Statistically-Hiding Commitment from Any One-Way Function

Iftach Haitner[*] Omer Reingold[†]

Abstract

We give a construction of statistically-hiding commitment schemes (ones where the hiding property holds information theoretically), based on the minimal cryptographic assumption that one-way functions exist. Our construction employs two-phase commitment schemes, recently constructed by Nguyen, Ong and Vadhan (FOCS '06), and universal one-way hash functions introduced and constructed by Naor and Yung (STOC '89) and Rompel (STOC '90).

[*] Dept. of Computer Science and Applied Math., Weizmann Institute of Science, Rehovot 76100, Israel. E-mail: iftach.haitner@weizmann.ac.il.

[†] Department of Computer Science, Weizmann Institute of Science, Rehovot 76100, Israel. E-mail:omer.reingold@weizmann.ac.il. Research supported by grant no. 1300/05 from the Israel Science Foundation.

1 Introduction

A commitment scheme defines a two-stage interactive protocol between a sender $\mathcal{S}$ and a receiver $\mathcal{R}$; informally, after the *commit stage*, $\mathcal{S}$ is bound to (at most) one value, which is not yet revealed to $\mathcal{R}$, and in the *reveal stage* $\mathcal{R}$ finally learns this value. The two security properties hinted at in this informal description are known as *binding* (namely, that $\mathcal{S}$ is bound to at most one value after the commit stage) and *hiding* (namely, that $\mathcal{R}$ does not learn the value to which $\mathcal{S}$ commits before the reveal stage). In a statistically-hiding computationally-binding commitment scheme (for short, statistical commitment) the hiding property holds *even against all-powerful receivers* (i.e., hiding holds information-theoretically), while the binding property is required to hold only for polynomial-bounded senders.

Statistical commitment schemes can be used as a building block in constructions of statistical zero-knowledge arguments [BCC88, NOVY98] and certain coin-tossing protocols [Blu82, Lin03]. It therefore implies, via standard reduction, a way to transform protocols that are secure assuming an all powerful honest-but-curious party, into one that is secure even when this party maliciously deviates from the protocol. More generally, when used within protocols in which certain commitments are never revealed, statistical commitments have the following advantage over computationally-hiding commitment schemes: in such a scenario, it need only be infeasible to violate the binding property *during the period of time the protocol is run*, whereas the committed values will remain hidden *forever* (i.e., regardless of how much time the receiver invests after completion of the protocol).

Perfectly-hiding[1] commitment schemes were first shown to exist based on specific number-theoretic assumptions [BCC88, BKK90] or, more generally, based on any collection of claw-free permutations [GMR88] with an efficiently-recognizable index set [GK96] (see [GK96] for a weaker variant of statistically-hiding commitment which suffices for some applications and for which an efficiently-recognizable index set is not needed). Statistical commitment schemes can also be constructed from collision-resistant hash functions [DPP93, HM96]. Naor et al. [NOVY98] showed a construction of a perfectly-hiding commitment scheme based on any one-way permutation. Haitner et. al. [HHK$^+$05] make progress by constructing statistical commitment based on regular one-way functions and also on the so called approximable-size one-way functions.

In their recent breakthrough result, Nguyen et al. [NOV06] show how to construct statistical zero-knowledge arguments for NP based on any one-way function. The question of whether one-way functions imply statistical commitments, however, was still open.

We mention that the complementary notion of commitment schemes, where the hiding is computational and the binding holds even w.r.t. an all powerful sender, was already known to be implied by the existence of one-way functions [Nao91, HILL99].

1.1 Our result

Our main result is that the existence of one-functions is a sufficient condition for the existence of statistical commitment. Namely, we prove the following theorem.

Theorem 1.1. *Assuming that one-way functions exist, then there exists a statistically-hiding computationally-binding commitment.*

By Impagliazzo and Luby [IL89], the existence of statistical commitment schemes implies the existence of one-way functions and thus the above result is tight.

1.2 Our technique

Our protocol combines, in a sense, the following two fundamental cryptographic primitives: two-phase commitment schemes recently presented by Nguyen et al. [NOV06] (extending a similar notion given in [NV06]) and universal one-way hash functions (UOWHF) presented by Naor and Yung [NY89]. Following is an informal description of the these primitives (a formal definition appears in Section 2).

Universal one-way hash functions Universal one-way hash functions is a relaxation of the notion of collision-resistant hash functions. A family of compressing hash functions is universal one-way if no

[1] Very informally, in a statistically-hiding commitment scheme the receiver learns only a negligible amount of information about the sender's committed value, whereas in a perfectly-hiding commitment scheme the receiver learns *nothing*. Note that any perfectly-hiding scheme is also statistically hiding.

efficient algorithm succeeds in the following game with more than negligible probability. The algorithm should first announce a value x. Then, on a uniformly selected hash function f (given to the algorithm *after* it announces x), it should find $x' \neq x$ such that $f(x') = f(x)$.

Rompel [Rom90] shows that the existence of one-way functions implies the existence of universal one-way hash functions, this result was recently rewritten by Koo and Katz [KK05], adding missing details and fixing some errors.

Two-phase commitments In a two-phase commitment scheme, the sender and the receiver interact in two consecutive phases. In each phase they carry out a commitment protocol (the commit stage and the reveal stage). The transcript of the first phase is used as input for the second-phase commitment. A two-phase commitment is hiding, if before each of the reveal stages the receiver has no information about the value the commitment. A two-phase commitment is $\binom{2}{1}$-binding, if the sender cannot cheat both in the first phase and in the second phase. Specifically, after the first-phase commit, there is only *one* value, which the sender can decommit to in the first-phase reveal, that does not guarantee the binding of the second commitment (in the standard sense).

Nguyen et al. [NOV06] prove that the existence of one-way functions implies some non-uniform version of two-phase commitment schemes.

The construction idea We would like to use two-phase commitment schemes to construct a (standard) statistical commitment. A naive attempt to design the commitment scheme may go as follows: First, the sender commits to some random string x using the first-phase commit. Then, the receiver flips a random bit dec, if dec is zero then the first-phase commit is used as the commitment (e.g., the sender sends to the receiver the exclusive or of its secret with the random string). Otherwise ($\mathsf{dec} = 1$), the two parties execute the first-phase reveal and if successful (i.e., the receiver does not reject), they use the second-phase commit (invoked with the transcript of the first-phase as input) as the commitment.

The intuition is that since the two-phase commitment is $\binom{2}{1}$-binding, the sender cannot cheat in both phases together and thus the receiver would catch a cheating sender with probability half. The problem is, however, that the sender can decide in which commitment he likes to cheat *after* knowing dec. Hence, the sender can cheat successfully in both cases without breaking the $\binom{2}{1}$-binding of the underlying protocol.

Our key idea is to use universal one-way hash functions in order to force the sender to decide in which phase it is about to cheat *before* knowing dec. Our actual implementation is as follows: After the first-phase commit, the receiver selects a random (universal one-way) hash function f and the sender sends him back $y = f(x)$. The protocol proceeds as the naive protocol above, where any time the first-phase reveal is executed in the naive protocol revealing the value x' (either in the commit-stage for $\mathsf{dec} = 0$ or in the reveal stage for $\mathsf{dec} = 1$), the receiver also verifies that $f(x') = y$.

Assuming that the hash function, f, is "compressing enough", the string x (committed to in the first-phase commitment) remains sufficiently hidden even $f(x)$ is sent to $\mathcal{R}$ (in the new variant of the protocol). Thus, in the case that $\mathsf{dec} = 0$, the string x can still be used to statistically hide the sender's secret (assuming it is sufficiently shorter than x). To show the statistical hiding in the complementary case when $\mathsf{dec} = 1$, it is sufficient to note that sending $f(x)$, does not compromise the hiding property of the second-phase commitment. All in all, the protocol is hiding for both choices of dec and thus it is hiding.

To argue about the binding of the protocol, recall that the $\binom{2}{1}$-binding of the two-phase commitment scheme informally states that after the first-phase commit, there exists a *single* value $\widetilde{x}$ that allows the sender to cheat in the second-phase commitment. Now, if the sender sends y such that $f(\widetilde{x}) = y$, then in order to cheat in the case $\mathsf{dec} = 0$, it will have to open the first-phase commitment to a value $x' \neq \widetilde{x}$ such that $f(x') = y = f(\widetilde{x})$. This would imply the breaking of the universal one-way hash functions. On the other hand, if $f(\widetilde{x}) \neq y$, then in the case $\mathsf{dec} = 1$ the sender is forced to open the first-phase commitment to a value different than $\widetilde{x}$. This guarantees that the sender cannot cheat in the second-phase commitment and thus in this case our protocol is binding. In conclusion, since y is sent before dec is chosen, we are guaranteed that our protocol is weakly-binding (since intuitively there always exist a choice of dec that prevent the sender from cheating). We complete the construction by amplifying the above protocol into a full-fledged statistical commitment scheme using standard techniques.

2 Preliminaries

2.1 Notation

We denote the i^{th} bit of a string x by $x[i]$. We denotes the exclusive or of the bits x and y by $x \oplus y$. For $k \in \mathbb{N}$, we denote by $[k]$ the set $\{1, \ldots, k\}$. Given a set L, we denote by $x \leftarrow L$ the experiment in which x is uniformly chosen from L. The statistical distance of two distributions P and Q over Ω, denoted $SD(P,Q)$, is defined as

$$SD(P,Q) \stackrel{\text{def}}{=} \frac{1}{2}\sum_{x\in\Omega}\left|\Pr_P(x) - \Pr_Q(x)\right|.$$

Given two interactive Turing machines A and B, we denote the protocol they define by (A,B) and denote the following experiment by $(o_A \mid o_B) \leftarrow \langle A(i_A), B(i_B)\rangle$: The protocol (A,B) is invoked with inputs i_A and i_B and the outputs of the parties are assigned to o_A and o_B respectively.

2.2 Families of pairwise-independent hash functions

Definition 2.1. *(efficient family of pairwise-independent hash functions) Let $\mathcal{H}$ be a family of functions mapping strings of length $\ell(n)$ to strings of length $m(n)$. We say that $\mathcal{H}$ is an* efficient family of pairwise independent hash functions *(following [CW77]) if the following hold:* [2]

Samplable *$\mathcal{H}$ is polynomially samplable (in n).*

Efficient *There exists a polynomial-time algorithm that given $x \in \{0,1\}^{\ell(n)}$ and a description of $h \in \mathcal{H}$ outputs $h(x)$.*

Pairwise independence *For every distinct $x_1, x_2 \in \{0,1\}^{\ell(n)}$ and every $y_1, y_2 \in \{0,1\}^{m(n)}$, we have:*

$$\Pr_{h \leftarrow \mathcal{H}}[h(x_1) = y_1 \bigwedge h(x_2) = y_2] = 2^{-2m(n)}.$$

It is well known ([CW77]) that there exists an efficient family of pairwise-independent hash functions for every choice of ℓ and m whose elements description size is $\mathcal{O}(\max\{\ell(n), m(n)\})$.

In this paper we focus on Boolean families of hash functions (i.e., $m(n) = 1$). The following standard lemma (see for example, [Gol01, Lemma 4.3.1]) states that a random pairwise independent hash function partitions a given set into (almost) equal size subsets.

Lemma 2.2. *Let $\mathcal{H}$ be a family of Boolean pairwise independent hash functions defined over strings of length $\ell(n)$ and let $L \subseteq \{0,1\}^{\ell(n)}$. Then for every $\delta > 0$*

$$\Pr_{h \leftarrow \mathcal{H}}\left[\left|\,\left|h^{-1}(1) \cap L\right| - \left|h^{-1}(0) \cap L\right|\,\right| > \delta \cdot |L|\right] < \tfrac{4}{\delta^2 \cdot |L|}.$$

2.3 Universal one-way hash functions (UOWHF)

Definition 2.3. *(universal one-way hash functions (UOWHF)) Let $\mathcal{F}$ be a family of functions mapping strings of length $\ell(n)$ to strings of length $m(n)$. We say that $\mathcal{F}$ is a family of* universal one-way hash functions *(following [NY89]) if the following hold:* [3]

Samplable *$\mathcal{F}$ is polynomially samplable (in n).*

[2] The first two properties, regarding the efficiency of the family, implicitly assume an ensemble of families (one family for every value of n). For simplify of presentation, we only refer to a single family.

[3] As in Definition 2.1, the first two properties, regarding the efficiency of the family, implicitly assume an ensemble of families (one family for every value of n).

Efficient *There exists a polynomial-time algorithm that given* $x \in \{0,1\}^{\ell(n)}$ *and a description of* $f \in \mathcal{F}$ *outputs* $f(x)$*.*

Compression $m(n) < \ell(n)$*.*

Hardness *For all* PPT A *and* $x \in \{0,1\}^{\ell(n)}$ *the following is negligible in* n*:*

$$\Pr[(x, \mathsf{state}) \leftarrow A(1^n), f \leftarrow \mathcal{F}, x' \leftarrow A(x, \mathsf{state}, f) : x' \neq x \bigwedge f(x') = f(x)].$$

By [Rom90] (full proof is given in [KK05]), it follows that assuming the existence of a one-way function, there exists a family of universal one-way hash functions for some polynomial $\ell(n) \geq n$*. Following [NY89, Lemma 2.1], we have that the latter construction implies a construction with* $m(n) \leq \frac{1}{2}\ell(n)$*.*

Remark 2.4. *The Hardness property of Definition 2.3 is somewhat stronger than the one given in [KK05] (and somewhat weaker than the original definition in [NY89]). The strengthening is in allowing* A *to transfer additional information, i.e.,* state*, between the selection of* x *and finding the collision. We note that the proof in [KK05] holds also w.r.t. to our stronger definition (and even w.r.t. the original definition of [NY89]).*

2.4 Commitment schemes

In this paper we focus on bit-commitment schemes (i.e., the committed string is a single bit). Bit-commitment schemes imply, via standard reductions, commitment schemes of any (polynomial) length.

An interactive bit-commitment scheme $(\mathcal{S}, \mathcal{R})$, with security parameter n, consists of two probabilistic polynomial-time interactive protocols: $(\mathcal{S}_c, \mathcal{R}_c)$ the commit stage, and $(\mathcal{S}_r, \mathcal{R}_r)$ the reveal stage. We note that in all the constructions of this paper, the reveal stage will always be non interactive, consisting of a single message from the sender to the receiver. Throughout, both parties receive the security parameter 1^n as an input.

1. In the commit stage: $\mathcal{S}_c$ receives a private input $b \in \{0,1\}$. At the end, $\mathcal{S}_c$ *locally* outputs some private information prvt and $\mathcal{R}_c$ outputs some public information pub.

2. In the reveal stage: $\mathcal{S}_r$ and $\mathcal{R}_r$ receive a common input pub and a bit b and $\mathcal{S}_r$ receives a private input prvt. At the end, $\mathcal{R}_r$ accepts or rejects.

We make the following correctness requirement: For all n, all $b \in \{0,1\}$, and every pair $(\mathsf{prvt}, \mathsf{pub})$ that may be output by $\langle \mathcal{S}_c(1^n, b), \mathcal{R}_c(1^n) \rangle$, it is the case that $\mathcal{R}_r$ accepts in $\langle \mathcal{S}_r(1^n, \mathsf{prvt}, \mathsf{pub}, b), \mathcal{R}_r(1^n, \mathsf{pub}, b) \rangle$.

The security of a commitment scheme can be defined in two complementary ways, protecting against either an all-powerful sender or an all-powerful receiver. In this paper, we are interested in the case of statistical commitment (i.e., the latter case).

Definition 2.5. *(hiding)* A bit-commitment scheme $(\mathcal{S}, \mathcal{R})$ is *ρ-hiding* (for ρ a function of n) if the following holds: Given an ITM $\mathcal{R}^*$, let $\mathsf{view}_{\langle \mathcal{S}_c(b), \mathcal{R}^* \rangle}(n)$ denote the distribution on the view of $\mathcal{R}^*$ when interacting with $\mathcal{S}_c(1^n, b)$ (this view simply consists of $\mathcal{R}^*$'s random-coins and the sequence of messages it receives from $\mathcal{S}_c$), where this distribution is taken over the random coins of $\mathcal{S}_c$ and $\mathcal{R}^*$. Then we require that for any (even all-powerful) $\mathcal{R}^*$ the two ensembles $\left\{\mathsf{view}_{\langle \mathcal{S}_c(0), \mathcal{R}^* \rangle}(n)\right\}$ and $\{\mathsf{view}_{\langle \mathcal{S}_c(1), \mathcal{R}^* \rangle}(n)\}$ have statistical difference at most ρ.

We say that a scheme is *statistically hiding* if it is ρ-hiding for negligible ρ. A 0-hiding scheme is called *perfectly hiding.*

Definition 2.6. *(Binding-break) Let* $(\mathcal{S}, \mathcal{R})$ *be a bit commitment protocol and let* $S^* = (S^*_c, S^*_r)$ *be an algorithm that is trying to break the binding of this protocol. For any possible values of the commit stage,* $\mathsf{outs} = (\mathsf{prvt}, \mathsf{pub})$*, we define the function*

$$\mathsf{BndBreak}^{S^*_r, \mathcal{R}_r}(\mathsf{outs}) \overset{def}{=} \min_{b \in \{0,1\}} \Pr[\langle S^*_r(\mathsf{outs}, b), \mathcal{R}_r(\mathsf{pub}, b) \rangle = \mathtt{Accept}].$$

Definition 2.7. *(binding) A bit-commitment scheme $(\mathcal{S},\mathcal{R})$ is ρ-binding (for ρ a function of n), if for all* PPT *S^* and any positive polynomial p, the following holds for large enough n:*

$$\Pr_{\mathsf{outs} \leftarrow <S_c^*(1^n),\mathcal{R}_c(1^n)>}[\mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}) > \frac{1}{p(n)}] < \rho(n).$$

Note that in the above, assuming that S^* consists of two separate algorithms is without loss of generality, since any information that S^* passes between the two stages can be encoded into its private output prvt. We say that a scheme is *computationally binding* if it is ρ-binding for negligible ρ. The following amplifications are standard (see for example [HHK$^+$05]).

Proposition 2.8. *There exists an efficient procedure that given polynomially many bit-commitment schemes which are all computationally binding and at least one of them is statistically hiding, outputs a computationally-binding statistically-hiding bit-commitment scheme.*

Proposition 2.9. *There exists an efficient procedure that given a ρ-binding bit-commitment scheme for noticeable ρ, outputs a computationally-binding bit-commitment scheme, which is statistically hiding if the given bit-commitment scheme is statistically hiding.*

2.5 Two-phase commitments

The following definitions for two-phase commitment schemes given in this section, are slight generalizations of the same definitions given in [NOV06] (which in turn are an extension of the definitions given in [NV06]).

Definition 2.10. *(two-phase commitments) A* two-phase commitment scheme *(S,R), with security parameter n and message lengths $(k_1,k_2) = (k_1(n),k_2(n))$, consists of four probabilistic polynomial-time interactive protocols: (S_c^1,R_c^1) the first commit stage, (S_r^1,R_r^1) the first reveal stage, (S_c^2,R_c^2) the second commit stage, and (S_r^2,R_r^2) the second reveal stage. Throughout, both parties receive the security parameter 1^n as input.*

1. *In the first commit stage, S_c^1 receives a private input $\sigma^{(1)} \in \{0,1\}^{k_1}$. At the end, S_c^1* locally *outputs some private information prvt^1 and R_c^1 outputs some public string pub^1.*

2. *In the first reveal stage, S_r^1 and R_r^1 receive as common input pub^1 and a string $\sigma^{(1)} \in \{0,1\}^{k_1}$ and S_r^1 receives as private input prvt^1. Let trans be the transcript of the first commit stage and the first reveal stage and includes R_r^1's decision to accept or reject.*

3. *In the second commit stage, S_c^2 and R_c^2 both receive the common input trans, and S_c^2 receives a private input $\sigma^{(2)} \in \{0,1\}^{k_2}$. At the end, S_c^2* locally *outputs some private information prvt^2 and R_c^2 outputs some public string pub^2.*

4. *In the second reveal stage, S_r^2 and R_r^2 receive as common input pub^2 and a string $\sigma^{(2)} \in \{0,1\}^{k_2}$, and S_r^2 receives as private input prvt^2. At the end, R_r^2 accepts or rejects.*

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

Definition 2.11. *(hiding) A two-phase commitment scheme (S,R), with security parameter n and message lengths $(k_1,k_2) = (k_1(n),k_2(n))$, is* statistically hiding *if the following hold: Given an ITM $\mathcal{R}^*$ and $\sigma^{(1)} \in \{0,1\}^{k_1}$, let $\mathsf{view}_{\langle\mathcal{S}_c^1(\sigma^{(1)}),\mathcal{R}^*\rangle}(n)$ denote the distribution on the view of $\mathcal{R}^*(1^n)$ when interacting with $\mathcal{S}_c^1(1^n,\sigma^{(1)})$. Similarly, for $\sigma^{(2)} \in \{0,1\}^{k_2}$ and $\Lambda \in \{0,1\}^*$ let $\mathsf{view}_{\langle\mathcal{S}_c^2(\sigma^{(2)}),\mathcal{R}^*\rangle}(\Lambda)$ denote the distribution on the view of $\mathcal{R}^*(\Lambda)$ when interacting with $\mathcal{S}_c^2(\sigma^{(2)},\Lambda)$. We require that for any (even all-powerful) $\mathcal{R}^*$,*

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 $\sigma^{(1)},\widetilde{\sigma}^{(1)} \in \{0,1\}^{k_1}$, $\mathsf{view}_{\langle\mathcal{S}_c^1(\sigma^{(1)}),\mathcal{R}^*\rangle}(n)$ is statistically indistinguishable to $\mathsf{view}_{\langle\mathcal{S}_c^1(\widetilde{\sigma}^{(1)}),\mathcal{R}^*\rangle}(n)$.*

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 $\sigma^{(1)} \in \{0,1\}^{k_1}$ and $\sigma^{(2)}, \widetilde{\sigma}^{(2)} \in \{0,1\}^{k_2}$, $\mathsf{view}_{\langle \mathcal{S}_c^2(\sigma^{(2)}), \mathcal{R}^* \rangle}(\Lambda)$ is statistically indistinguishable to $\mathsf{view}_{\langle \mathcal{S}_c^2(\widetilde{\sigma}^{(2)}), \mathcal{R}^* \rangle}(\Lambda)$, where $\Lambda = \mathsf{transcript}\langle S^1(1^n, \sigma^{(1)}), R^*(1^n)\rangle$.*

We stress that the second condition of the above hiding definition (Definition 2.11) requires that the view of receiver in the second phase be indistinguishable for any two messages even given the transcript of the first phase, $\Lambda = \mathsf{transcript}\langle S^1(1^n, \sigma^{(1)}), R^*(1^n)\rangle$.

Loosely speaking, the binding property says that *at least* one of the two commit 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 the common output pub^j, S^* can open pub^j successfully both as $\sigma^{(1)}$ and $\widetilde{\sigma}^{(1)} \neq \sigma^{(1)}$ with non-negligible probability. Actually, we allow this bad phase to be determined dynamically by S^*. Moreover, the second phase is *statistically* binding if the sender breaks the first phase. [4]

Definition 2.12. *($\binom{2}{1}$-binding) A two-phase commitment scheme (S,R), with security parameter n and message lengths $(k_1,k_2) = (k_1(n), k_2(n))$, is* computationally $\binom{2}{1}$-binding *if there exists a set $\mathcal{B}$ of first-phase transcripts and a negligible function ε such that:*

1. *For every (even unbounded) sender S^*, the first-phase transcripts in $\mathcal{B}$ make the second phase statistically binding, i.e. $\forall S^*, \forall \mathsf{trans} \in \mathcal{B}$, with probability at least $1-\varepsilon(n)$ over pub^2, the output of R_c^2 in $\langle S^*(\mathsf{trans}), R_c^2(\mathsf{trans})\rangle$, there is at most one value $\sigma^{(2)} \in \{0,1\}^{k_2}$ such that $\langle S^*(\mathsf{pub}^2, \sigma^{(2)}), R_r^2(\mathsf{pub}^2, \sigma^{(2)})\rangle = \mathtt{Accept}$.*

2. *$\forall$ nonuniform PPT S^*, 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 R_c^1 outputs pub^1. Let trans^1 be the transcript of the interaction.*

   (b) *S^* outputs two full transcripts trans and $\widetilde{\mathsf{trans}}$ of* both *phases with the following three properties:*
   - *Transcripts trans and $\widetilde{\mathsf{trans}}$ both start with prefix trans^1.*
   - *The transcript trans contains a successful opening of pub^1 to the value $\sigma^{(1)} \in \{0,1\}^{k_1}$ using a first-phase transcript not in $\mathcal{B}$, and R_r^1 and R_r^2 both accept in trans.*
   - *The transcript $\widetilde{\mathsf{trans}}$ contains a successful opening of pub^1 to the value $\widetilde{\sigma}^{(1)} \in \{0,1\}^{k_1}$ using a first-phase transcript not in $\mathcal{B}$, and R_r^1 and R_r^2 both accept in $\widetilde{\mathsf{trans}}$.*

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

The following theorem appears in [NOV06].

Theorem 2.13. *([NOV06, Theorem 7.10]) If one way functions exist, then on security parameter n, we can construct in time $\mathsf{poly}(n)$ a collection of public-coin two-phase commitment schemes $Com_1, \ldots, Com_m$ for $m = \mathsf{poly}(n)$ such that:*

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

2.6 Extending the message length

While Theorem 2.13 implies a set of two-phase commitment schemes with some given message lengths, for our purposes we need the message length of the first-phase commitment to be sufficiently (though still polynomially) long. The following lemma allows us to expand the message length of the first-phase commitment.

[4] In this paper, we do not use the fact that the second phase is statistically binding and not merely computationally binding.

Lemma 2.14. *There exists an efficient procedure that given a two-phase commitment scheme with message lengths $(k_1(n), k_2(n))$ and a positive polynomial p, outputs a two-phase commitment scheme with message lengths $(p(n), 1)$, which is hiding whenever the given scheme is hiding and it is $\binom{2}{1}$-binding whenever the given scheme is $\binom{2}{1}$-binding.*

Remark 2.15. *Lemma 2.14 transforms any two-phase commitment scheme with message lengths $(k_1(n), k_2(n))$ into a two-phase commitment scheme with message lengths $(p(n), 1)$ for any positive polynomial p. It is then possible to use similar ideas in order to construct a two-phase commitment scheme with message lengths $(p(n), q(n))$ for any positive polynomial q. For the purpose of this paper, however, the $(p(n), 1)$ reduction suffices.*

Proof. (of Lemma 2.14) Let $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ be a two-phase commitment with message lengths $(k_1(n), k_2(n))$. We assume w.l.o.g. that $k_1(n) = k_2(n) = 1$, since we can always decide to use only the first bit of the commitments. We define the two-phase commitment $(\mathcal{S}, \mathcal{R})$ with message lengths $(p(n), 1)$ as follows:

First-phase commit:

Common input: 1^n.
Sender's private input: $x_1 \in \{0,1\}^{p(n)}$.

1. For $i = 1 \ldots, p(n)$,

   $(\mathcal{S}^1_c, \mathcal{R}^1_c)$ run $\langle \widetilde{\mathcal{S}}^1_c(x_1[i]), \widetilde{\mathcal{R}}^1_c(1^{\ell(n)})\rangle$, with $\mathcal{S}^1_c$ and $\mathcal{R}^1_c$ acting as $\widetilde{\mathcal{S}}^1_c$ and $\widetilde{\mathcal{R}}^1_c$ respectively.

   Let pub^1_i be the public output and let prvt^1_i be the private output of $\widetilde{\mathcal{S}}^1_c$ in the above execution.
2. $\mathcal{S}^1_c$ locally outputs $\mathsf{prvt}^1 = (\mathsf{prvt}^1_1, \ldots, \mathsf{prvt}^1_{p(n)})$ and $\mathcal{R}^1_c$ outputs $\mathsf{pub}^1 = (\mathsf{pub}^1_1, \ldots, \mathsf{pub}^1_{p(n)})$.

First-phase reveal:

Common input: 1^n, $\mathsf{pub}^1 = (\mathsf{pub}^1_1, \ldots, \mathsf{pub}^1_{p(n)})$ and $x_1 \in \{0,1\}^{p(n)}$.
Sender's private input: $\mathsf{prvt}^1 = (\mathsf{prvt}^1_1, \ldots, \mathsf{prvt}^1_{p(n)})$.

1. For $i = 1 \ldots, p(n)$,

   $(\mathcal{S}^1_c, \mathcal{R}^1_c)$ run $\langle \widetilde{\mathcal{S}}^1_r(\mathsf{prvt}^1_i, \mathsf{pub}^1_i, x_1[i]), \widetilde{\mathcal{R}}^1_r(\mathsf{pub}^1_i), x_1[i]\rangle$, with $\mathcal{S}^1_c$ and $\mathcal{R}^1_c$ acting as $\widetilde{\mathcal{S}}^1_c$ and $\widetilde{\mathcal{R}}^1_c$ respectively. Let trans_i be the transcript of the execution.

2. $\mathcal{S}^1_c$ accepts if $\widetilde{\mathcal{S}}^1_r$ accepts in all of the above executions.

Second-phase commit:

Common input: $\mathsf{trans} = (\mathsf{trans}_1, \ldots, \mathsf{trans}_{p(n)})$.
Sender's private input: $b \in \{0,1\}$.

1. For $i = 1 \ldots, p(n)$,

   $(\mathcal{S}^2_c, \mathcal{R}^2_c)$ run $\langle \widetilde{\mathcal{S}}^2_c(b, \mathsf{trans}_i), \widetilde{\mathcal{R}}^1_2(\mathsf{trans}_i)\rangle$, with $\mathcal{S}^2_c$ and $\mathcal{R}^2_c$ acting as $\widetilde{\mathcal{S}}^2_c$ and $\widetilde{\mathcal{R}}^2_c$ respectively.

   Let pub^2_i be the public output and let prvt^2_i be the private output of $\widetilde{\mathcal{S}}^2_c$ in the above execution.
2. $\mathcal{S}^2_c$ locally outputs $\mathsf{prvt}^2 = (\mathsf{prvt}^2_1, \ldots, \mathsf{prvt}^2_{p(n)})$ and $\mathcal{R}^2_c$ outputs $\mathsf{pub}^2 = (\mathsf{pub}^2_1, \ldots, \mathsf{pub}^2_{p(n)})$.

Second-phase reveal:

Common input: $\mathsf{pub}^2 = (\mathsf{pub}^2_1, \ldots, \mathsf{pub}^2_{p(n)})$ and $b \in \{0,1\}$.
Sender's private input: $\mathsf{prvt}^2 = (\mathsf{prvt}^2_1, \ldots, \mathsf{prvt}^2_{p(n)})$.

1. For $i = 1 \ldots, p(n)$,

   $(\mathcal{S}^2_c, \mathcal{R}^2_c)$ run $\langle \widetilde{\mathcal{S}}^2_r(\mathsf{prvt}^2_i, \mathsf{pub}^2_i, b), \widetilde{\mathcal{R}}^1_r(\mathsf{pub}^2_i), b\rangle$, with $\mathcal{S}^2_c$ and $\mathcal{R}^2_c$ acting as $\widetilde{\mathcal{S}}^2_c$ and $\widetilde{\mathcal{R}}^2_c$ respectively.
2. $\mathcal{S}^2_c$ accepts if $\widetilde{\mathcal{S}}^1_r$ accepts in all of the above executions.

The correctness of $(\mathcal{S}, \mathcal{R})$ is evident, and it is also clear that $(\mathcal{S}, \mathcal{R})$ is hiding given that $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ is. Assuming that $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ is $\binom{2}{1}$-binding, we show that $(\mathcal{S}, \mathcal{R})$ is $\binom{2}{1}$-binding as follows: We define $\mathcal{B}$, a set of first-phase transcripts of $(\mathcal{S}, \mathcal{R})$ as $B \overset{\text{def}}{=} \left\{\mathsf{outs}^2 = (\mathsf{outs}^2_1, \ldots, \mathsf{outs}^2_{p(n)}) : \exists i \in p(n) \text{ s.t. } \mathsf{outs}^2_i \in \widetilde{B}\right\}$, where $\widetilde{B}$ is the set of first-phase transcripts of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ that make that second-phase commitment statistically binding. It is easy to see that indeed any transcript in $\mathcal{B}$, makes the second-phase commitment of $(\mathcal{S}, \mathcal{R})$ statistically binding (as in Definition 2.12). Finally, let A be an adversary that breaks the $\binom{2}{1}$-binding of $(\mathcal{S}, \mathcal{R})$ by outputting two transcripts $\mathsf{trans} = (\mathsf{trans}_1 \ldots, \mathsf{trans}_{p(n)})$ and $\widetilde{\mathsf{trans}} = (\widetilde{\mathsf{trans}}_1 \ldots, \widetilde{\mathsf{trans}}_{p(n)})$. By our definition of $\mathcal{B}$, there must exists an index $i \in p(n)$ such that both trans_i and $\widetilde{\mathsf{trans}}_i$ are not in $\widetilde{B}$, trans_i and $\widetilde{\mathsf{trans}}_i$ contain different first-phase openings $\sigma^{(1)} \neq \widetilde{\sigma}^{(1)}$, and $\widetilde{\mathcal{R}}^1_r$ and $\widetilde{\mathcal{R}}^2_r$ accept in both transcripts.

Since the latter holds for any breaking of the $\binom{2}{1}$-binding of $(\mathcal{S}, \mathcal{R})$, there must exist $i' \in p(n)$ (which can be efficiently found) such that A breaks the $\binom{2}{1}$-binding of $(\mathcal{S}, \mathcal{R})$ conditioned that the above holds w.r.t. $\mathsf{trans}_{i'}$ and $\widetilde{\mathsf{trans}}_{i'}$. Thus, the existence of A implies an adversary the breaks the $\binom{2}{1}$-binding of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$. □

3 The Construction

Given a two-phase commitment scheme, we construct a bit-commitment scheme such that the following holds: The scheme is statistically hiding whenever the two-phase commitment scheme is hiding and the scheme is computationally binding whenever the two-phase commitment scheme is $\binom{2}{1}$-binding. Thus, assuming that one-way functions exist, the existence of a polynomial set of computationally-binding bit-commitment schemes where at least one of them is statistically hiding follows by [NOV06, Theorem 7.10]. Finally, we use standard reductions to amplify the latter set of commitment schemes into a full-fledged statistical commitment scheme.

3.1 Main reduction

In this section we construct a bit-commitment scheme such that the following hold: The scheme is statistically hiding whenever the two-phase commitment is hiding, and the scheme is *weekly* binding whenever the two-phase commitment is $\binom{2}{1}$-binding.

Construction 3.1. *(The basic scheme) Let $\mathcal{F}$ be a family of universal one-way hash functions mapping strings of length $\ell(n)$ to strings of length $m(n) \leq \frac{1}{2}\ell(n)$, let $\mathcal{H}$ be a family of Boolean pairwise independent hash functions defined over strings of length $\ell(n)$ and finally let $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ be a two-phase commitment scheme with message lengths $(\ell(n), 1)$.*

Commit stage:

Common input: 1^n.
Sender's private input: $b \in \{0,1\}$.

// **First-phase commit:**

1. *$\mathcal{S}_c$ chooses uniformly at random $x_1 \in \{0,1\}^{\ell(n)}$.*
2. *$(\mathcal{S}_c, \mathcal{R}_c)$ run $\langle \widetilde{\mathcal{S}}_c^1(x_1), \widetilde{\mathcal{R}}_c^1(1^n) \rangle$, with $\mathcal{S}_c$ and $\mathcal{R}_c$ acting as $\widetilde{\mathcal{S}}_c^1$ and $\widetilde{\mathcal{R}}_c^1$ respectively.*

   Let pub^1 be the public output and let prvt^1 be the private output of $\widetilde{\mathcal{S}}_c^1$ in the above execution.
3. *$\mathcal{R}_c$ chooses uniformly at random $f \in \mathcal{F}$ and sends it to $\mathcal{S}$.*
4. *$\mathcal{S}_c$ sends $y = f(x_1)$ back to $\mathcal{R}$.*
5. *$\mathcal{R}_c$ flips a random coin $\mathsf{dec} \in \{0,1\}$.*

   If $\mathsf{dec} = 0$, // **Relying on the first-phase commitment.**

   (a) *$\mathcal{S}_c$ chooses uniformly at random $h \in \mathcal{H}$ and sends h and $c = b \oplus h(x)$ to $\mathcal{R}_c$.*

   (b) *$\mathcal{R}_c$ outputs $\mathsf{pub} = (\mathsf{dec}, \mathsf{pub}^1, f, y, h, c)$.*

   (c) *$\mathcal{S}_c$ locally outputs $\mathsf{prvt} = (\mathsf{prvt}^1, x_1)$.*

   Otherwise (i.e., $\mathsf{dec} = 1$*),* // **Verifying the first-phase commitment and moving to second-phase commitment.**

   $\mathcal{S}_c$ sends x_1 to $\mathcal{R}_c$ and $(\mathcal{S}_c, \mathcal{R}_c)$ run $\langle \widetilde{\mathcal{S}}_r^1(\mathsf{prvt}^1, \mathsf{pub}^1, x_1), \widetilde{\mathcal{R}}_r^1(\mathsf{pub}^1), x_1) \rangle$, with $\mathcal{S}_c$ and $\mathcal{R}_c$ acting as $\widetilde{\mathcal{S}}_c^1$ and $\widetilde{\mathcal{R}}_c^1$ respectively. Let trans be the transcript of the execution.

   If $\widetilde{\mathcal{R}}_c^r$ rejects, then $\mathcal{R}_c$ outputs $\perp$ (i.e., it will be impossible to decommit this execution).

   Otherwise,

   (a) *$(\mathcal{S}_c, \mathcal{R}_c)$ run $\langle \widetilde{\mathcal{S}}_c^2(b, \mathsf{trans}), \widetilde{\mathcal{R}}_c^2(\mathsf{trans}) \rangle$, with $\mathcal{S}_c$ and $\mathcal{R}_c$ acting as $\widetilde{\mathcal{S}}_c^2$ and $\widetilde{\mathcal{R}}_c^2$ respectively. Let pub^2 be the public output and let prvt^2 be the private input of $\widetilde{\mathcal{S}}_c^2$ in the above execution.*

   (b) *$\mathcal{S}_c$ locally outputs $\mathsf{prvt} = \mathsf{prvt}^2$ and $\mathcal{R}_c$ outputs $\mathsf{pub} = (\mathsf{dec}, \mathsf{pub}^2)$.*

Reveal stage:

In case $\mathsf{dec} = 0$**,**
Common input: 1^n, $b \in \{0,1\}$ *and* $\mathsf{pub} = (0, \mathsf{pub}^1, f, y, h, c)$.
Sender's private input: $\mathsf{prvt} = (\mathsf{prvt}^1, x_1)$.

$\mathcal{S}_r$ sends x_1 to $\mathcal{R}_r$ and $(\mathcal{S}_r, \mathcal{R}_r)$ run $\langle \widetilde{\mathcal{S}}_r^1(\mathsf{prvt}^1, \mathsf{pub}^1, x_1), \widetilde{\mathcal{R}}_r^1(\mathsf{pub}^1, x_1) \rangle$, with $\mathcal{S}_r$ and $\mathcal{R}_r$ acting as $\widetilde{\mathcal{S}}_r^1$ and $\widetilde{\mathcal{R}}_r^1$ respectively.

If $\widetilde{\mathcal{R}}_r^1$ rejects, or $f(x_1) \neq y$ or $c \oplus h(x_1) \neq b$, then $\mathcal{R}_r$ outputs `Reject`.

Otherwise, $\mathcal{R}_r$ outputs `Accept`.

In case $\mathsf{dec} = 1$**,**
Common input: 1^n, $b \in \{0,1\}$ *and* $\mathsf{pub} = (1, \mathsf{pub}^2)$.
Sender's private input: $\mathsf{prvt} = \mathsf{prvt}^2$.

$(\mathcal{S}_r, \mathcal{R}_r)$ run $\langle \widetilde{\mathcal{S}}_r^2(\mathsf{prvt}^2, \mathsf{pub}^2, b), \widetilde{\mathcal{R}}_r^2(\mathsf{pub}^2, b) \rangle$, with $\mathcal{S}_r$ and $\mathcal{R}_r$ acting as $\widetilde{\mathcal{S}}_r^2$ and $\widetilde{\mathcal{R}}_r^2$ respectively. $\mathcal{R}_r$ outputs the same output as $\widetilde{\mathcal{R}}_r^2$ does in the above execution.

The correctness of the above commitment scheme is evident given that the underlying two-phase commitment is correct. In Section 3.1.1, we prove that above scheme is statistically hiding whenever the underlying two-phase commitment is hiding. In Section 3.1.2, we prove that if $\mathcal{F}$ is a family of universal one-way hash functions and the underlying two-phase commitment is $\binom{2}{1}$-binding, then the above scheme is weakly binding.

Remark 3.2. *We note that by changing slightly the protocol of Construction 3.1, we could get a weakly-binding statistically-hiding commitment scheme for any polynomial length rather than for a mere bit. Since the proof of the current version is somewhat simpler, and since the shift from bit-commitment scheme to commitment scheme of any (polynomial) length is standard, we chose to present the above version.*

3.1.1 The scheme is hiding

Lemma 3.3. *If* $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ *is hiding, then* $(\mathcal{S}, \mathcal{R})$ *is statistically hiding.*

Proof. Assuming that $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ is hiding, then the hiding in the case that $\mathsf{dec} = 1$ is evident. That is, by the hiding of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$, no information about x_2 (and thus about b) has leaked to the receiver. Note that the receiver also gets the values of f and $f(x_1)$, but this information could be generated from x_1 and thus it reveals no additional information about x_2.

In the complementary case ($\mathsf{dec} = 0$) the situation is a bit more involved. Essentially, the only information that the receiver obtains about b is $y = f(x_1)$ and $c = b \oplus h(x_1)$. Since f is condensing and by the pairwise independence of $\mathcal{H}$, it is easy to see that with overwhelming probability (y, c) contains only negligible information about b and thus the protocol is hiding. Let us turn to the formal proof. Let $(\mathcal{S}', \mathcal{R}')$ be the same protocol as $(\mathcal{S}, \mathcal{R})$, but where in Line (2) of the commit stage, the first-phase commit of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$, is always excused with $\widetilde{\mathcal{S}}_c$'s input set to $0^{\ell(n)}$ (instead of x_1) and dec is always set to zero. Since $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ is hiding, $(\mathcal{S}', \mathcal{R}')$ is statistically hiding if and only if $(\mathcal{S}, \mathcal{R})$ is. Otherwise, one could have designed a statistical test that distinguishes a commitment to $0^{\ell(n)}$ from a commitment to a random x_1 (that is known to the test), which contradicts the hiding of the first phase of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$. Hence, for the following discussion we concentrate on the hiding property of the protocol $(\mathcal{S}'', \mathcal{R}'')$, where Line (2) is not executed at all (it is obvious that $(\mathcal{S}'', \mathcal{R}'')$ is statistically hiding if and only if $(\mathcal{S}', \mathcal{R}')$ is).

Let us fix a deterministic ITM $\mathcal{R}^*$ that interacts with $\mathcal{S}''_c$ in the commit stage of $(\mathcal{S}'', \mathcal{R}'')$, note that since we allow $\mathcal{R}^*$ to be unbounded, assuming that $\mathcal{R}^*$ is deterministic is without loss of generality. For a given value of n, it follows that since $\mathcal{R}^*$ is deterministic and it sends the hash function f as the first message of the interaction, f is the same in all interactions. We denote this value of f by f^*. The view of $\mathcal{R}^*$ when interacting with $\mathcal{S}''_c$ consists of the values of $y = f^*(x_1)$, h and $c = b \oplus h(x_1)$. Note that the only difference between a commitment to one and a commitment to zero is the value of c. Let v be a possible view of $\mathcal{R}^*$ in the interaction with $\mathcal{S}''_c$ and let h, y and c be the values of these variables in v. It follows that for both $b \in \{0,1\}$ $Pr[\mathsf{view}_{\langle \mathcal{S}''_c(b), \mathcal{R}^* \rangle}(n) = v] = \frac{1}{|\mathcal{H}|} \cdot \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[f^*(x_1) = y] \cdot \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[b \oplus h(x_1) = c \mid f^*(x_1) = y]$. Therefore,

$$
\begin{aligned}
& SD(\mathsf{view}_{\langle \mathcal{S}''_c(0), \mathcal{R}^* \rangle}(n), \mathsf{view}_{\langle \mathcal{S}''_c(1), \mathcal{R}^* \rangle}(n)) \\
= \; & \frac{1}{2} \sum_{v} \left| Pr[\mathsf{view}_{\langle \mathcal{S}''_c(0), \mathcal{R}^* \rangle}(n) = v] - Pr[\mathsf{view}_{\langle \mathcal{S}''_c(1), \mathcal{R}^* \rangle}(n) = v] \right| \\
= \; & \frac{1}{2} \cdot \frac{1}{|\mathcal{H}|} \sum_{y,h,c} \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[f^*(x_1) = y] \cdot \\
& \left| \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[0 \oplus h(x_1) = c \mid f^*(x_1) = y] - \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[1 \oplus h(x_1) = c \mid f^*(x_1) = y] \right| \\
= \; & \frac{1}{2} \cdot \frac{1}{|\mathcal{H}|} \sum_{y,h} \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[f^*(x_1) = y] \cdot \\
& 2 \cdot \left| \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[h(x_1) = 0 \mid f^*(x_1) = y] - \Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[h(x_1) = 1 \mid f^*(x_1) = y] \right| \\
= \; & \mathop{\mathsf{Ex}}_{x_1 \leftarrow \{0,1\}^{\ell(n)}, h \leftarrow \mathcal{H}} \left[\frac{\left| |(f^*)^{-1}(x_1) \cap h^{-1}(0)| - |(f^*)^{-1}(x_1) \cap h^{-1}(1)| \right|}{|(f^*)^{-1}(x_1)|} \right].
\end{aligned}
$$

The proof of Lemma 3.3 is concluded by the following claim and Lemma 2.2.

Claim 3.4. *For any $f \in \mathcal{F}$ it holds that* $\Pr_{x_1 \leftarrow \{0,1\}^{\ell(n)}}[\left|f^{-1}(f(x_1))\right| \le 2^{\frac{1}{4}\ell(n)}] \le 2^{-\frac{1}{4}\ell(n)}$.

Proof. For a given value of $f \in \mathcal{F}$, we say that $y \in \{0,1\}^{m(n)}$ is *light*, if $\left|f^{-1}(y)\right| < 2^{\frac{1}{4}\ell(n)}$. Clearly, f has at most $2^{m(n)}$ light images and therefore there are at most $2^{\frac{1}{4}\ell(n)} \cdot 2^{m(n)} \le 2^{\frac{3}{4}3\ell(n)}$ elements in $\{0,1\}^{\ell(n)}$ for which $\left|f^{-1}(f(x_1))\right| \le 2^{\ell(n)/4}$. □

3.1.2 The scheme is weakly binding

Lemma 3.5. *If $\mathcal{F}$ is a family of universal one-way hash functions and $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ is $\binom{2}{1}$-binding, then $(\mathcal{S}, \mathcal{R})$ is $\frac{17}{18}$-binding.*

Proof. Let $S^* = (S^*_c, S^*_r)$ be an algorithm trying to break the binding of $(\mathcal{S}, \mathcal{R})$ and recall $\mathsf{BndBreak}$ from Definition 2.6. Let $i \in \{0,1\}$ and let p be a positive polynomial, we define $\gamma_i^{S^*,p}(n) \overset{\text{def}}{=} \Pr_{\mathsf{outs} \leftarrow <S^*_c(1^n), \mathcal{R}_c(1^n)>}[\mathsf{BndBreak}^{S^*_r, \mathcal{R}_r}(\mathsf{outs}) > \frac{1}{p(n)} | \mathsf{dec} = i]$. Namely, $\gamma_i^{S^*,p}(n)$ is the probability that the output of the commit stage enables S^* to cheat in the reveal stage with noticeable probability. The proof of the Lemma 3.5 follows by the next claim.

Claim 3.6. *For any* PPT S^* *and any positive polynomial p, for large enough n there exists $i \in \{0,1\}$ such that $\gamma_i^{S^*,p}(n) < \frac{8}{9}$.*

Therefore, for any positive polynomial p and large enough n, $\Pr_{\mathsf{outs} \leftarrow <S^*(1^n), \mathcal{R}_c(1^n)>}[\mathsf{BndBreak}^{S^*, \mathcal{R}_r}(\mathsf{outs}) > \frac{1}{p(n)}] = \Pr[\mathsf{dec} = 0] \cdot \gamma_0^{S^*,p}(n) + \Pr[\mathsf{dec} = 1] \cdot \gamma_1^{S^*,p}(n) \le 1 - \frac{1}{2} \cdot \frac{1}{9}$, and the proof of Lemma 3.5 follows.

Proof. (of Claim 3.6) We assume toward a contradiction that the claim does not hold and prove that either the hardness of the universal one-way hash functions or the $\binom{2}{1}$-binding of the underlying two-phase commitment scheme are violated. More formally, let S^* be algorithm and p be a positive polynomial such that for infinitely many n's and for both values of $i \in \{0,1\}$, it holds that $\gamma_i^{S^*,p}(n) \ge \frac{9}{10}$. Assuming that the $\binom{2}{1}$-binding of the underlying bit-commitment scheme holds, we use S^* to construct an algorithm M^{S^*}, described next, that breaks with noticeable probability the hardness of the universal one-way hash functions. Recall that in order to break the hash function, M^{S^*} should first select a value x and then given a random hash function f, it needs to output another element $x' \neq x$ such that $f(x) = f(x')$.

Before presenting the algorithm, we would like first to make the dependency of S^*_c and $\mathcal{R}_c$ on the their random-coins explicit. That is, we assume that S^*_c and $\mathcal{R}_c$ are deterministic efficient algorithms that get as additional inputs random strings $rand_{S^*_c} \in \{0,1\}^{\ell_{S^*_c}(n)}$ and $(\mathsf{dec}, f, rand_{\mathcal{R}_c}) \in \{0,1\} \times \mathcal{F} \times \{0,1\}^{\ell_{\mathcal{R}_c}(n)}$ respectively. We assume w.l.o.g. that both $\ell_{S^*_c}$ and $\ell_{\mathcal{R}_c}$ are some known polynomials.

```
$M^{S^*}$:
First stage, selecting a value $x$.
Input: $1^n$
a Select uniformly at random $rand_{S_c^*} \in \{0,1\}^{\ell_{S_c^*}(n)}$, $rand_{\mathcal{R}_c} \in \{0,1\}^{\ell_{\mathcal{R}_c}(n)}$ and $f \in \mathcal{F}$.
b Simulate $\langle S_c^*(rand_{S_c^*}), \mathcal{R}_c(1, f, rand_{\mathcal{R}_c})\rangle$.
    Let $\mathsf{outs} = (\mathsf{prvt}, \mathsf{pub})$ be the private output of $S_c^*$ and the public output in the above
    simulation and let $\mathsf{outs}[x_1]$ be the value of $x_1$ in $\mathsf{pub}$ (see the commit stage of Construction 3.1
    for $\mathsf{dec} = 1$).
c Output $x = \mathsf{outs}[x_1]$ and $\mathsf{state} = (rand_{S_c^*}, rand_{\mathcal{R}_c})$.

Second stage, finding a collision.
Input: $x$, $\mathsf{state} = (rand_{S_c^*}, rand_{\mathcal{R}_c})$, $f' \in \mathcal{F}$
d Simulate $\langle S_c^*(rand_{S_c^*}), \mathcal{R}_c(0, f', rand_{\mathcal{R}_c})\rangle$.
    Let $\mathsf{outs}' = (\mathsf{prvt}', \mathsf{pub}')$ be the private output of $S_c^*$ and the public output in the above
    simulation.
e For both $i \in \{0,1\}$:
    Simulate $\langle S_r^*(\mathsf{prvt}', \mathsf{pub}', b), \mathcal{R}_r(\mathsf{pub}', i)\rangle$.
    Let $z_i$ be the value of the variable $x_1$ that $R_r$ gets from $S_r^*$ in the simulation (see the reveal
    stage of Construction 3.1 for $\mathsf{dec} = 0$).
f If $\mathcal{R}_r$ accepts for both $i \in \{0,1\}$, output $x' = z_j$, where $j \in \{0,1\}$ is such that $z_j \neq x$. (Note
    that since $\mathcal{R}_r$ accepts in both cases, it follows that $i = c \oplus h(z_i)$ for both $i \in \{0,1\}$ and thus
    $z_0 \neq z_1$).
```

Some intuition: *By the $\binom{2}{1}$-binding of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$, it follows that after the first-phase commit, there is only a* single *value, $\widetilde{x}$, such that if the first-phase commitment is "opened" to this value, it might be possible to cheat in the second-phase commitment. Since S^* manages to cheat (also) for* $\mathsf{dec} = 1$ *and therefore S^* is able to break the second-phase commitment of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$, it holds w.h.p. that x, defined in the first-stage of M^{S^*}, is equal to $\widetilde{x}$.*

Let us now consider the second-stage of M^{S^*}. Since S_c^* does not know the value of dec *when sending y in the simulation of Line (d), it should send y such that $y = f'(\widetilde{x})$ where y is the value sent by S_r^* to $\mathcal{R}_r$ after the first-phase commit. The point is that since we are using the same random coins as in the first stage, this is the same $\widetilde{x}$ as before. Whenever S^* breaks the commitment for* $\mathsf{dec} = 0$*, it needs to open the first-phase commitment into two elements $z_0 \neq z_1$ such that $f'(z_0) = f'(z_1) = y$. Thus, w.h.p. it holds that $f'(z_0) = f'(z_1) = f'(\widetilde{x})$ and M^{S^*} violates the hardness of $\mathcal{F}$.*

We now return to the formal proof. For any value of the parties random coins $frand = (rand_{S_c^*}, \mathsf{dec}, f, rand_{\mathcal{R}_c}) \in \{0,1\}^{\ell_{S_c^*}(n)} \times \{0,1\} \times \mathcal{F} \times \{0,1\}^{\ell_{\mathcal{R}_c}(n)}$, let $\mathsf{outs}(frand) \overset{\text{def}}{=} (\mathsf{prvt}(frand), \mathsf{pub}(frand))$, where $\mathsf{prvt}(frand)$ and $\mathsf{pub}(frand)$ are the private output of S_c^* and the public output in $\langle S_c^*(rand_{S_c^*}), \mathcal{R}_c((\mathsf{dec}, f, rand_{\mathcal{R}_c})\rangle$ respectively. The following lemma is the heart of our proof.

Lemma 3.7. *Assuming that $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ is $\binom{2}{1}$-binding and that Claim 3.6 does not hold w.r.t. $\mathcal{S}^*$, then there exists a set $L \subseteq \{0,1\}^{\ell_{S_c^*}(n)} \times \{0,1\}^{\ell_{\mathcal{R}_c}(n)}$ of density $\frac{1}{6}$ for which the following hold:*

1. *For all $(rand_{S_c^*}, rand_{\mathcal{R}_c}) \in L$ and any value of* $\mathsf{dec} \in \{0,1\}$,

$$\Pr_{f \leftarrow \mathcal{F}}[\mathsf{BndBreak}^{S_r^*, \mathcal{R}_r}(\mathsf{outs}(rand_{S_c^*}, \mathsf{dec}, f, rand_{\mathcal{R}_c})) \geq \frac{1}{p(n)}] \geq \frac{2}{3},$$

2. *There exists a mapping $\sigma : \{0,1\}^{\ell_{S_c^*}(n)} \times \{0,1\}^{\ell_{\mathcal{R}_c}(n)} \to \{0,1\}^{\ell(n)}$ s.t. for all $(rand_{S_c^*}, rand_{\mathcal{R}_c}) \in L$,*

$$\Pr_{f \leftarrow \mathcal{F}}[\mathsf{outs}(rand_{S_c^*}, 1, f, rand_{\mathcal{R}_c})[x_1] = \sigma(rand_{S_c^*}, rand_{\mathcal{R}_c})] \geq \frac{1}{2}.$$

We now conclude the proof of Claim 3.6 by using the above lemma to prove that if $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ is $\binom{2}{1}$-binding and Claim 3.6 does not hold w.r.t. $\mathcal{S}^*$, then M^{S^*} breaks the hardness of $\mathcal{F}$.

For $rand = (rand_{S_c^*}, rand_{\mathcal{R}_c}) \in \{0,1\}^{\ell_{S_c^*}(n)} \times \{0,1\}^{\ell_{\mathcal{R}_c}(n)}$ and $f \in \mathcal{F}$, consider the value of y that S_r^* sends to R_r (as the value of $f(x_1)$) in the execution of $\langle S_c^*(rand_{S_c^*}), \mathcal{R}_c(\mathsf{dec}, f, rand_{\mathcal{R}_c}\rangle$ for some value of $\mathsf{dec} \in \{0,1\}$. Note that y is sent before dec is made public and therefore its value depends only on $rand$ and f. We denote the value of y for a given values of $rand$ and f by $y(rand, f)$.

Note that whenever $\mathcal{R}_r$ accepts in the simulation of Line (e), it must hold that $f'(z_i) = y(\mathsf{state}, f')$. In addition, recall that if $\mathcal{R}_r$ accepts in the execution of M^{S^*} for both $i \in \{0,1\}$, then $z_0 \neq z_1$ (see the inline remark in Line (f)). Hence, conditioned that $\mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}') > \frac{1}{p(n)}$, we have that $\Pr[z_0 \neq z_1 \bigwedge f'(z_0) = f'(z_1) = y(\mathsf{state}, f)] \geq \frac{1}{p(n)^2}$. Let's assume now that after Line (d) algorithm M^{S^*} would execute the following line,

(d') Simulate $\langle S_c^*(rand_{S_c^*}), \mathcal{R}_c(1, f', rand_{\mathcal{R}_c})\rangle$.

Let $\mathsf{outs}'' = (\mathsf{prvt}'', \mathsf{pub}'')$ be the private and public outputs of the above simulation. Note that whenever $\mathsf{pub}'' \neq \perp$ (i.e., $\mathcal{R}_c$ did not abort the commit stage), it must hold that $f'(\mathsf{outs}''[x_1]) = y(\mathsf{state}, f)$. The main observation is that since S^* does not know the value of dec in advance, it must succeed with noticeable probability in Line (d) and in Line (d') *simultaneously*. Namely, Lemma 3.7 yields that $\Pr_{f' \leftarrow \mathcal{F}}[\mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}') > \frac{1}{p(n)} \bigwedge \mathsf{outs}''[x_1] = \sigma(\mathsf{state}) \mid \mathsf{state} \in L] \geq \frac{2}{3} - \frac{1}{2} = \frac{1}{6}$. Thus, by the above observations

$$\Pr_{f' \leftarrow \mathcal{F}}[z_0 \neq z_1 \bigwedge f'(\sigma(\mathsf{state})) = f'(z_0) = f'(z_1)] \geq \tfrac{1}{6} \cdot \tfrac{1}{p(n)^2}. \tag{1}$$

Let us return to the value x, that M^{S^*} outputs in its first stage. Applying Lemma 3.7 once more yields that

$$\Pr_{f \leftarrow \mathcal{F}}[x = \sigma(\mathsf{state}) \mid \mathsf{state} \in L] \geq \tfrac{1}{2}. \tag{2}$$

Since, conditioning on state, the events of Eq. 1 and Eq. 2 are independent, it follows that $\Pr[z_0 \neq z_1 \bigwedge f'(x) = f'(z_0) = f'(z_1) \mid \mathsf{state} \in L] \geq \frac{1}{12} \cdot \frac{1}{p(n)^2}$. Since L is noticeable, M^{S^*} breaks the hardness of $\mathcal{F}$ with noticeable probability.

Proof. (of Lemma 3.7) For both $i \in \{0,1\}$, let G_i be the set of random coins on which conditioned on $\mathsf{dec} = i$ S^* manages to break the binding with high probability. Namely, $G_i \overset{\text{def}}{=} \left\{(rand_{S_c^*}, rand_{\mathcal{R}_c}) : \Pr_{f \leftarrow \mathcal{F}}[\mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}(rand_{S_c^*}, i, f, rand_{\mathcal{R}_c}))) \geq \frac{1}{p(n)}] \geq \frac{2}{3}\right\}$. Since for both $i \in \{0,1\}$ we assumed that $\gamma_i^{S^*,p}(n) \geq \frac{9}{10}$, it follows by a straight forward averaging argument that for both $i \in \{0,1\}$ it holds that $\Pr[G_i] \geq \frac{2}{3}$ and therefore $G \overset{\text{def}}{=} G_0 \cap G_1$ is of density at least $\frac{1}{3}$. For any $x \in \{0,1\}^{\ell(n)}$ and $rand = (rand_{S_c^*}, rand_{\mathcal{R}_c})$, let $w^{rand}(x) \overset{\text{def}}{=} \Pr_{f \leftarrow \mathcal{F}}[\mathsf{outs}(rand_{S_c^*}, 1, f, rand_{\mathcal{R}_c})[x_1] = x]$.

Claim 3.8. $\Pr_{rand \leftarrow G}[\nexists x \in \{0,1\}^{\ell(n)}$ *s.t.* $w^{rand}(x) > \frac{1}{2}] = neg$.

Thus, we conclude the proof of Lemma 3.7, by letting $\sigma(rand) = \widetilde{x}$ if there exists $\widetilde{x}$ such that $w^{rand}(\widetilde{x}) > \frac{1}{2}$ and letting $\sigma(rand) = 0$ otherwise, and defining $L \overset{\text{def}}{=} G \cap \left\{rand : w^{rand}(\sigma(rand)) > \frac{1}{2}\right\}$.

Proof. (of Claim 3.8) For any random coins $frand = (rand_{S_c^*}, 1, f, rand_{\mathcal{R}_c}) \in \{0,1\}^{\ell_{S_c^*}(n)} \times \{0,1\} \times \mathcal{F} \times \{0,1\}^{\ell_{\mathcal{R}_c}(n)}$, let $\mathsf{trans}(frand)$ be the first-phase transcript of the interaction with $\widetilde{\mathcal{R}}$ embedded in the transcript of $\langle S_c^*(rand_{S_c^*}), \mathcal{R}_c((1, f, rand_{\mathcal{R}_c})\rangle$ (i.e., the transcripts of the interactions with $\widetilde{\mathcal{R}}_c^1$ and $\widetilde{\mathcal{R}}_r^1$). Recall the set $\mathcal{B}$ from Definition 2.10 w.r.t. $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$, which has the property that if a first-phase transcript of an interaction with $\widetilde{\mathcal{R}}$ is in $\mathcal{B}$, then the second-phase commitment with $\widetilde{\mathcal{R}}$ is statistically binding. It follows that for almost all $(rand_{S_c^*}, rand_{\mathcal{R}_c}) \in G$ (save but a set of negligible probability) it holds that,

$$\Pr_{f \leftarrow \mathcal{F}}[\mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}(rand_{S_c^*}, 1, f, rand_{\mathcal{R}_c})) \geq \frac{1}{p(n)} \bigwedge \mathsf{trans}(rand_{S_c^*}, 1, f, rand_{\mathcal{R}_c}) \notin \mathcal{B}] \geq \frac{2}{3} - neg(n).$$

Let's assume towards a contradiction that Claim 3.8 does not hold. Therefore, by the above observation there exists non-negligible set $G' \subseteq G$, such that the following holds for any $rand \in G'$:

1. $\nexists x \in \{0,1\}^{\ell(n)}$ s.t. $w^{rand}(x) > \frac{1}{2}$,
2. $\Pr_{f \leftarrow \mathcal{F}}[\mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}(rand_{S_c^*}, 1, f, rand_{\mathcal{R}_c}) \geq \frac{1}{p(n)} \bigwedge \mathsf{trans}(rand_{S_c^*}, 1, f, rand_{\mathcal{R}_c}) \notin \mathcal{B}]) \geq \frac{3}{5}$.

We conclude the proof, by showing that the above set implies violation of the $\binom{2}{1}$-binding of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$. Before doing that, we would like to make the dependence of $\mathcal{R}_c$ in its random coins even more explicit. Recall that we assume that $\mathcal{R}_c$ is a deterministic algorithm gets as additional input the random coins $(\mathsf{dec}, f, rand_{\mathcal{R}_c}) \in \{0,1\} \times \mathcal{F} \times \{0,1\}^{\ell_{\mathcal{R}_c}(n)}$. To make the discussion more precise, we write that $rand_{\mathcal{R}_c} = (rand_{\widetilde{\mathcal{R}}_c^1}, rand_{other})$ where $rand_{\widetilde{\mathcal{R}}_c^1} \in \{0,1\}^{\ell_{\widetilde{\mathcal{R}}_c^1}(n)}$ is the random-coins used in the execution of $\widetilde{\mathcal{R}}_c^1$ embedded in the execution of $\mathcal{R}_c$. The following algorithm breaks the $\binom{2}{1}$-binding of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$.

```
$T^{S^*}$:
Input: $1^n$

The interaction part.
a Select uniformly at random $rand_{S_c^*} \in \{0,1\}^{\ell_{S_c^*}(n)}$.
b Interact with $\widetilde{\mathcal{R}}_c^1(1^n)$ by invoking $S_c^*(rand_{S_c^*})$ and simulating its interaction with $\mathcal{R}_c$ by forwarding messages between $S_c^*$ and $\widetilde{\mathcal{R}}_c^1$.
    Let $\mathsf{trans}^1$ be the transcript of the above interaction and let $rand_{\widetilde{\mathcal{R}}_c^1}$ be the random coins used by $\widetilde{\mathcal{R}}_c^1$ in the above interaction. (We do not need to actually know the value of $rand_{\widetilde{\mathcal{R}}_c^1}$ for the run of $T^{S^*}$ and only use it in order to simplify notation.)

Producing two transcripts.
a Select uniformly at random $rand_{other} \in \{0,1\}^{\ell_{\mathcal{R}_c^1}(n) - \ell_{\widetilde{\mathcal{R}}_c^1}(n)}$.
b For $i \in \{0,1\}$:
        1. Select uniformly at random $f_i \in \mathcal{F}$.
        2. Simulate $\langle S_c^*(rand_{S_c^*}), \mathcal{R}_c(1, f_i, rand_{\widetilde{\mathcal{R}}_c}, rand_{other})\rangle$ starting from Line 3 of Construction 3.1 (note that given $\mathsf{trans}^1$, we do not need to know $rand_{\widetilde{\mathcal{R}}_c}$ in order to simulate).
           Let $\mathsf{outs}_i^2 = (\mathsf{prvt}_i^2, \mathsf{pub}_i^2)$, where $\mathsf{prvt}_i^2$ and $\mathsf{prvt}_i^2$ are the private output of $S_c^*$ and the public output in the above simulation respectively. Let $\mathsf{trans}_i^2$ and $\mathsf{trans}_i^3$ be the transcripts of the interactions with $\widetilde{\mathcal{R}}_r^1$ and $\widetilde{\mathcal{R}}_c^2$ in the above simulation.
        3. Simulate $\langle S_r^*(\mathsf{prvt}_i^2, \mathsf{pub}_i^2, 0), \mathcal{R}_r(\mathsf{pub}_i^2, 0)\rangle$.
           Let $\mathsf{trans}_i^4$ be the transcript of the interaction with $\widetilde{\mathcal{R}}_r^2$ in the above simulation.
        4. Set $\mathsf{trans}_i = (\mathsf{trans}^1, \mathsf{trans}_i^2, \mathsf{trans}_i^3, \mathsf{trans}_i^4)$.
c Output $(\mathsf{trans}_0, \mathsf{trans}_1)$.
```

Claim 3.9. *T^{S^*} breaks the $\binom{2}{1}$-binding of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$ with non-negligible probability.*

Proof. Conditioned on $rand = (rand_{S_c^*}, rand_{\widetilde{\mathcal{R}}_c^1}, rand_{other}) \in G'$, we have by the second property of G'

$$\Pr_{f_0 \leftarrow \mathcal{F}}[\mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}_0) \geq \tfrac{1}{p(n)} \bigwedge (\mathsf{trans}^1, \mathsf{trans}_0^2) \notin \mathcal{B}] \geq \tfrac{3}{5}. \tag{3}$$

Clearly, the above also holds w.r.t. f_1, outs_1 and trans_1^2. Moreover, by the first property of G', we have the following w.r.t. any $z \in \{0,1\}^{\ell(n)}$,

$$\Pr_{f_1 \leftarrow \mathcal{F}}[\mathsf{outs}_1[x_1] \neq z \bigwedge \mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}_1) \geq \tfrac{1}{p(n)} \bigwedge (\mathsf{trans}^1, \mathsf{trans}_1^2) \notin \mathcal{B}] \geq \tfrac{3}{5} - \tfrac{1}{2} = \tfrac{1}{10}. \tag{4}$$

Setting $z = \mathsf{outs}_0[x_1]$, since f_1 is independent of f_0, it follows that

$$\Pr_{f_0 \leftarrow \mathcal{F}, f_1 \leftarrow \mathcal{F}}[\mathsf{outs}_0[x_1] \neq \mathsf{outs}_1[x_1] \bigwedge \forall i \in \{0,1\}\ \mathsf{BndBreak}^{S_r^*,\mathcal{R}_r}(\mathsf{outs}_i) \geq \frac{1}{p(n)} \bigwedge (\mathsf{trans}^1, \mathsf{trans}_i^2) \notin \mathcal{B}]$$
$$\geq \frac{3}{5} \cdot \frac{1}{10} = \frac{3}{25}.$$

Therefore, we conclude that condition that $rand \in G'$, the following happens with probability at list $\frac{3}{25} \cdot \frac{1}{p(n)^2}$:

1. both trans_0 and trans_1 starts with trans^1,
2. the first-phase transcripts (i.e., $(\mathsf{trans}^1, \mathsf{trans}_i^2)$) in both trans_0 and trans_1 are not in $\mathcal{B}$,
3. the value of x_1 in trans_0 and in trans_1 is different,
4. $\widetilde{\mathcal{R}}_r^1$ and $\widetilde{\mathcal{R}}_r^2$ accept in both trans_0 and trans_1.

Since we assume that G' is non-negligible, T^{S^*} breaks the $\binom{2}{1}$-binding of $(\widetilde{\mathcal{S}}, \widetilde{\mathcal{R}})$. $\square$

Thus we have concluded the proof of Lemma 3.7 and thus the proof of Lemma 3.5. $\square$

3.2 Completing the construction

The following corollary follows by the lemmata about Construction 3.1 (Lemma 3.1 and Lemma 3.5) and the standard bit-commitment binding amplification (Proposition 2.9).

Corollary 3.10. *There exists an efficient procedure that given a family of universal one-way hash functions and a two-phase commitment scheme, outputs a bit-commitment scheme which is statistically hiding whenever the underlying protocol is hiding and it is computationally binding whenever the underlying protocol is* $\binom{2}{1}$*-binding.*

By the above Corollary, the existence of universal one-way hash functions ([Rom90, KK05]), the existence of a collections of two-phase commitment schemes that are all $\binom{2}{1}$-binding and at least one of them is hiding (Theorem 2.13) and the standard bit-commitment hiding amplification (Proposition 2.8). It follows that statistical bit-commitment can be constructed using any one-way function. Finally, the proof of Theorem 1.1 follows by the above conclusion and the standard transformation of a bit-commitment scheme into a commitment scheme of any polynomial length.

Remark 3.11. *Note that since the reveal stage of the commitments guaranteed by Theorem 2.13 are non-interactive (i.e., consistent on a single message from the sender to the receiver), the reveal stage of our bit-commitment is non-interactive as well.*

Acknowledgments

We are grateful to Danny Harnik, Oded Goldreich, Tal Moran, Moni Naor, Alon Rosen and Salil Vadhan for helpful conversations.

References

[BCC88] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *J. Computer and System Sciences*, 37(2):156–189, 1988.

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

[Blu82] M. Blum. Coin flipping by phone. In *IEEE COMPCOM*, 1982.

[CW77] I. Carter and M. Wegman. Universal classes of hash functions. In *9th ACM Symposium on Theory of Computing*, pages 106–112, 1977.

[DPP93] I. Damgård, T. Pedersen, and B. Pfitzmann. On the existence of statistically-hiding bit commitment and fail-stop signatures. In *Crypto*, 1993.

[GK96] O. Goldreich and A. Kahan. How to construct constant-round zero-knowledge proof systems for NP. *Journal of Cryptology*, 9(2):167–189, 1996.

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

[Gol01] O. Goldreich. Randomized methods in computation - lecture notes. 2001.

[HHK$^+$05] Haitner, Horvitz, Katz, Koo, Morselli, and Shaltiel. Reducing complexity assumptions for statistically-hiding commitment. In *EUROCRYPT: Advances in Cryptology: Proceedings of EUROCRYPT*, 2005.

[HILL99] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal of Computing*, 29(4):1364–1396, 1999.

[HM96] S. Halevi and S. Micali. Practical and provably-secure commitment schemes from collision-free hashing. In *Crypto*, 1996.

[IL89] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. pages 230–235, 1989.

[KK05] J. Katz and C. Koo. On constructing universal one-way hash functions from arbitrary one-way functions. Cryptology ePrint Archive, Report 2005/328, 2005. http://eprint.iacr.org.

[Lin03] Y. Lindell. Parallel coin-tossing and constant-round secure two-party computation. *Journal of Cryptology*, 16(3):143–184, 2003.

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

[NOV06] M. Nguyen, S. Ong, and S. Vadhan. Statistical zero-knowledge arguments for NP from any one-way function. Electronic Colloquium on Computational Complexity (ECCC), TR06-075, 2006.

[NOVY98] M. Naor, R. Ostrovsky, R. Venkatesan, and M. 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] M. Nguyen and S. Vadhan. Zero knowledge with efficient provers. In *Proceedings of the 38th ACM Symposium on Theory of Computing*, 2006.

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

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