{B}_{j*}$ and the corresponding second phase commitment is opened successfully is at least $\delta$.

If we take $\delta = 1/\mathrm{poly}(n)$, then the adversary $S^\star$ breaks the computational binding property of $\mathrm{Com}_{j*}$ with probability at least $\delta^3 \cdot (1/t\ell)^3 = 1/\mathrm{poly}(n)$ which is nonnegligible. We have reached a contradiction hence Case 1 does not occur.

**Case 2**: with probability at least $1 - \delta$ over $T$, it holds that for every prefix $\tau$ of $T$ and for every first phase commitment $z^r_{i,j,k}$ in $\tau$, there is at most one value $(b^r_{i,j,k})^\star \in \{0,1\}$ such that $p_{\tau,z^r_{i,j,k},(b^r_{i,j,k})^\star} \geq \delta$.

We say that $z^r_{i,j,k}$ *is opened properly* if $z^r_{i,j,k}$ is opened to the value $(b^r_{i,j,k})^\star$ specified above.

Let $x \notin L$ and assume that the verifier $V'$ accepts in the interaction $(P^\star, V')(x)$ with probability $s'(n)$. By definition of Case 2, with probability $1 - \delta$ over $T$, each $z^1_{i,j,k}$ has at most one proper decommitment value $(b^1_{i,j,k})^\star$. Consider the sequence $(b^{1\star}_{1,1}, \cdots, b^{1\star}_{\ell,1}, \cdots, b^{1\star}_{1,j}, \cdots, b^{1\star}_{\ell,j}, \cdots, b^{1\star}_{1,t}, \cdots, b^{1\star}_{\ell,t})$ where $b^{1\star}_{i,j}$ is the majority (over $k \in [t\ell^2]$) of $(b^1_{i,j,k})^\star$. By soundness of the generic protocol, the verifier would reject with probability $1 - s(n)$ if the prover opens consistently with $(b^{1\star}_{1,1}, \cdots, b^{1\star}_{\ell,1}, \cdots, b^{1\star}_{1,t}, \cdots, b^{1\star}_{\ell,t})$. Thus except with probability $s(n)$, there must be an index $i^{1\star} \in U^1$ and an index $j^\star \in [t]$ such that the prover opens inconsistently with $b^{1\star}_{i^{1\star},j^\star}$, i.e. at least half of the commitments in the block $B^1_{i^{1\star},j^\star}$ are not opened properly. Without loss of generality, we may assume that these first phase commitments use $\mathrm{Com}_1$, i.e. $j^\star = 1$. Recall that each of the corresponding second phase commitments will be statistically binding with probability at least $1 - \delta$.

Let us consider the second phase commitments using $\mathrm{Com}_1$ in the subsequent executions $2, \cdots, (t+1)$. For the $p$th execution ( $p \in \{2, \cdots, t+1\}$) and for the shared bit $b^p_{i^p}$ $(i^p \in [\ell])$, the probability that the $\ell$ first phase commitments of $B^1_{i^{\star 1},1}$ corresponding to $C^p_{i^p,1}$ were opened properly is at most $(1/2)^\ell = \mathrm{neg}(n)$. Hence the probability that the block $C^p_{i^p,1}$ contains no binding second phase commitment is at most $\mathrm{neg}(n) + \delta\ell$. By a union bound, the probability that there exists some execution $p$ and some shared bit $b^p_{i^p}$ for which the block $C^p_{i^p,1}$ contains no binding (second phase) commitment is at most $\delta\ell^2 t + \mathrm{neg}(n)$. This implies that with probability at least $1-(\delta+s(n)+\delta\ell^2 t+\mathrm{neg}(n))$, any shared bit committed to using $\mathrm{Com}_1$ in the executions $2, \cdots, t+1$ can be opened in at most one way.

By a similar reasoning, assume that the schemes $\mathrm{Com}_1, \cdots, \mathrm{Com}_{r-1}$ are binding in the $r$th execution. By definition of Case 2, each first phase commitment $z^r_{i,j,k}$ in the block $B^r_{i,j}$ has at most one proper decommitment value $(b^r_{i,j,k})^\star$ . Consider the sequence $(b^{r\star}_{1,1}, \cdots, b^{r\star}_{\ell}, \cdots, b^{r\star}_{1,j}, \cdots, b^{r\star}_{\ell,t})$ where $b^{r\star}_{i,j}$ is the majority (over $k \in [(t-r)\ell^2]$) of $(b^r_{i,j,k})^\star$. By soundness of the generic protocol, the verifier would reject with probability $1 - s(n)$ if the prover opens consistently with $(b^{r\star}_{1,1}, \cdots, b^{r\star}_{\ell}, \cdots, b^{r\star}_{1,j}, \cdots, b^{r\star}_{\ell,t})$. Thus there must be an index $i^{r\star}$ and an index $j^\star$ such that the prover opens inconsistently with $b^{r\star}_{i^{r\star},j^\star}$, i.e. at least half of the commitments in the block $B^r_{i^{r\star},j^\star}$ are not opened properly. We know that $j^\star \notin \{1, \cdots, r-1\}$ since we have assumed the blocks $C^r_{i,j}$ for $j \in \{1, \cdots, r-1\}$ contain a binding commitment. Hence $j^\star \in \{r, \cdots, t\}$ and without loss of generality, we may assume that these first phase commitments use $\mathrm{Com}_r$, i.e. $j^\star = r$. Recall that each of the second phase commitments will be statistically binding with probability at least $1-\delta$.

By reasoning similarly to above, the probability that there exists $p \in \{r+1, \cdots, t+1\}$ and a shared bit $b^p_{i^p}$ such that the block $C^p_{i^p,r}$ contains no binding second phase commitment is at most $\delta\ell^2 t+\mathrm{neg}(n)$. Hence with probability at least $1-(\delta+s(n)+\delta\ell^2 t+\mathrm{neg}(n))$, any shared bit committed to using $\mathrm{Com}_r$ in the executions $r+1, \cdots, t+1$ can be opened in at most one way.

$$
\begin{aligned}
&\Pr[V' \text{ accepts in all } (t+1) \text{ executions}] \\
&\quad\leq \Pr_T[\exists r, i, j, k\ z^r_{i,j,k} \text{ has more than one proper decommitment value}] \\
&\qquad + \Pr[\exists r \in [t],\ V' \text{ accepts } r\text{th execution AND } \forall i, j, k,\ z^r_{i,j,k} \text{ is opened properly}] \\
&\qquad + \Pr[\text{some } z^r_{i,j,k} \text{ opened improperly AND } \exists p \in \{2, \cdots, t+1\}, j \in [t] \\
&\qquad\qquad \text{such that } C^p_{i^p,j} \text{ has no binding commitment}] \\
&\qquad + \Pr[V' \text{ accepts in } (t+1)\text{th execution AND } \forall i, j,\ C^{t+1}_{i,j} \text{ has a binding commitment}] \\
&\quad\leq \delta + t\cdot s(n) + t\cdot(\delta\ell^2 t) + \epsilon + \mathrm{neg}(n) \\
&\quad\leq \delta + (t+1)s(n) + t^2\delta\ell^2 + \mathrm{neg}(n) < 1/n
\end{aligned}
$$

if we take $\delta = \frac{1}{3n(t^2\ell^2+1)}$. □

**Lemma 8.7** (Zero-knowledge property)**.** *If $x \in L$ and one of the commitments* $\mathrm{Com}_1, \cdots, \mathrm{Com}_t$ *is statistically hiding, then $(P', V')$ is statistical zero knowledge.*

*Proof Sketch.* Recall that each bit $b_i$ in each execution of the generic protocol is shared using $t$ random values $b_{i,1}, \cdots, b_{i,t}$ such that $b_i = \bigoplus_j b_{i,j}$ and the prover commits to the $j$th share of $b_i$ using one of the phases of $\mathrm{Com}_j$.

If at least one of the commitment schemes used is statistically (resp. computationally) hiding, then the secret sharing scheme ensures that each bit $b_i$ committed to in the generic protocol is hidden. Given a bit $\sigma$, we can define a new 2-phase commitment scheme $\mathrm{Com}(\sigma) = (C_1(\sigma_1), C_2(\sigma_2), \cdots, C_t(\sigma_t))$ where $\sigma = \bigoplus_i \sigma_i$ and the commitment scheme $C_j$ consists of both first phase commitments using $\mathrm{Com}_j$ and second phase commitments (coming from different first phase transcripts) using $\mathrm{Com}_j$. If one of the commitments $\mathrm{Com}_1, \cdots, \mathrm{Com}_t$, say $\mathrm{Com}_j$ is statistically hiding, then Com is statistically hiding since the commitments in the $j$th block are always hiding. We can then apply a reasoning identical to that in the case of a single 1-out-of-2 binding scheme, except that the simulator will need to simulate $(t+1)$ sequential executions of the generic protocol as opposed to 2 executions (the simulator will guess the challenge for the $i$th execution successfully with all but exponentially small probability with polynomially many trials and once the guess for the $i$th execution is successful, the simulator goes on to guess the challenge for the $(i+1)$ execution). □

## Acknowledgements

We thank Oded Goldreich, Alex Healy, Rafail Ostrovsky and Omer Reingold for helpful discussions.

# References

[AH91] William Aiello and Johan Håstad. Statistical zero-knowledge languages can be recognized in two rounds. *Journal of Computer and System Sciences*, 42(3):327–345, 1991.

[BBR88] Charles H. Bennett, Gilles Brassard, and Jean-Marc Robert. Privacy amplification by public discussion. *SIAM Journal on Computing*, 17(2):210–229, 1988. Special issue on cryptography.

[BCC88] Gilles Brassard, David Chaum, and Claude Crépeau. Minimum disclosure proofs of knowledge. *Journal of Computer and System Sciences*, 37(2):156–189, 1988.

[BCY91] Gilles Brassard, Claude Crépeau, and Moti Yung. Constant-round perfect zero-knowledge computationally convincing protocols. *Theoretical Computer Science*, 84(1, Algorithms Automat. Complexity Games):23–52, 1991. 16th International Colloquium on Automata, Languages, and Programming (Stresa, 1989).

[Bel02] Mihir Bellare. A note on negligible functions. *J. Cryptology*, 15(4):271–284, 2002.

[BHZ87] Ravi B. Boppana, Johan Håstad, and Stathis Zachos. Does co-NP have short interactive proofs? *Information Processing Letters*, 25(2):127–132, 1987.

[BIN97] Mihir Bellare, Russell Impagliazzo, and Moni Naor. Does parallel repetition lower the error in computationally sound protocols? In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, pages 374–383, 1997.

[BKK90] Joan Boyar, S. A. Kurtz, and Mark W. Krentel. A discrete logarithm implementation of perfect zero-knowledge blobs. *Journal of Cryptology*, 2(2):63–76, 1990.

[Blu87] Manuel Blum. How to prove a theorem so no one else can claim it. In *Proceedings of the International Congress of Mathematicians, (Berkeley, California, 1986)*, pages 1444–1451. American Mathematical Society, 1987.

[BLV04] Boaz Barak, Yehuda Lindell, and Salil Vadhan. Lower bounds for non-black-box zero knowledge. Technical Report TR04–083, Electronic Colloquium on Computational Complexity, September 2004. Extended abstract in *FOCS '04*.

[Dam87] Ivan Damgård. Collision free hash functions and public key signature schemes. In *EUROCRYPT*, pages 203–216, 1987.

[DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Transactions in Information Theory*, 22(6):644–654, 1976.

[Din06] Irit Dinur. The PCP theorem via gap amplification. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[DPP98] Ivan B. Damgård, Torben P. Pedersen, and Birgit Pfitzmann. Statistical secrecy and multibit commitments. *IEEE Transactions on Information Theory*, 44(3):1143–1151, 1998.

[For89] Lance Fortnow. The complexity of perfect zero-knowledge. *Advances in Computing Research: Randomness and Computation*, 5:327–343, 1989.

[GGL98] Oded Goldreich, Shafi Goldwasser, and Nathan Linial. Fault-tolerant computation in the full information model. *SIAM Journal on Computing*, 27(2):506–544, 1998.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, 1986.

[GK96] Oded Goldreich and Ariel Kahan. How to construct constant-round zero-knowledge proof systems for NP. *Journal of Cryptology*, 9(3):167–190, 1996.

[GMR88] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989.

[GMW91] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(1):691–729, 1991.

[Gol93] Oded Goldreich. A uniform-complexity treatment of encryption and zero-knowledge. *Journal of Cryptology*, 6(1):21–53, 1993.

[Gol01] Oded Goldreich. *Foundations of Cryptography: Basic Tools*. Cambridge University Press, 2001.

[HHK$^+$05] Iftach Haitner, Omer Horvitz, Jonathan Katz, Chiu-Yuen Koo, Ruggero Morselli, and Ronen Shaltiel. Reducing complexity assumptions for statistically-hiding commitment. In *Proceedings of the 24th Annual International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT '05)*, pages 58–77, 2005.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography. In *FOCS30*, pages 230–235, 1989.

[ILL89] Russell Impagliazzo, Leonid A. Levin, and Michael Luby. Pseudo-random generation from one-way functions. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 12–24, 1989.

[IR89] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 44–61, 1989.

[KSS00] Jeff Kahn, Michael Saks, and Cliff Smyth. A dual version of Reimer's inequality and a proof of Rudich's conjecture. In *15th Annual IEEE Conference on Computational Complexity (Florence, 2000)*, pages 98–103. IEEE Computer Soc., Los Alamitos, CA, 2000.

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *Journal of Cryptology*, 4(2):151–158, 1991.

[NOVY98] Moni Naor, Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Perfect zero-knowledge arguments for NP using any one-way permutation. *Journal of Cryptology*, 11(2):87–108, 1998. Preliminary version in *CRYPTO '92*.

[NV06] Minh-Huyen Nguyen and Salil Vadhan. Zero knowledge with efficient provers. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[NY89] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 33–43, 1989.

[Ost91] Rafail Ostrovsky. One-way functions, hard on average problems, and statistical zero-knowledge proofs. In *Proceedings of the 6th Annual Structure in Complexity Theory Conference*, pages 133–138, 1991.

[OVY93] Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Fair games against an all-powerful adversary. *AMS DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 155–169, 1993.

[OW93] Rafail Ostrovsky and Avi Wigderson. One-way functions are essential for non-trivial zero-knowledge. In *Proceedings of the 2nd Israel Symposium on Theory of Computing Systems*, pages 3–17, 1993.

[Rei05] Omer Reingold. Undirected ST-connectivity in log-space. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 376–385, 2005.

[Rom90] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing*, pages 387–394, 1990.

[Rud88] Steven Rudich. *Limits on the Provable Consequences of One-Way Functions*. PhD thesis, U.C. Berkeley, 1988.

[Sha49] Claude Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28(4):656–715, 1949.

[Sim98] Daniel Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *Proceedings of the 17th Annual International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT '98)*, pages 334–345, 1998.

[Vad99] Salil Vadhan. *A Study of Statistical Zero-Knowledge Proofs*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, August 1999.

# A Proofs of Collision Probability Lemmas

We prove the lemmas presented in Section 6.1.

**Lemma A.1** (Restatement of Lemma 6.3)**.** *For independent pairs of random variables $(X_1, Y_1), \ldots, (X_m, Y_m)$,*

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) = \prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|Y_i).$$

*Note that $X_i$ and $Y_i$ can be correlated, it is only required that the pair $(X_i, Y_i)$ be independent from the other tuples.*

*Proof of Lemma 6.3.* Since the $X_i$'s are independent, for all $y_1, \ldots, y_m$, we have

$$\mathrm{CP}((X_1, \ldots, X_m)|_{Y_1=y_1,\ldots,Y_m=y_m}) = \prod_{i=1}^{m} \mathrm{CP}(X_i|_{Y_i=y_i}).$$

This gives us

$$\begin{aligned}
&\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) \\
&= \mathop{\mathrm{E}}_{(y_1,\ldots,y_m)\leftarrow(Y_1,\ldots,Y_m)} \left[\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|_{Y_1=y_1,\ldots,Y_m=y_m})\right] \\
&= \mathop{\mathrm{E}}_{(y_1,\ldots,y_m)\leftarrow(Y_1,\ldots,Y_m)} \left[\prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|_{Y_i=y_i})\right] && \text{(by independence of } X_i\text{'s given } Y_1, \ldots, Y_m) \\
&= \prod_{i=1}^{m} \mathop{\mathrm{E}}_{y_i\leftarrow Y_i} \left[\mathrm{CP}^{1/2}(X_i|_{Y_i=y_i})\right] && \text{(by independence of } Y_i\text{'s)} \\
&= \prod_{i=1}^{m} \mathrm{CP}^{1/2}(X_i|Y_i).
\end{aligned}$$

□

**Lemma A.2** (Restatement of Lemma 6.4)**.** *Suppose random variables $(X_1, Y_1), \ldots, (X_m, Y_m)$ satisfy the following conditions for some values of $\alpha_1, \ldots, \alpha_m \in \mathbb{R}^+$ and all $i = 1, 2, \ldots, m$:*

1. *For any given $(y_1, \ldots, y_{i-1}) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_{i-1})$,*

$$\mathrm{CP}^{1/2}(X_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}} \mid Y_i|_{Y_1=y_1,\ldots,Y_{i-1}=y_{i-1}}) \leq \alpha_i.$$

2. *For any given $(y_1, \ldots, y_i) \in \mathrm{Supp}(Y_1, Y_2, \ldots, Y_i)$, even if we condition on $Y_1 = y_1, \ldots, Y_i = y_i$, the $i + 1$ random variables $X_1, X_2, \ldots, X_i, Y_{i+1}$ are independent.*

*Then,*

$$\mathrm{CP}^{1/2}((X_1, \ldots, X_m)|(Y_1, \ldots, Y_m)) \leq \prod_{i=1}^{m} \alpha_i.$$

*Proof of Lemma 6.4.* By induction, it suffices to prove

$$\mathrm{CP}^{1/2}((X_1,\dots,X_m)|(Y_1,\dots,Y_m)) \le \alpha_m \cdot \mathrm{CP}^{1/2}\left((X_1,\dots,X_{m-1})|(Y_1,\dots,Y_{m-1})\right), \tag{6}$$

and then by iteratively expanding $\mathrm{CP}^{1/2}\left((X_1,\dots,X_{m-1})|(Y_1,\dots,Y_{m-1})\right)$ in terms of $\alpha_j$'s, we get our result. To simplify notation, we write $X'_m = X_m|_{Y_1=y_1,\dots,Y_{m-1}=y_{m-1}}$ and $Y'_m = Y_m|_{Y_1=y_1,\dots,Y_{m-1}=y_{m-1}}$ when $y_1,\dots,y_{m-1}$ are clear from context. We prove (6) as follows:

$$\mathrm{CP}^{1/2}((X_1,\dots,X_m)|(Y_1,\dots,Y_m)) \tag{7}$$

$$= \mathop{\mathrm{E}}_{(y_1,\dots,y_m)\leftarrow(Y_1,\dots,Y_m)}\left[\mathrm{CP}^{1/2}((X_1,\dots,X_m)|_{Y_1=y_1,\dots,Y_m=y_m})\right] \tag{8}$$

$$= \mathop{\mathrm{E}}_{(y_1,\dots,y_{m-1})\leftarrow(Y_1,\dots,Y_{m-1})}\left[\mathop{\mathrm{E}}_{y_m\leftarrow Y'_m}\left[\mathrm{CP}^{1/2}((X_1,\dots,X_m)|_{Y_1=y_1,\dots,Y_m=y_m})\right]\right] \tag{9}$$

$$= \mathop{\mathrm{E}}_{(y_1,\dots,y_{m-1})}\left[\mathop{\mathrm{E}}_{y_m\leftarrow Y'_m}\left[\mathrm{CP}^{1/2}((X_1,\dots,X_{m-1})|_{Y_1=y_1,\dots,Y_m=y_m})\cdot \mathrm{CP}^{1/2}(X_m|_{Y_1=y_1,\dots,Y_m=y_m})\right]\right] \tag{10}$$

$$= \mathop{\mathrm{E}}_{(y_1,\dots,y_{m-1})}\left[\mathrm{CP}^{1/2}((X_1,\dots,X_{m-1})|_{Y_1=y_1,\dots,Y_{m-1}=y_{m-1}})\cdot \mathop{\mathrm{E}}_{y_m\leftarrow Y'_m}\left[\mathrm{CP}^{1/2}(X_m|_{Y_1=y_1,\dots,Y_m=y_m})\right]\right] \tag{11}$$

$$= \mathop{\mathrm{E}}_{(y_1,\dots,y_{m-1})}\left[\mathrm{CP}^{1/2}((X_1,\dots,X_{m-1})|_{Y_1=y_1,\dots,Y_{m-1}=y_{m-1}})\cdot \mathrm{CP}^{1/2}(X'_m|Y'_m)\right] \tag{12}$$

$$\le \alpha_m \cdot \mathop{\mathrm{E}}_{(Y_1,\dots,Y_{m-1})}\left[\mathrm{CP}^{1/2}((X_1,\dots,X_{m-1})|_{Y_1=y_1,\dots,Y_{m-1}=y_{m-1}})\right] \tag{13}$$

$$\le \alpha_m \cdot \mathrm{CP}^{1/2}\left((X_1,\dots,X_{m-1})|(Y_1,\dots,Y_{m-1})\right) \tag{14}$$

Equation (10) follows because $X_1,\dots,X_m$ conditioned on $Y_1=y_1,\dots,Y_m=y_m$ are independent. Equation (11) follows because $X_1,\dots,X_{m-1},Y_m$ conditioned on $Y_1=y_1,\dots,Y_{m-1}=y_{m-1}$ are also are independent. Finally, Equation (13) follows from the assumption that $\mathrm{CP}^{1/2}(X'_m|Y'_m) = \mathrm{CP}^{1/2}(X_m|_{Y_1=y_1,\dots,Y_{m-1}=y_{m-1}} \mid Y_m|_{Y_1=y_1,\dots,Y_{m-1}=y_{m-1}}) \le \alpha_m$. □

**Lemma A.3** (Restatement of Lemma 6.5)**.** *Let $(X,Y)$ be any (possibly correlated) pair of random variables, and let $H \leftarrow \mathcal{H}$ be chosen randomly (and independently from $(X,Y)$) from a family of pairwise-independent hash functions with a range of $\{0,1\}^\alpha$. Then,*

$$\mathrm{CP}^{1/2}((\mathcal{H},\mathcal{H}(X))|Y) \le \mathrm{CP}^{1/2}(\mathcal{H})\cdot(\mathrm{CP}^{1/2}(X|Y)+\sqrt{2^{-\alpha}}).$$

*Proof of Lemma 6.5.*

$$\begin{aligned}
&\mathrm{CP}^{1/2}(\mathcal{H},\mathcal{H}(X)|Y)\\
&= \mathop{\mathrm{E}}_{y \leftarrow Y}\left[\mathrm{CP}^{1/2}(\mathcal{H},\mathcal{H}(X)|_{Y=y})\right]\\
&= \mathop{\mathrm{E}}_{y \leftarrow Y}\left[\mathrm{CP}^{1/2}(\mathcal{H},\mathcal{H}(X)|_{Y=y})\right]\\
&\leq \mathop{\mathrm{E}}_{y \leftarrow Y}\left[\mathrm{CP}^{1/2}(\mathcal{H}) \cdot \sqrt{\mathrm{CP}(X|_{Y=y}) + 2^{-\alpha}}\right] \qquad \text{(since } \mathrm{CP}(\mathcal{H},\mathcal{H}(Z)) \leq \mathrm{CP}(\mathcal{H}) \cdot (\mathrm{CP}(Z) + 2^{-\alpha}))\\
&\leq \mathop{\mathrm{E}}_{y \leftarrow Y}\left[\mathrm{CP}^{1/2}(\mathcal{H}) \cdot \left(\mathrm{CP}^{1/2}(X|_{Y=y}) + \sqrt{2^{-\alpha}}\right)\right]\\
&= \mathrm{CP}^{1/2}(\mathcal{H}) \cdot \left(\left(\mathop{\mathrm{E}}_{y \leftarrow Y}\left[\mathrm{CP}^{1/2}(X|_{Y=y})\right]\right) + \sqrt{2^{-\alpha}}\right)\\
&= \mathrm{CP}^{1/2}(\mathcal{H}) \cdot (\mathrm{CP}^{1/2}(X|Y) + \sqrt{2^{-\alpha}}),
\end{aligned}$$

hence our result. □

**Corollary A.4** (Restatement of Corollary 6.6)**.** *Let $(X, Y)$ be any (possibly correlated) pair of random variables, and let $H \leftarrow \mathcal{H}$ be chosen randomly (and independently from $(X, Y)$) from a family of pairwise-independent hash functions with a range of $\{0, 1\}^{\alpha}$. Suppose the hash functions from $\mathcal{H}$ are represented by $(q - \alpha)$-bit strings and $\mathrm{CP}^{1/2}(X|Y) \leq \sqrt{2^{-(\alpha+3)}}$. Then,*

$$\mathrm{CP}^{1/2}((\mathcal{H},\mathcal{H}(X))|Y) \leq \sqrt{2^{-(q-1)}}.$$

*Proof of Corollary 6.6.* Since $|h| = q - \alpha$, we have $\mathrm{CP}(H) = 2^{-(q-\alpha)}$. Therefore, by Lemma 6.5,

$$\begin{aligned}
\mathrm{CP}^{1/2}(H, H(X)|Y) &\leq \mathrm{CP}^{1/2}(H) \cdot (\mathrm{CP}^{1/2}(X|Y) + \sqrt{2^{-\alpha}})\\
&\leq \sqrt{2^{-(q-\alpha)}} \cdot \left(\sqrt{\frac{2^{-\alpha}}{8}} + \sqrt{2^{-\alpha}}\right)\\
&< \sqrt{2^{-(q-\alpha)}} \cdot \left(\sqrt{2^{-\alpha}} \cdot \sqrt{2}\right)\\
&= \sqrt{2^{-(q-1)}}.
\end{aligned}$$

□

**Lemma A.5** (Restatement of Lemma 6.7)**.** *For any triple of (possibly correlated) random variables $X$, $Y$ and $Z$,*

$$\mathrm{CP}^{1/2}(X|Y) \leq \mathrm{CP}^{1/2}(X|(Y, Z)) \leq \sqrt{|\mathrm{Supp}(Z)|} \cdot \mathrm{CP}^{1/2}(X|Y).$$

*Proof of Lemma 6.7.* For the upper bound,

$$\begin{aligned}
\mathrm{CP}^{1/2}(X|(Y,Z)) &= \mathop{\mathrm{E}}_{(y,z)\leftarrow(Y,Z)}\left[\mathrm{CP}^{1/2}(X|_{(Y,Z)=(y,z)})\right] \\
&= \mathop{\mathrm{E}}_{y\leftarrow Y}\left[\sum_z \Pr[Z=z|Y=y]\cdot \mathrm{CP}^{1/2}(X|_{(Y,Z)=(y,z)})\right] \\
&\leq \mathop{\mathrm{E}}_{y\leftarrow Y}\left[\sqrt{|\mathrm{Supp}(Z)|}\cdot\sqrt{\sum_z \left(\Pr\left[Z=z|Y=y\right]\right)^2\cdot \mathrm{CP}(X|_{(Y,Z)=(y,z)})}\right] \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(by Cauchy-Schwartz)} \\
&= \sum_y \Pr\left[Y=y\right]\cdot\sqrt{|\mathrm{Supp}(Z)|}\cdot \mathrm{CP}^{1/2}(X|_{Y=y}) \\
&= \sqrt{|\mathrm{Supp}(Z)|}\cdot \mathop{\mathrm{E}}_{y\leftarrow Y}\left[\mathrm{CP}^{1/2}(X|_{Y=y})\right] \\
&= \sqrt{|\mathrm{Supp}(Z)|}\cdot \mathrm{CP}^{1/2}(X|Y).
\end{aligned}$$

For the lower bound, consider the following: For each $y \in \mathrm{Supp}(Y)$ and $z \in \mathrm{Supp}(Z)$, let $v_{y,z}$ be the vector $(\Pr[X=x \wedge Z=z|Y=y])_{x\in\mathrm{Supp}(X)}$. Then,

$$\begin{aligned}
\mathrm{CP}^{1/2}(X|_{Y=y}) &= \left\|\sum_z v_{y,z}\right\|_2 \\
&\leq \sum_z \|v_{y,z}\|_2 && \text{(triangle inequality)} \\
&= \mathrm{CP}^{1/2}\left((X|_{Y=y})\,|\,(Z|_{Y=y})\right).
\end{aligned}$$

Taking expectations over $Y$ for both sides yield our result. □

**Lemma A.6** (Restatement of Lemma 6.8)**.** *Let $\mathcal{H} = \{h\colon \{0,1\}^n \to \{0,1\}^\alpha\}$ be a family of pairwise-independent hash functions, and let $q-\alpha$ be the description of length of each element in $\mathcal{H}$. If $\mathrm{CP}(X) \leq \varepsilon^2\cdot 2^{-\alpha}$, then $\Delta((\mathcal{H},\mathcal{H}(X)),U_q)\leq\varepsilon$.*

*Proof of Lemma 6.8.* Let $D = 2^{q-\alpha}$ and $L = 2^{\alpha}$. We bound the statistical distance of $(\mathcal{H}, \mathcal{H}(X))$ from uniform as follows:

$$\begin{aligned}
\Delta((\mathcal{H}, \mathcal{H}(X)), U_q) &= \frac{1}{2} \left|(\mathcal{H}, \mathcal{H}(X)) - U_q\right|_1 \\
&\leq \frac{\sqrt{DL}}{2} \left\|(\mathcal{H}, \mathcal{H}(X)) - U_q\right\|_2 \\
&\leq \frac{\sqrt{DL}}{2} \cdot \sqrt{\mathrm{CP}(\mathcal{H}, \mathcal{H}(X)) - 2^{-q}} \\
&\leq \frac{\sqrt{DL}}{2} \cdot \sqrt{\frac{1}{D}\left(\mathrm{CP}(X) + \frac{1}{L}\right) + \frac{1}{DL}} \\
&= \frac{\sqrt{\mathrm{CP}(X) \cdot L}}{2} \\
&\leq \frac{\varepsilon}{2} \\
&\leq \varepsilon.
\end{aligned}$$

□

# B Proofs of NOVY IH Hiding and Binding Properties

In order to prove Theorem 4.4, we need to show that the Interactive Hashing Scheme $(S_{\mathrm{IH}}, R_{\mathrm{IH}})$, namely Protocol 4.5, satisfies the hiding and binding properties of Definition 4.2. The correctness of Protocol 4.5 is easy to see. The hiding and binding properties are captured by Lemmas B.1 and B.2, respectively.

## B.1 Hiding Property

**Lemma B.1** (perfect hiding)**.** *Protocol 4.5 is perfectly hiding in the sense of the Definition 4.2.*

The proofs presented in this section and the next are very similar in nature to those in [NOVY98], with additional analysis needed to handle interactive hashing for multiple outputs.

*Proof.* The view of any $R^*$ will be the hash functions $h_0, h_1, \cdots, h_{q-k-1}$ together with $S$'s respond $c_0, c_1, \ldots, c_{q-k-1}$. Given these values, we show that there are $2^{q-k}$ possible $y$'s that would make $S(y)$ respond to $c_0, c_1, \ldots, c_{q-k-1}$ (given queries $h_0, h_1, \cdots, h_{q-k-1}$ from $R^*$).

Consider the matrix $H = (h_0, h_1, \cdots, h_{q-k-1})$ whose rows are the $h_i$'s, vector $c = (c_0, c_1, \ldots, c_{q-k-1})$, and the equation $Hy = c$. Since $h_i$ is of the form $0^i 1\{0,1\}^{q-i-1}$, the first $q-k$ columns of the matrix are linearly independent. Hence, any setting of the last $k$ bits of $y$ will fully determine the first $q-k$ bits of it. These are the $2^{q-k}$ strings $y$ that satisfy $Hy = c$. □

## B.2 Binding Property

**Lemma B.2** (computational binding)**.** *Protocol 4.5 is computationally binding in the sense of the Definition 4.2.*

We prove Lemma B.2 by providing an algorithm $A$ that finds a valid witness (according to relation $W$) for a random string $y \leftarrow \{0,1\}^q$ with nonnegligible probability. Before describing $A$, we provide the following definitions.

**Definitions.** In the enumerated definitions below, $h_i$ is of the form $0^i 1\{0,1\}^{q-i-1}$, and $h_i(y) = \langle h_i, y\rangle$. Without loss of generality, we can assume that $S^*$ is deterministic because every probabilistic $S^*$ can be converted to a (nonuniform) deterministic one with the same success probability and running time by fixing its random coins to maximize its success probability.

1. For $0 \leq i < q$, let $\mathcal{H}_i$ denote the set of hash functions of the form $0^i 1\{0,1\}^{q-i-1}$, i.e., $\mathcal{H}_i = \{0^i 1w : w \in \{0,1\}^{q-i-1}\}$.
2. A *node* $N$ at level $i$ is defined by a series of hash functions $(h_0, h_1, \dots, h_{i-1})$, where each $h_j \in \mathcal{H}_j$. (Since $S^*$ is deterministic, this determines $c_0, \dots, c_{i-1}$ where $c_j = S^*(h_0, \dots, h_j)$.) Let $L_i$ denote the set of nodes at level $i$.
3. The set of compatible hash functions at node $N \in L_i$ is denoted as
$$\mathrm{Comp}(N, y) = \{h_i \in \mathcal{H}_i : S^*(N, h_i) = h_i(y)\},$$
where $S^*(N, h_i)$, with $N = (h_0, \dots, h_{i-1})$, denotes $S^*(h_0, \dots, h_i)$.
4. A string $y$ is $\gamma$-*balanced* at $N \in L_i$ if
$$\frac{1-\gamma}{2} \leq \frac{\mathrm{Comp}(N,y)}{|\mathcal{H}_i|} \leq \frac{1+\gamma}{2}.$$
A string $y$ is $\gamma$-*fully-balanced* at $N \in L_i$ if it is $\gamma$-balanced at all its parental nodes. That is, letting $N = (h_0, \dots, h_{i-1})$, $y$ is required to be $\gamma$-balanced at all $N_0 = (h_0), N_1 = (h_0, h_1), \dots, N = N_{i-1} = (h_0, \dots, h_{i-1})$.
5. A string $y$ is said to be *compatible* with a node $N = (h_0, \dots, h_{i-1})$ if $h_j(y) = S^*(h_1, \dots, h_j)$ for all $0 \leq j < i$. Let $U(N)$ denote the set of compatible $y$'s with node $N$. Note that for every $N \in L_i$, we have $|U(N)| = 2^{q-i}$.
6. Let $B(N)$ and $F(N)$ denote the set of $\gamma$-balanced strings and $\gamma$-fully-balanced strings at node $N$ respectively. Moreover, let $G(N) = U(N) \setminus F(N)$ be the set of strings that are not fully-balanced. Note that for every node $N$, we have $F(N) \subseteq B(N) \subseteq U(N)$.
7. At every node $N \in L_{q-k}$, we can assume WLOG that $S^*(N)$ outputs a pair of strings $(x_0, z_0)$ and $(x_1, z_1)$, but it is not necessarily the case that any of $x_b \in W_{C(z_b)}$.

**Description of the witness finding algorithm.** Algorithm $A$: On input $y \in \{0,1\}^q, 1^q, 1^k$ and $\varepsilon$, do the following.

1. Set parameters $\gamma = 1/q$, $\beta = \log(1/\varepsilon) + 2\log(q) + 4\log(1/\gamma) + 4$, and $\alpha = q - \beta - k$.
2. Repeat the following for $i = 1, \dots, \alpha - 1$:

When $A$ is at node $N \in L_i$, explore along a random $h_i \leftarrow \mathrm{Comp}(N, y)$ to get to a new node $N' = (N, h_i) \in L_{i+1}$. (This can be done efficiently by choosing a random $h_i \leftarrow \mathcal{H}_i$ and querying $S^*$ to make sure that $h_i \in \mathrm{Comp}(N, y)$, and repeat up to $8q$ times if not. If after $8q$ repetitive tries and fail to encounter any $h_i \in \mathrm{Comp}(N, y)$, then output `fail`.)

3. At node $N \in L_\alpha$, choose random $h_\alpha \leftarrow \mathcal{H}_\alpha, \ldots, h_{\alpha+\beta-1} \leftarrow \mathcal{H}_{\alpha+\beta-1}$, to arrive at node $\widetilde{N} = (N, h_\alpha, h_{\alpha+1}, \ldots, h_{\alpha+\beta-1}) \in L_{\alpha+\beta}$. (Note that $q - k = \alpha + \beta$, and hence $\widetilde{N} \in L_{\alpha+\beta} = L_{q-k}$.)

4. Query $S^*(\widetilde{N})$ to get $(x_0, z_0)$ and $(x_1, z_1)$. If either of $C(z_b) = y$, then output $x_b$. Else, output `fail`.

It is clear that the above algorithm runs in polynomial time (with oracle queries to $S^*$). All we need to show is that it succeeds with nonnegligible property, and we prove that property in the following claims.

**Claim B.3.** *For every node $N \in L_i$, the set of unbalanced strings, $U(N) \setminus B(N) \leq 2/\gamma^2$.*

*Proof of claim.* Let $X \subseteq U(N)$ be a set of size $2^d$, for some value of $d$. We also interpret $X$ as a distribution that puts equal weights on each of its $2^d$ elements.

Let $\mathcal{H}_i$ be the set of hash functions after node $N$ of the form $0^i 1\{0,1\}^{q-i-1}$. Observe that for every $x \neq x'$, $Pr_{h_i \leftarrow \mathcal{H}_i}[h_i(x) = h_i(x')] \leq 1/2$. Also, note that $h_i$ requires exactly $q - i - 1$ bits to describe.

Computing the collision probabilities (using the notation $\mathcal{H}_i$ to denote a random hash function from that family), we get

$$\begin{aligned}
\mathrm{Col}((\mathcal{H}_i, \mathcal{H}_i(X))) &\leq \mathrm{Col}(\mathcal{H}_i)(\mathrm{Col}(X) + \Pr[\mathcal{H}_i(X) = \mathcal{H}_i(X') : X \neq X']) \\
&\leq \mathrm{Col}(\mathcal{H}_i) \cdot (1/2^d + 1/2) \\
&= 2^{-(q-i-1)}(1/2^d + 1/2), \text{ whereas} \\
\mathrm{Col}((\mathcal{H}_i, U_1)) &= \mathrm{Col}(\mathcal{H}_i) \cdot 1/2 \\
&= 2^{-(q-i-1)} \cdot (1/2).
\end{aligned}$$

Therefore,

$$\begin{aligned}
\Delta((\mathcal{H}_i, \mathcal{H}_i(X)), (\mathcal{H}_i, U_1)) &= 1/2\,|(\mathcal{H}_i, \mathcal{H}_i(X)) - (\mathcal{H}_i, U_1)|_1 \\
&\leq 1/2 \cdot \sqrt{2^{q-i-1}}\sqrt{\mathrm{Col}((\mathcal{H}_i, \mathcal{H}_i(X))) - \mathrm{Col}((\mathcal{H}_i, U_1))} \\
&\leq 1/2\sqrt{1/2^d} \\
&= 2^{-d/2-1}.
\end{aligned}$$

Setting $d = 2\log(1/\gamma)$, we get that $\Delta((\mathcal{H}_i, \mathcal{H}_i(X)), (\mathcal{H}_i, U_1)) \leq \gamma/2$. Next, assume for sake of contradiction that $U(N) \setminus B(N) > 2^{d+1} = 2/\gamma^2$. Then we will have a set $M \subseteq U(N) \setminus B(N)$ of size greater then $2^d$ with elements that are unbalanced in one direction (i.e. all $> 1/2 + \gamma$, or all $< 1/2 - \gamma$). But this contradicts the assumption that $\Delta((\mathcal{H}_i, \mathcal{H}_i(T)), (\mathcal{H}_i, U_1)) \leq \gamma/2$ (since $|T| > 2^d$). □

The next claim follows by a union bound on the unbalanced elements.

**Claim B.4.** *For every node $N \in L_i$, the set of strings that are not fully balanced, $G(N) = U(N) \setminus F(N) \leq 2i/\gamma^2$. In particular, for $\gamma = 1/q$, $|F(N)| \geq |U(N)|/2$ for $i \leq q - 4\log q$.*

**Claim B.5.** *For every node $N \in L_\alpha$, the fraction of children nodes $N_{\alpha+\beta}$ with greater than one element from $G(N)$ is at most $\varepsilon/4$.*

> *Proof of claim.* Consider any fixed node $N \in L_\alpha$. The number of non-fully-balanced (aka bad) elements in that node is $G(N)$. Hence, the number of pairs of these bad elements is at most $|G(N)|^2$. Since for each $x \neq y \in U(N)$, $\Pr[h_i(x) = h_i(y)] \leq 1/2$ for all $\alpha \leq i < \alpha + \beta$, the fraction of children nodes $N' \in L_{\alpha+\beta}$ with greater than one element from $G(N)$ is at most $|G(N)|^2/2^\beta$.
>
> Since $\beta = \log(1/\varepsilon) + 2\log(q) + 4\log(1/\gamma) + 4$, we can bound $|G(N)|^2/2^\beta$ as follows:
>
> $$\begin{aligned} |G(N)|^2 \cdot 2^{-\beta} &\leq (2\alpha\gamma^{-2})^2 2^{-\beta} \\ &\leq 4q^2\gamma^{-4}2^{-\beta} \\ &< \varepsilon/4. \end{aligned}$$
>
> The result follows. □

A node $N \in L_{\alpha+\beta} = L_{q-k}$ is *witness revealing* if both of $S^*(N)$'s outputs, namely $(x_0, z_0)$ and $(x_1, z_1)$, satisfy $C(z_b) \in U(N)$ and $x_b \in W_{C(z_b)}$, for $b \in \{0, 1\}$. A node $N \in L_\alpha$ is said to be *good* if greater than $\varepsilon/2$ of its children at level $q - k$ are witness revealing.

**Claim B.6.** *The fraction of good nodes at level $\alpha$ is at least $\varepsilon/2$.*

> *Proof of claim.* By the assumption that
>
> $\Pr[x_0 \in W_{C(z_0)} \wedge x_1 \in W_{C(z_1)} : C = (S^*, R)(1^q, 1^k);\ ((x_0, z_0), (x_1, z_1)) = \mathsf{output}_{S^*}(S^*, R)] > \varepsilon,$
>
> we know that at least $\varepsilon$ fraction of all the nodes at level $q - k$ are nonbinding. And, by a Markov bound, we have that $\varepsilon/2$ fraction of nodes at level $\alpha$ are good. □

**Claim B.7.** *For any fixed $N \in L_\alpha$ and $y' \in F(N)$, we have*

$$\frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1+\gamma)^\alpha} \cdot \frac{1}{|L_\alpha|} \leq \Pr[A \text{ reaches } N \wedge y = y'] \leq \frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1-\gamma)^\alpha} \cdot \frac{1}{|L_\alpha|},$$

*where the probability is taken over $y \in \{0, 1\}^q$ and the random coins of $A$.*

> *Proof of claim.* Let $N = (h_0, h_2, \ldots, h_{\alpha-1})$, and for $1 \leq j \leq \alpha$, define $N_j = (h_0, \ldots, h_{j-1})$. To get the upper bound,
>
> $$\begin{aligned} \Pr[A \text{ reaches } N \wedge y = y'] &= \Pr[y = y'] \cdot \Pr[A \text{ reaches } N] \\ &= 2^{-q} \prod_{j=0}^{\alpha-1} \frac{1}{\mathrm{Comp}(N_j, y)} \\ &\leq 2^{-q} \prod_{j=0}^{\alpha-1} \frac{2}{1-\gamma} \cdot \frac{1}{|\mathcal{H}_j|} \\ &= \frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1-\gamma)^\alpha} \cdot \frac{1}{|L_\alpha|}. \end{aligned}$$

To get the lower bound, we use very similar techniques.

$$\begin{aligned}
\Pr[A \text{ reaches } N \wedge y = y'] &= 2^{-q} \prod_{j=0}^{\alpha-1} \frac{1}{\mathrm{Comp}(N_j, y)} \\
&\geq 2^{-q} \prod_{j=0}^{\alpha-1} \frac{2}{1+\gamma} \cdot \frac{1}{|\mathcal{H}_j|} \\
&= \frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1+\gamma)^{\alpha}} \cdot \frac{1}{|L_\alpha|}.
\end{aligned}$$

Our result follows. □

**Claim B.8.**

$$\Pr[\textit{The node } N \textit{ reached by } A \textit{ is good } \wedge y \in F(N)] \geq \frac{\varepsilon}{4(1+\gamma)^{\alpha}}.$$

*where the probability is taken over $y \in \{0,1\}^q$ and the random coins of $A$.*

*Proof of claim.* Let $N \in L_\alpha$ be any good node at level $\alpha$. Then,

$$\begin{aligned}
\Pr[A \text{ reaches } N \wedge y \in F(N)] &= \sum_{y' \in F(N)} \Pr[A \text{ reaches } N \wedge y = y'] \\
&\geq \sum_{y' \in F(N)} \frac{1}{|L_\alpha|} \cdot \frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1+\gamma)^{\alpha}} \\
&= \frac{|F(N)|}{2^{q-\alpha}} \cdot \frac{1}{|L_\alpha|} \cdot \frac{1}{(1+\gamma)^{\alpha}} \\
&= \frac{|F(N)|}{|U(N)|} \cdot \frac{1}{|L_\alpha|} \cdot \frac{1}{(1+\gamma)^{\alpha}} \\
&\geq \frac{1}{2} \cdot \frac{1}{|L_\alpha|} \cdot \frac{1}{(1+\gamma)^{\alpha}},
\end{aligned}$$

with the last inequality following from the fact that $|F(N)| / |U(N)| \geq 1/2$, noting $\alpha \leq q - 3\log q$ (refer to Claim B.4).

There are $|L_\alpha|$ nodes at level $\alpha$, and at least $\varepsilon/2$ fraction of them are good. Hence, we multiply the above probability by $(\varepsilon/2)|L_\alpha|$ to get our stated result. □

**Claim B.9.** *In any good node $N \in L_\alpha$, the fraction of nonbinding children of $N$ at level $\alpha + \beta$ that has one or less image in $G(N)$ is at least $\varepsilon/4$.*

*Proof of claim.* The fraction of nonbinding children is greater than $\varepsilon/2$, and by Claim B.5, the fraction of children nodes of $N$ with greater than one element from $G(N)$ is at most $\varepsilon/4$. □

**Claim B.10.** *For any fixed $N \in L_\alpha$ and $y' \in F(N)$, we have*

$$\Pr[y = y' | A \textit{ reaches } N \wedge y \in F(N)] \geq \frac{1}{|F(N)|} \left(\frac{1-\gamma}{1+\gamma}\right)^{\alpha},$$

*where the probability is taken over $y \in \{0,1\}^q$ and the random coins of $A$.*

*Proof of claim.* For any fixed $N \in L_\alpha$ and $y' \in F(N)$,

$$\Pr[y = y' | A \text{ reaches } N \wedge y \in F(N)] = \frac{\Pr[A \text{ reaches } N \wedge y = y']}{\Pr[A \text{ reaches } N \wedge y \in F(N)]}.$$

For the numerator, by Claim B.7,

$$\Pr[A \text{ reaches } N \wedge y = y'] \geq \frac{1}{|L_\alpha|} \cdot \frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1+\gamma)^\alpha}.$$

For the denominator, also using Claim B.7,

$$\begin{aligned}
\Pr[A \text{ reaches } N \wedge y \in F(N)] &= \sum_{y' \in F(N)} \Pr[A \text{ reaches } N \wedge y = y'] \\
&\leq \sum_{y' \in F(N)} \frac{1}{|L_\alpha|} \cdot \frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1-\gamma)^\alpha} \\
&= |F(N)| \cdot \frac{1}{|L_\alpha|} \cdot \frac{1}{2^{q-\alpha}} \cdot \frac{1}{(1-\gamma)^\alpha}.
\end{aligned}$$

Combining the two, we have our result. □

We have now reached our final claim to complete the proof of the binding theorem.

**Claim B.11.**

$$\Pr_{y \leftarrow \{0,1\}^q}[A(y) \in R_y] > c \cdot (\varepsilon^3 q^{-6} 2^{-k}) - \exp(q), \textit{ for some constant } c > 0.$$

*Proof of claim.* Note how $A$ operates. On input $y$, it follows a random compatible (with $y$) hash functions $h_i$ out of node $N \in L_i$, for $1 \leq i < \alpha$, and then takes random $h_i$'s (not necessarily compatible with $y$) when $\alpha \leq i < \alpha + \beta$. (For now, we can ignore failure to obtain compatible hash functions.)

Our algorithm $A$ will find a valid witness for $y$ if the following conditions happen.

1. Algorithm $A$ reaches a good node $N \in L_\alpha$ such that $y \in F(N)$. By Claim B.8, this happens with probability at least $\varepsilon/(4(1+\gamma)^\alpha)$.
2. Algorithm $A$ reaches a witness revealing child with at most one element in $G(N)$. Given that (1) occurs, by Claim B.9, this happens with probability at least $\varepsilon/4$.

   In this case, $S^*$ will output $(x_0, z_0)$ and $(x_1, z_1)$, such that at least one $(x_b, z_b)$ will be such that $x_b \in W_{C(z_b)}$ and $C(z_b) \in U(N) \setminus G(N) = F(N)$. Let $y' = C(z_b)$.
3. The string $y = y' = C(z_b)$. If this happens, then $A$ will output $x_b \in R_y$, a valid witness for $y$. By Claim B.10, we have that

$$\Pr[y = y' | A \text{ reaches } N \wedge y' \in F(N)] \geq \frac{1}{|F(N)|} \left( \frac{1-\gamma}{1+\gamma} \right)^\alpha.$$

Combining all the probabilities, we have

$$\begin{aligned}\Pr_{y \leftarrow \{0,1\}^q}[A(y) \in R_y] &\geq \frac{\varepsilon}{4(1+\gamma)^\alpha} \cdot \frac{\varepsilon}{4} \cdot \frac{1}{|F(N)|} \left(\frac{1-\gamma}{1+\gamma}\right)^\alpha \\ &\geq \frac{1}{2^{\beta+k}} \cdot \frac{\varepsilon^2}{32} \cdot \left(\frac{1-\gamma}{(1+\gamma)^2}\right)^q .\end{aligned}$$

With settings of $\gamma = 1/q$ and $\beta = \log(1/\varepsilon) + 2\log(q) + 4\log(1/\gamma)) + 4$, we have the probability of finding a witness to be greater than $c \cdot (\varepsilon^3 q^{-6} 2^{-k})$, for some constant $c \geq 0$.

Finally, we need to account for the case when we fail to find compatible hash functions $h_i$ out of node $N \in L_i$, for $1 \leq i < \alpha$. However, because our analysis has only focused on fully balanced $y$, and we repeat $8q$ times to find a compatible hash, the probability of failure is exponentially small. Therefore, the overall success probability is greater than $c \cdot (\varepsilon^3 q^{-6} 2^{-k}) - \exp(q)$. □