Round-optimal black-box statistically binding selective-opening secure commitments

David Xiao*

LIAFA, CNRS, Université Paris 7
dxiao@liafa.univ-paris-diderot.fr,

Abstract. Assuming t-round statistically hiding commitments in the stand-alone model, we build a $(t+2)$-round statistically binding commitment secure against selective opening attacks under parallel composition. In particular, assuming collision-resistant hash functions, we build such commitments in 4 rounds.

Keywords: Commitments, selective opening attacks

1 Introduction

Selective opening attacks against commitment schemes occur when the commitment scheme is repeated in parallel and an adversary can choose depending on the commit-phase transcript to see the values and openings to some subset of the committed bits. Commitments are secure under such attacks if one can prove that the remaining, unopened commitments stay secret. Related notions such as chameleon blobs, equivocal commitments, and trapdoor commitments have been widely studied in the literature [BCC88, BCY89, Fis01, Bea96, DCIO98, DCO99]. The notion of selective opening security that we study here was defined by [DNRS03]. One of the primary motivations of studying such commitment schemes is their application to parallel composition of zero knowledge: when used as the commitment scheme in, say, the zero knowledge protocol of [GMW86], that protocol remains zero knowledge under parallel composition (which is not known to be the case when using a commitment scheme without selective opening attack security).

[BHY09, Xia11] studied the optimal round complexity for commitments secure against selective opening attacks. [Xia11] claimed round-optimal constructions under parallel composition, but it was subsequently shown in [ORSV11] that there were flaws in the argument of [Xia11]. In particular, [ORSV11] gave a 3-round construction in the case of computationally binding (and statistically hiding) commitments secure against selective opening attacks. The (corrected) lower bound of [Xia11] states that this is optimal (for black-box simulation).

[ORSV11] leave open the question of round-optimal black-box constructions of *statistically binding commitments* secure against selective opening attacks. The statistically binding commitment of [Xia11] is 5 rounds, but in light of the flaw discovered by [ORSV11], the lower bound is 4 rounds.

* Partially supported by the French ANR Defis program under contract ANR-08-EMER-012 (QRAC project).

Our contribution: in this paper we construct a $(t+2)$-round scheme that is secure under parallel composition assuming the existence of t-round stand-alone statistical hiding commitments (SHC). In particular, 2-round SHC can be built from collision-resistant hash functions [DPP93, DPP98, HM96], which gives an optimal 4-round construction of statistically binding selective-opening attack secure commitments.

Our analysis introduces a novel simulation strategy that generalizes the Goldreich-Kahan simulation strategy for constant-round zero knowledge [GK90]. At a high level, our strategy differs from the Goldreich-Kahan simulation strategy because it allows the simulator to continue even if the receiver aborts individual sessions, and it guarantees that the simulator's output distribution will be indistinguishable from the distribution in an interaction with the honest sender even when taking into account the aborted sessions. In contrast, the Goldreich-Kahan simulation strategy completely aborts if any individual session is aborted.

2 Preliminaries

We adopt the following standard notation: for a distribution $\mathcal{D}$ and a variable x, $x \leftarrow_R D$ denotes that x is sampled according to D. For distributions $\mathcal{D}_0, \mathcal{D}_1$, we let $\Delta(\mathcal{D}_0, \mathcal{D}_1)$ denote their statistical distance. We say that a function $\varepsilon(n)$ is negligible if $\varepsilon(n) \leq n^{-\omega(1)}$. For a bit b, $\bar{b}$ denotes the complement of b. We frequently use underlined variables to represent vectors, *e.g.* $\underline{b} \in \{0,1\}^k$ and $b_i \in \{0,1\}$ for every $i \in [k]$.

A *commitment protocol* is given by a pair of interactive algorithms Send and Rec. Both algorithms take an input indicating the phase (either *com* or *decom*) and a security parameter 1^n, which we often omit. Send takes a one-bit input and both algorithms may be randomized.

Commit phase: Generate a transcript $\tau \leftarrow_R \langle \mathsf{Rec}(com), \mathsf{Send}(com, b) \rangle$. Send also generates an internal state variable σ.

Decommit phase: Generate $(v, b') \leftarrow_R \langle \mathsf{Rec}(decom, \tau), \mathsf{Send}(decom, b, \sigma) \rangle$, where v is the receiver's view (including the entire transcript and its random coins) and $b' \in \{0, 1, \perp\}$, where $\perp$ denotes that the receiver rejects the sender's opening.

We will often omit the phase variable (*i.e.* either *com*, *decom*) and the state variable σ when it is convenient and their values are implicitly defined by the context. In this paper the round complexity refers only to the number of rounds in the commit phase, and we work only with commitments with non-interactive openings (*i.e.* the opening consists of a single message from sender to receiver).

We will study commitments under parallel composition, *i.e.* the commitment is executed many times simultaneously and, for each i, the i'th step of the commitment is finished in all sessions before the $i+1$'th step begins in any session.

Definition 1 (Binding). *Define* $\mathsf{Adv}_{\mathsf{bind}}$ *to be the supremum over all possible strategies* Send^* *(computationally unbounded) that the probability of the following experiment succeeds:*

1. *Generate* $\tau \leftarrow_R \langle \mathsf{Rec}(com), \mathsf{Send}^*(com) \rangle$ *along with sender state* σ.

2. *Generate* $(v_0, b_0) \leftarrow_R \langle \mathsf{Rec}(decom, \tau), \mathsf{Send}^*(decom, 0, \sigma) \rangle$ *and* $(v_1, b_1) \leftarrow_R \langle \mathsf{Rec}(decom, \tau), \mathsf{Send}^*(decom, 1, \sigma) \rangle$.
3. *The experiment is a success if* $b_0 = 0$ *and* $b_1 = 1$.

We say that $(\mathsf{Rec}, \mathsf{Send})$ *is* statistically binding *if* Adv_{bind} *is negligible. We say it is* perfectly binding *if* $\mathsf{Adv}_{bind} = 0$. *We say it is* computationally binding *if the* Adv_{bind} *is negligible when the supremum is taken over polynomial-size* Send^*.

The binding property is preserved under parallel composition.

Definition 2 (Statistically hiding (stand-alone)). *A commitment* $\mathsf{Send}, \mathsf{Rec}$ *is (stand-alone) statistically hiding (*i.e. *it is a SHC) if for all (possibly unbounded)* Rec^**, it holds that* $\Delta(\mathcal{D}_0, \mathcal{D}_1)$ *is negligible, where* $b \in \{0, 1\}$ *and* $\mathcal{D}_b$ *denotes the distribution of* $\tau \leftarrow_R \langle \mathsf{Rec}^*(com), \mathsf{Send}(com, b) \rangle$ *along with the private coins of* Rec^*.

Definition 3 (Hiding under selective opening attacks). *A commitment is* secure against selective opening attacks *with black-box simulation if for all* $k = \mathrm{poly}(n)$*, there is an efficient simulator* Sim_k *such that the following holds. Define* Adv_{hide} *to be the supremum over all polynomial-size cheating receiver strategies* Rec^**, all polynomial-size distinguisher circuits* D*, all inputs* $\underline{b}$*, of the difference between the probability that* D *outputs* 1 *in the following two experiments (in the following,* $I \subseteq [k]$ *and* $\underline{b}_I$ *denotes the vector containing* b_i *for* $i \in I$*, and likewise for* $\underline{\tau_I}, \underline{sigma_I}$*):*

1. *Let* $\underline{\mathsf{Send}}$ *denote* k *parallel instances of the sender algorithm,* $\mathsf{Send}_1, \ldots, \mathsf{Send}_k$.
   (a) *Generate* $\tau_i \leftarrow_R \langle \mathsf{Rec}^*(com), \underline{\mathsf{Send}}(com, b_i) \rangle$ *along with sender state* σ_i *for all* $i \in [k]$.
   (b) Rec^* *outputs a set* $I \subseteq [k]$.
   (c) *Generate* $(v, \underline{b}_I) \leftarrow_R \langle \mathsf{Rec}^*(decom, \underline{\tau_I}), \underline{\mathsf{Send}}(decom, \underline{b}_I, \underline{\sigma_I}) \rangle$.
   (d) *Output* $D(v, \underline{b}_I)$.
2. Sim_k *samples random coins for* Rec^* *and fixes them; in the following* Sim_k *queries* Rec^* *for this fixed choice of coins.*
   (a) *Generate* $I \leftarrow_R \mathsf{Sim}_k^{\mathsf{Rec}^*}(com)$.
   (b) *Generate* $v \leftarrow_R \mathsf{Sim}_k^{\mathsf{Rec}^*}(decom, I, \underline{b}_I)$.
   (c) *Output* $D(v, \underline{b}_I)$.

Security against selective opening attacks holds if Adv_{hide} *is negligible.*

This definition is stronger than necessary for many applications (where security is only needed with respect to certain message distributions and certain families of valid subsets to be opened). However since we only study constructions in this paper, working with this definition is stronger than working with weaker definitions.

3 Construction

Let $\mathsf{Send}_{\mathsf{SH}}, \mathsf{Rec}_{\mathsf{SH}}$ be the sender and receiver algorithms for a t-round statistically hiding bit commitment. A construction for $t = 2$ exists based on collision-resistant hash functions [DPP93, DPP98, HM96]. Let $\mathsf{Send}_{\mathsf{NI}}, \mathsf{Rec}_{\mathsf{NI}}$ be the sender and receiver algorithms for a non-interactive perfectly binding commitment (*e.g.* based on one-way permutations). Our commitment is given in Algorithm 1. The following two lemmas prove the security of the commitment.

Send's input: $b \in \{0,1\}$.

Commitment phase

1. Rec samples $\underline{\beta} \leftarrow_R \{0,1\}^n$ and commits to each bit in parallel using $(\mathsf{Send}_{\mathsf{SH}}, \mathsf{Rec}_{\mathsf{SH}})$ (namely, Rec plays the role of $\mathsf{Send}_{\mathsf{SH}}$ and Send plays the role of $\mathsf{Rec}_{\mathsf{SH}}$). Let c_i denote the transcript of the commitment to β_i. If any c_i is not well-formed, the protocol aborts.
2. Define $M(b,\eta) = \begin{pmatrix} b\ \eta \oplus b \\ b\ \eta \oplus \bar{b} \end{pmatrix}$
   For each $i = 1,\ldots,n$ in parallel, Send samples $\eta_i \leftarrow_R \{0,1\}$. In parallel, Send uses $\mathsf{Send}_{\mathsf{NI}}$ to generate a commitment to all bits of $M(b,\eta_i)$. Call this commitment d_i. Send sends d_i to Rec. If any d_i is not well-formed, Rec aborts.
3. For each $i \in [n]$ in parallel, Rec generates an opening ϕ_i to c_i and sends it to Send. Send calculates $\beta_i = \mathsf{Rec}_{\mathsf{SH}}(\mathit{decom}, c_i, \phi_i)$. If any $\beta_i = \perp$, Send aborts.

Opening phase

1. Send sends b to Rec, and for each $i \in [n]$, Send opens the bits in d_i that correspond to the row in $M(b,\eta_i)$ that equals $(b, \beta_i \oplus b)$.
2. For each $i = 1,\ldots,n$, Rec computes the two bits in the row opened by Send, call these (x_i^0, x_i^1). Rec checks that $x_i^0 = b$ and $x_i^1 = b \oplus \beta_i$. If the check fails, then Rec rejects and outputs $\perp$, otherwise Rec outputs b.

Algorithm 1 *4-round statistically binding and selective opening attack secure commitment*

Lemma 1. *Algorithm 1 is statistically binding.*

Lemma 2. *The simulator given in Algorithm 2 proves that Algorithm 1 is secure against selective opening attacks.*

Proof (of Lemma 1). Fix any Send^* a (possibly cheating and computationally unbounded) sender strategy. Let $(\underline{c}, \underline{d}, \underline{\phi}) \leftarrow_{\mathrm{R}} \langle \mathsf{Rec}(\underline{\beta}), \mathsf{Send}^* \rangle$ be the commit-phase transcript: $\underline{c}$ denotes the n parallel SHC to $\underline{\beta}$ sent by Rec, $\underline{d}$ denotes the n non-interactive commitments to matrices $m_1, \ldots, m_n$ sent by Send^* in response to $\underline{c}$, and $\underline{\phi}$ denotes Rec's opening of $\underline{c}$. Note that given $\underline{d}$, $\underline{m} = (m_1, \ldots, m_n)$ are well-defined because the non-interactive commitment is perfectly binding.

We say that a matrix m_i *matches* a bit β_i if it holds that the bits in the first column are different and the XOR of the two bits in each row equals β_i. Formally, this holds if $m_i^{0,0} \neq m_i^{1,0}$ and also $m_i^{j,0} = m_i^{j,1} \oplus \beta_i$ for all $j \in \{0,1\}$. We say that $\underline{m}$ matches $\underline{\beta}$ if for all $i \in [n]$ it holds that m_i matches β_i.

Claim. Let $(\underline{c}, \underline{d}, \underline{\phi})$ denote a valid commit-phase transcript. Let $\underline{\beta}$ be the opening of $\underline{c}$ using $\underline{\phi}$, and let $\underline{m}$ be the opening of $\underline{d}$. Given this commit-phase transcript, Send^* can successfully break binding iff $\underline{m}$ matches $\underline{\beta}$.

Proof. Suppose $\underline{m}$ matches $\underline{\beta}$, then for each $i \in [n]$ there exists a row in m_i that equals $(0, \beta_i)$ and a row that equals $(1, \overline{\beta_i})$. Therefore, it is possible for Send^* to generate an opening to the row that equals $(b, b \oplus \beta_i)$ for all $b \in \{0,1\}$.

Suppose that Send^* can open to both values of $b \in \{0,1\}$. This means it can open each d_i to $(b, b \oplus \beta_i)$ for both $b \in \{0,1\}$. Since $\mathsf{Send}_{\mathsf{NI}}$ is perfectly binding, therefore it must be that $\underline{m}$ matches $\underline{\beta}$. ∎

A standard argument says that soundness holds because the initial commitment to $\underline{\beta}$ by Rec does not contain any information about $\underline{\beta}$, therefore it is impossible for Send^* to commit to $\underline{m}$ that will match $\underline{\beta}$.

For the sake of completeness, we give a formal proof: by the statistical hiding property of the initial commitment, we have that for any $\underline{\beta}$, it holds that $(\underline{c}, \underline{d}) \leftarrow_{\mathrm{R}} \langle \mathsf{Rec}(\underline{\beta}), \mathsf{Send}^* \rangle$ and $(\underline{c}, \underline{d}) \leftarrow_{\mathrm{R}} \langle \mathsf{Rec}(0^n), \mathsf{Send}^* \rangle$ are $n^{-\omega(1)}$-close in statistical distance. Therefore we may write:

$$
\begin{aligned}
\mathsf{Adv}_{\mathsf{bind}} &\leq \Pr_{\substack{\underline{\beta} \leftarrow_{\mathrm{R}} \{0,1\}^n \\ (\underline{c}, \underline{d}, \underline{\phi}) \leftarrow_{\mathrm{R}} \langle \mathsf{Rec}(\underline{\beta}), \mathsf{Send}^* \rangle}} [\underline{d} \text{ opens to } \underline{m} \text{ matching } \underline{\beta}] \\
&= \sum_{\underline{\beta} \in \{0,1\}^n} 2^{-n} \Pr_{(\underline{c}, \underline{d}, \underline{\phi}) \leftarrow_{\mathrm{R}} \langle \mathsf{Rec}(\underline{\beta}), \mathsf{Send}^* \rangle} [\underline{d} \text{ opens to } \underline{m} \text{ matching } \underline{\beta}] \\
&\leq 2^{-n} \sum_{\underline{\beta} \in \{0,1\}^n} \Pr_{(\underline{c}, \underline{d}, \underline{\phi}) \leftarrow_{\mathrm{R}} \langle \mathsf{Rec}(0^n), \mathsf{Send}^* \rangle} [\underline{d} \text{ opens to } \underline{m} \text{ matching } \underline{\beta}] + n^{-\omega(1)} \\
&\leq 2^{-n} + n^{-\omega(1)}
\end{aligned}
$$

where the last inequality holds because if $\underline{m}$ matches one $\underline{\beta}$, it cannot match any $\underline{\beta}' \neq \underline{\beta}$, and so the sum of the probabilities is bounded by 1. This means that Send^* has a negligible probability of breaking binding. ∎

4 Analyzing the simulator

Proof (of Lemma 2). We use "initial commitment" to denote the SHC used by the receiver in the first step of the protocol. We use "final commitment" to denote the overall selective-opening attack secure commitment we are trying to simulate. As is typical with black-box simulation strategies, this simulator tries to rewind Rec^* to discover the values in the initial commitment, and then use those values to construct a final commitment that can be opened to both 0 and 1. One subtlety where care is need is the possibility of individual sessions aborting. We observe that, for each session j that successfully completes the initial commitment, the simulator needs to successfully discover the committed $\underline{\beta}^j$ just once (the receiver cannot change it, otherwise this would contradict the binding property of the initial commitment). Once $\underline{\beta}^j$ is discovered, the simulator can always successfully open the final commitment to both 0 and 1 in session j.

The idea is to successively increase the number of sessions where the simulator knows $\underline{\beta}^j$, so that eventually Rec^* will be non-aborting only in sessions where the simulator can open to any value and therefore the simulation can be successfully terminated. Care must be taken that one does not bias the distribution of non-aborting sessions in the final transcript. The intuition is the following strategy: suppose at some point we know that how to reveal arbitrary values for some set of sessions X. The next time the simulator queries the receiver, if the set S of non-aborting sessions returned satisfies $S \subseteq X$ then the simulator can successfully open any subset of S that the receiver requests and so we can terminate the simulation. Otherwise, $S \not\subseteq X$ and so we have increased the number of sessions where we know $\underline{\beta}^j$, and so in future samples we have a better chance of being able to open all the non-aborting sessions to both 0 and 1. However, in order not to bias the distribution of opened sessions, in future iterations, if the simulator receives a response from the receiver whose non-aborting sessions are contained in X, we ignore it and resample. The actual simulator follows this intuition, although there are details about how exactly to query the receiver that need to be taken care of.

As in the case of constant-round zero knowledge, one point we must be careful about is that the simulator must run in expected polynomial time. This is done by using the Goldreich-Kahan strategy of estimating the success probability of queries, and then setting a timeout based on this estimate. In the case of Goldreich-Kahan "success" means obtaining a response without aborting sessions, while in our case success means that the set of non-aborting sessions lies inside X but not inside Y, where $Y \subsetneq X$ are subsets that evolve during the course of simulation.

4.1 General observations.

The simulator is given in Algorithm 2. In the following we omit k from the notation and write simply Sim. We can divide the simulator algorithm into two parts: the initial commitment where the receiver commits to some $\underline{\beta}^j$ (consisting of all the steps up to Step 3), and the remainder. We will frequently analyze the simulator for a fixed value of Rec^*'s random coins and a fixed initial commitment transcript, since this part is executed exactly once and is distributed identically to the honest interaction.

Given oracle access to a cheating k-fold receiver Rec^*:

1. Initialize $X, Y = \varnothing$. Initialize variables $\underline{\beta}^1, \ldots, \underline{\beta}^k$ to empty. Initialize a counter t to 0 and a timeout T to 0.
2. Sample random coins for Rec^* and fix them. Sample coins for the honest sender and execute the initial commitment with Rec^*. Write Rec^*'s random coins and the initial commitment phase transcript to the output.
3. Let $\Sigma \subseteq [k]$ denote the set of sessions in which Rec^* does not abort in the initial commitment. In the following, only continue interaction in Σ.
4. In the following, if Rec^* ever outputs an invalid opening of a commitment in session j, the simulator interprets this as the receiver aborting in session j. The simulator also checks the values of all the valid openings, and if Rec^* ever opens the same commitment to two distinct values then the simulator outputs "binding broken" and halts.
5. Define $F(\gamma, \beta) = \begin{pmatrix} \gamma & \beta \oplus \gamma \\ \overline{\gamma} & \beta \oplus \overline{\gamma} \end{pmatrix}$.
6. *First loop:* Repeat the following:
   (a) *Dummy commitments:* For each $j \in \Sigma, i \in [n]$, sample $\gamma_i^j \leftarrow_R \{0,1\}, \nu_i^j \leftarrow_R \{0,1\}$ and generate commitments to $F(\gamma_i^j, \nu_i^j)$. Call these commitments $\underline{d}^j = (d_1^j, \ldots, d_n^j)$. Send $\underline{d}^j$ to Rec^*.
   (b) Read Rec^*'s response, call this s. Let $S \subseteq \Sigma$ be the set of non-aborting sessions in s. Do the following:
      i. If $S = X = Y = \varnothing$ (this can only occur in the first iteration), write the $\underline{d}^j$ and s to the output and halt.
      ii. If $S \subseteq Y$, continue the loop.
      iii. If $S \not\subseteq Y$ and $S \subseteq X$ then break the loop.
      iv. If $S \not\subseteq X$ then set $Y \leftarrow X$, $X \leftarrow X \cup S$, and for all $j \in S \setminus X$, set $\underline{\beta}^j$ to be the value that was opened by Rec^*. Continue the loop.
7. *Calculate timeout:* Repeat the following trial until $(nk)^2$ successes occur: for each $j \in \Sigma$, generate $\underline{d}^j$ by the method in Step 6a, and let S' denote the set of sessions in Rec^*'s response that are not aborted; the trial is a success if $S' \not\subseteq Y$ and $S' \subseteq X$. Let ℓ denote the number of repetitions that were used to obtain $(nk)^2$ successes. Set $T = \min(\frac{\ell}{nk}, nk2^{nk})$ and set $t = 0$.
8. *Second loop:* Repeat the following while $t \leq T$
   (a) For $j \in \Sigma$, construct and send $\underline{d}^j$ to the receiver, defined as:
      i. For each $j \in \Sigma \setminus X$, let $\underline{d}^j$ be generated by the method in Step 6a.
      ii. For $j \in X$ and for each $i \in [n]$, sample $\gamma_i^j \leftarrow_R \{0,1\}$ and construct d_i^j to be a commitment to $F(\gamma_i^j, \beta_i^j)$.
   (b) Let s be Rec^*'s response and S the set of non-aborted sessions in s.
      i. If $S \subseteq Y$ or $S \not\subseteq X$ then increment t and continue the loop.
      ii. Otherwise, it must be that $S \not\subseteq Y$ and $S \subseteq X$. Write all the $\underline{d}^j$ and s to the output. Complete the simulation as follows:
         A. Ask Rec^* for a set I to be opened. If Rec^* aborts, then the simulator halts. Otherwise, Rec^* picks a subset $I \in \mathcal{I}, I \subseteq S$ to be revealed and the simulator asks for the values $\{b_j\}_{j \in I}$. Write I to the output.
         B. For each $j \in I$, each $i \in [n]$, the simulator outputs b_j and an opening to the row in $F(\gamma_i^j, \beta_i^j)$ that equals $(b_j, \beta_i^j \oplus b_j)$.
         C. Halt.
9. We exceeded the timeout, so output "timeout".

Algorithm 2 *Simulator* Sim_k *for Algorithm 1*

Input: black-box access to a distribution $\mathcal{D}$ over $[k]$.

1. Initialize $X = Y = \varnothing$.
2. Repeat the following:
   (a) Sample $S \leftarrow_R \mathcal{D}$. If $S = X = Y = \varnothing$, output S and halt.
   (b) If $S \subseteq Y$, continue the loop.
   (c) If $S \not\subseteq Y$ and $S \subseteq X$, output S.
   (d) If $S \not\subseteq X$, then we have seen some new elements ($S \setminus X$). Set $Y \leftarrow X$ and $X \leftarrow X \cup S$ and continue the loop.

Algorithm 3 *Abstraction of simulator*

Fix any choice of Rec^*'s random coins and the initial commitment transcript, which in turn fixes some $\Sigma \subseteq [k]$ of non-aborting sessions so far. This defines a distribution $\mathcal{D}_{\mathsf{dummy}}$ as follows: construct dummy commitments $\underline{d}^j$ for $j \in \Sigma$ as in Step 6a and send these to Rec^*, and let s denote the receiver's response. Let $S = S(s)$ denote the set of sessions where s contains a non-aborting response (*i.e.* in those sessions, Rec^* produces a valid opening of the initial commitment). Let $\mathcal{D}_{\mathsf{dummy}}$ denote the distribution over S thus sampled.

For $X \subseteq \Sigma$, let q_X denote

$$q_X = \Pr_{S \leftarrow_R \mathcal{D}_{\mathsf{dummy}}}[S \not\subseteq X] \tag{4.1}$$

Observe that $q_\varnothing$ is the probability that Rec^*'s response contains at least one non-aborting session.

For $Y \subsetneq X \subseteq \Sigma$, define:

$$q_{X|Y} = \Pr_{S \leftarrow_R \mathcal{D}_{\mathsf{dummy}}}[S \not\subseteq X \mid S \not\subseteq Y] \tag{4.2}$$

Remark 1. For any $Y \subsetneq X$, it holds that $q_X = q_{X|Y} \cdot q_Y$, and so $q_X = q_{X|\varnothing} \cdot q_\varnothing$.

An abstraction of the simulator. The simulator basically solves the following problem: we are given black-box access to a distribution $\mathcal{D}$ over subsets of $[k]$. Each time we obtain a sample $S \leftarrow_R \mathcal{D}$, we say that we have "seen" all the elements $j \in S$. The goal is to output some $S' \subseteq [k]$ such that S' is distributed identically to $\mathcal{D}$, and each element of S' was already seen during the execution of the algorithm. (In our simulator, having seen some $j \in S$ means we have the opening for $\underline{\beta}^j$ and so can equivocate in the j'th session. We also have to do some additional work (Steps 7 and 8 in Algorithm 2) because we want to output a complete transcript, not just S.)

In the setting of this abstract problem, the strategy of our simulator is given in Algorithm 3.

The intuition why Algorithm 3 (and hence our simulator) produces a set that is distributed according to $\mathcal{D}$ is the following claim: for any $V \subsetneq U \subseteq \Sigma$, if we run

Algorithm 3 with $X = U, Y = V$ (rather than $X = Y = \varnothing$), then it outputs a random $S \leftarrow_R \mathcal{D}$ conditioned on $S \not\subseteq V$. The reasoning is as follows.

- With probability $1 - q_{U|V}$ we get a sample distributed according to $S \leftarrow_R D$ conditioned on $S \subseteq U$ and $S \not\subseteq V$, and this is our output.
- With probability $q_{U|V}$ we get $S \not\subseteq U$ and so we see some new elements, and we update X, Y. In this case we can use induction to show that, since the new value Y is U, the final output will be distributed according to $S \leftarrow_R D$ conditioned on $S \not\subseteq U$.

Combining the two, the overall distribution is correct. (This intuition is formalized later in Lemma 10.)

In the following we will analyze the simulator directly (*i.e.* with all the details pertaining to outputting a transcript and not just the set of non-aborting sessions), but it helps to keep this abstraction in mind for intuition.

4.2 Running time.

We first show that the expected running time of the simulator in Algorithm 2 is polynomial. Clearly the steps before Step 6 are efficient, so fix any choice of random coins for Rec^* and any initial commitment transcript and let Σ be the set of non-aborting sessions so far. We count the number of steps starting at Step 6 and afterwards.

We will count the number of iterations in each of the loops, and multiply this by the number of steps each iteration takes. Therefore, let $c_{\mathsf{iteration}}$ denote the maximum amount of time it takes in one iteration of any of the loops: it upper bounds the time to construct $\underline{d}^j$, send them to Rec^*, and calculate S the set of sessions where Rec^*'s responses are non-aborting and do not break binding, and compare S to Y and X, and possibly updating Y, X. It holds that $c_{\mathsf{iteration}} = \mathrm{poly}(n, k)$.

Let Σ^* denote $\cup_{S \subseteq \mathrm{supp}(\mathcal{D}_{\mathsf{dummy}})} S$. Suppose at some point in its execution, the simulator sets $X = U$ and $Y = V$ for some $V \subsetneq U \subseteq \Sigma^*$. Let $c_{U,V}$ denote the total expected number of steps the simulator takes after having set $X = U, Y = V$.

Lemma 3. *For all $V \subsetneq U \subseteq \Sigma^*$, let $v = |V|$, then it holds that $c_{U,V} \leq (k-v)((nk)^2 + 4nk)c_{\mathsf{iteration}}/q_V$.*

Proof. We prove the lemma by induction.

Base case. Consider the base case where $U = \Sigma^*$ (and $V \subsetneq \Sigma^*$ is arbitrary). The simulator repeatedly samples S until it obtains $S \not\subseteq V$. It takes $1/q_V$ executions of the loop at Step 6 on average to sample $S \not\subseteq V$. Each such execution takes $c_{\mathsf{iteration}}$ steps, so this part contributes a total of $c_{\mathsf{iteration}}/q_V$ on average.

Since $U = \Sigma^*$, therefore for any $S \not\subseteq V$ that is sampled, $S \subseteq \Sigma^*$ and so the simulator goes to Step 7. We count the number of iterations needed to calculate the timeout: a success in each trial means sampling $S' \not\subseteq V$, and so on average it takes $1/q_V$ samples to get one success, and $(nk)^2/q_V$ to get $(nk)^2$ successes. Each sample takes $c_{\mathsf{iteration}}$ steps, so overall we execute on average $(nk)^2 c_{\mathsf{iteration}}/q_V$ steps.

Next, the simulator goes to the loop at Step 8. Here it executes at most T iterations. There are two cases: either $T \leq 2nk/q_V$ or $T > 2nk/q_V$. The number of iterations

in the first case is at most $2nk/q_V$. By a standard Chernoff bound, the probability that the second case occurs is at most 2^{-nk}, and in this case we can apply the bound $T \leq nk2^{nk}$. Therefore the expected contribution of this loop is at most $2nkc_{\text{iteration}}/q_V + c_{\text{iteration}}nk \leq (4nk-1)c_{\text{iteration}}/q_V$.

Summing up, we get that $c_{\Sigma^*,V} \leq ((nk)^2+4nk)c_{\text{iteration}}/q_V \leq (k-v)((nk)^2+4nk)c_{\text{iteration}}/q_V$.

Inductive case. Suppose $U \neq \Sigma^*$. Suppose the lemma holds for all U', V' where $|U'| > |U|$.

It takes on average $1/q_V$ samples to obtain $S \not\subseteq V$. Each sample takes $c_{\text{iteration}}$ so this contributes $c_{\text{iteration}}/q_V$.

For the set $S \not\subseteq V$ that is sampled, there are two cases:

1. With conditional probability $1 - q_{U|V}$, we obtain $S \subseteq U$. Let us write $p_{U,V} = 1 - q_{U|V}$. In this case we calculate the timeout (Step 7). Calculating the timeout takes on average $1/(q_V p_{U,V})$ samples to obtain a success, and each sample requires $c_{\text{iteration}}$ steps, so overall this contributes on average $p_{U,V} \cdot (nk)^2 \cdot c_{\text{iteration}}/(q_V \cdot p_{U,V}) = (nk)^2 c_{\text{iteration}}/q_V$ steps.
   Next the simulator enters the loop at Step 8. This loop runs at most T times. As with the base case, there are two cases: either $T \leq 2nk/(q_V p_{U,V})$ or $T > 2nk/(q_V p_{U,V})$. As before, we may argue that in the first case T contributes at most $2nk/(q_V p_{U,V})$ and the expected contribution of the second case is at most nk, so overall the contribution is $p_{U,V} \cdot c_{\text{iteration}}(2nk/(q_V p_{U,V}) + nk) \leq (4nk-1)c_{\text{iteration}}/q_V$.
2. The other case is when $S \not\subseteq U$. Such an S is sampled with conditional probability $q_{U|V}$. In this case we update the variables so that $X = U \cup S$ and $Y = U$, as well as updating the values of the $\underline{\beta}^j$. From this point on, the remaining number of steps spent in the loop is given by $c_{U\cup S,U}$. Since $S \not\subseteq U$, therefore $|U \cup S| > |U|$ and $|U| \geq v+1$, and we can apply the inductive hypothesis. That is, for any such S, the inductive hypothesis states that

   $$c_{U\cup S,U} \leq (k-v-1)((nk)^2+4nk)c_{\text{iteration}}/q_U$$

   Therefore, by applying Remark 1, this contributes $q_{U|V} \cdot (k-v-1)((nk)^2+4nk)c_{\text{iteration}}/q_U = (k-v-1)((nk)^2+4nk)c_{\text{iteration}}/q_V$.

Taking the sum of all the terms we have $c_{U,V} \leq (k-v)((nk)^2+4nk)/q_V$. ■

Finally, we observe that the expected running time C of the simulator is bounded by:

$$\begin{aligned} C &= \text{poly}(n,k) + q_\varnothing \cdot \mathbb{E}_S[c_{S,\varnothing} \mid S \neq \varnothing] \\ &\leq \text{poly}(n,k) + q_\varnothing \cdot \mathbb{E}_S[k((nk)^2+4nk)c_{\text{iteration}}/q_\varnothing \mid S \neq \varnothing] \\ &\leq \text{poly}(n,k) \end{aligned}$$

The first $\text{poly}(n,k)$ comes from the steps before Step 6 and the contribution from when the very first iteration of the first loop samples $S = \varnothing$. The second term is the contribution from when $S \neq \varnothing$. ■

4.3 Indistinguishability

Next we prove that the output of the simulator is computationally indistinguishable from the honest interaction. To do this we use a sequence of hybrid simulators, which unlike the simulator know the input $\underline{b}$ during the entire simulation.

$\mathsf{HSim}(\underline{b})^{\mathsf{Rec}^*}$ which is identical to Sim except it knows the input $\underline{b}$ beforehand and it has the following modifications:

1. In Step 6a, Step 7, and Step 8a, to construct $\underline{d}^j$ for $j \in \Sigma$ do the following: for each $i \in [n]$, sample $\eta_i^j \leftarrow_R \{0,1\}$ and construct d_i^j to be a commitment to $M(b_j, \eta_i^j)$ (recall that M was defined in Algorithm 1).
2. In Step 8(b)iiB, for each $j \in \Sigma$, open the row in d_i^j that equals $(b_j, \beta_i^j \oplus b_j)$.

Namely, it constructs all the commitments honestly, which it can do because it knows $\underline{b}$. Observe that the simulator can still successfully open its final commitments because they are generated honestly (without relying on learning the $\underline{\beta}^j$ sent in the initial commitment by Rec^*).

We define a second hybrid BSim that is identical to HSim except it does not check whether or not the openings given by Rec^* are consistent (*i.e.* whether binding is ever broken). However BSim still calculates and enforces the timeout.

We define a third hybrid TSim that is identical to BSim except it does not check the timeout condition. (Namely, TSim is like HSim except it enforces neither the timeout nor the binding broken conditions.)

Lemma 4. *$\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$ and $\mathsf{BSim}^{\mathsf{Rec}^*}(\underline{b})$ both run in expected polynomial time.*

Proof. The proof of the expected polynomial running time of Sim applies to each of these simulators as well: it only used the fact that with high probability the timeout calculation is accurate, and then afterwards bounds the running time by using the timeout.

Namely, one can apply the entire proof with the sole modification being the definition of the $q_X, q_{X|Y}$ (Equation 4.1, Equation 4.2), which, instead of using $\mathcal{D}_{\mathsf{dummy}}$, are now defined with respect to the following distribution $\mathcal{D}_{\underline{b}}$:

Definition 4. *Fix a transcript of the initial commitment. Let $S \leftarrow_R \mathcal{D}_{\underline{b}}$ be defined as follows: construct $\underline{d}^j$ commitments to b_j for $j \in \Sigma$ as an honest sender would and send them to Rec^*. Let S be the sessions in Rec^*'s response that are non-aborting.*

Since the actual steps in each iteration of the loops at Step 6a, Step 7, and Step 8 (which are the only differences between Sim and HSim) never really entered into the proof, one can apply the rest of the proof for Sim to HSim.

Since the proof never used the fact that Sim sometimes outputs "binding broken", and since outputting "binding broken" can only reduce the running time, this same argument also extends to BSim. ∎

Let $(\mathsf{Sim}^{\mathsf{Rec}^*} \mid \underline{b})$ denote the distribution of the output of the simulator, where the "conditioned on $\underline{b}$" notation emphasizes the fact that the simulator does not see $\underline{b}$ until it requests some subset I to be opened, and even then it only sees $\underline{b}_I$. The following four lemmas show that, by using these hybrids, it holds that $(\mathsf{Sim}^{\mathsf{Rec}^*} \mid \underline{b})$ and $\langle \underline{\mathsf{Send}}, \mathsf{Rec}^* \rangle(\underline{b})$ are computationally indistinguishable.

Lemma 5. *For all sufficiently large n, k and all $\underline{b} \in \{0,1\}^k$, the two distributions $(\mathsf{Sim}^{\mathsf{Rec}^*} \mid \underline{b})$ and $\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$ are computationally indistinguishable.*

Lemma 6. *For all sufficiently large n, k and all $\underline{b} \in \{0,1\}^k$, the two distributions $\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$ and $\mathsf{BSim}^{\mathsf{Rec}^*}(\underline{b})$ have negligible statistical distance.*

Lemma 7. *For all n, k and all $\underline{b} \in \{0,1\}^k$, the two distributions $\mathsf{BSim}^{\mathsf{Rec}^*}(\underline{b})$ and $\mathsf{TSim}^{\mathsf{Rec}^*}(\underline{b})$ have negligible statistical distance.*

Lemma 8. *For all n, k and all $\underline{b} \in \{0,1\}^k$, the two distributions $\mathsf{TSim}^{\mathsf{Rec}^*}(\underline{b})$ and $\langle \underline{\mathsf{Send}}, \mathsf{Rec}^* \rangle(\underline{b})$ are identical.*

We now turn to proving these lemmas.

Proof (of Lemma 5, Sim and HSim are computationally indistinguishable.). Suppose there exists an efficient distinguisher D, a polynomial $P(n)$ and infinitely many $n, k = \mathrm{poly}(n), \underline{b} \in \{0,1\}^k$ such that D distinguishes $(\mathsf{Sim}^{\mathsf{Rec}^*} \mid \underline{b})$ from $\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$ with advantage $1/P(n)$. We build a distinguisher that breaks hiding for $(\mathsf{Send}_{\mathsf{NI}}, \mathsf{Rec}_{\mathsf{NI}})$.

Let C denote the maximum of the expected running times of $\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$ and $(\mathsf{Sim}^{\mathsf{Rec}^*} \mid \underline{b})$ and the running time of the distinguisher D. Construct the following algorithm E, which is supposed to distinguish oracle $\mathcal{O}_1$ from $\mathcal{O}_2$ taking input $b \in \{0,1\}, \beta \in \{0,1\}$ and behaving as follows:

1. $\mathcal{O}_1(b, \beta)$ outputs a commitment using $\mathsf{Send}_{\mathsf{NI}}$ to $(b, \beta \oplus \overline{b})$.
2. $\mathcal{O}_2(b, \beta)$ outputs a commitment using $\mathsf{Send}_{\mathsf{NI}}$ to $(\overline{b}, \beta \oplus \overline{b})$.

As advice E receives an input $(n, k, \underline{b})$ where D achieves advantage $1/P(n)$.

E executes $\mathsf{Sim}^{\mathsf{Rec}^*}$ (*i.e.* Algorithm 2) except for the following modifications. For each $j \in X$,

1. In Step 6a and Step 7, for each $j \in \Sigma, i \in [n]$, construct d_i^j as follows: E samples $\nu_i^j \leftarrow_{\mathrm{R}} \{0,1\}$ and calculates by itself commitments under $\mathsf{Send}_{\mathsf{NI}}$ to the bits $(b_j, \nu_i^j \oplus b_j)$, call these $d_{i,0}^j$. It calls $\mathcal{O}(b_j, \nu_i^j)$ to get a commitment to two more bits, call these $d_{i,1}^j$. E creates d_i^j by setting with probability $1/2$ the commitments $d_{i,0}^j$ as the top row and $d_{i,1}^j$ as the bottom row, and with probability $1/2$ the other way around.
2. In Step 8a, for each $j \in \Sigma, i \in [n]$, generate d_i^j as follows: E calculates by itself commitments under $\mathsf{Send}_{\mathsf{NI}}$ to the bits $(b_j, \beta_i^j \oplus b_j)$, call these $d_{i,0}^j$. It calls $\mathcal{O}(b_j, \beta_i^j)$ to get a commitment to two more bits, call these $d_{i,1}^j$. E creates d_i^j by setting with probability $1/2$ the commitments $d_{i,0}^j$ as the top row and $d_{i,1}^j$ as the bottom row, and with probability $1/2$ the other way around.
3. In Step 8(b)iiB, opens the row in d_i^j where it inserted $d_{i,0}^j$.

Finally, E applies the distinguisher D to the output transcript and outputs the same thing as D. Let $E^{\mathcal{O}}(\underline{b})$ denote E run with oracle $\mathcal{O}$ and input $\underline{b}$.

Claim. $\Pr[E^{\mathcal{O}_2}(\underline{b}) = 1] = \Pr[D(\mathsf{Sim}^{\mathsf{Rec}^*} \mid \underline{b}) = 1]$.

Proof. The only place where E differs from Sim is in how it constructs $\underline{d}^j$.

Let us look at Step 6a, the case of the other steps is identical (for Step 8a, replace ν_i^j by β_i^j). For each $j \in X, i \in [n]$, observe that d_i^j constructed according to E using $\mathcal{O}_2$ gives a commitment to a matrix where one randomly chosen row equals $(b_j, \nu_i^j \oplus b_j)$ and the other row equals $(\overline{b_j}, \nu_i^j \oplus \overline{b_j})$. This is the same as a commitment to $F(\gamma, \nu_i^j)$ for $\gamma \leftarrow_R \{0,1\}$, which is how Sim constructs d_i^j. Since the openings to the non-interactive commitments are deterministic given fixed d_i^j, this means that the distribution of output of $E^{\mathcal{O}_2}(\underline{b})$ is identical to the distribution of $(\mathsf{Sim}^{\mathsf{Rec}^*} \mid \underline{b})$. ■

Claim. $\Pr[E^{\mathcal{O}_1}(\underline{b}) = 1] = \Pr[D(\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})) = 1]$

Proof. Again it suffices to look only at the loop at Step 6a. For each $j \in X, i \in [n]$, observe that d_i^j constructed according to E using $\mathcal{O}_1$ gives a commitment to a matrix where one randomly chosen row equals $(b_j, \nu_i^j \oplus b_j)$ and the other row equals $(b_j, \nu_i^j \oplus \overline{b_j})$. This is the same as a commitment to $M(b_j, \eta)$ for $\eta \leftarrow_R \{0,1\}$, which is how HSim constructs d_i^j. Since the opening to the non-interactive commitments are deterministic given fixed d_i^j, this means that the distribution of output of $E^{\mathcal{O}_1}(\underline{b})$ is identical to the distribution of $\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$. ■

These two claims imply that $E^{(\cdot)}(\underline{b})$ distinguishes between $\mathcal{O}_1$ and $\mathcal{O}_2$ with advantage $1/P(n)$. Furthermore, the expected running time of E is bounded by $2C$. Let us truncate its running time to $6P(n)C$, then the distinguishing advantage remains at least $1/(3P(n))$. Furthermore, the fact that HSim and Sim are expected polynomial time means that $6P(n)C$ is polynomial. By a standard hybrid argument, this can be transformed into an efficient distinguisher for a single call to $\mathcal{O}_1$ vs $\mathcal{O}_2$. By another standard argument, this can be transformed into an efficient distinguisher breaking the hiding property of the commitment. ■

Proof (of Lemma 6, HSim and BSim are statistically close.). By definition, HSim and BSim are identical except in the case that HSim outputs "binding broken". This can only happen with negligible probability: otherwise using a standard argument, *e.g.* given in Goldreich-Kahan, if C bounds the expected running time of $\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$ and $\mathsf{HSim}^{\mathsf{Rec}^*}(\underline{b})$ outputs "binding broken" with non-negligible $1/P(n)$, then by truncating the execution of HSim at $2P(n)C$ we get an algorithm that outputs "binding broken" with non-negligible probability $\frac{1}{2P(n)}$. By Lemma 4 $C = \mathrm{poly}(n,k)$ and so this algorithm is efficient. This can then be used to break the binding of the commitment used by the receiver, which contradicts the computational binding property of the commitment. ■

Proof (of Lemma 7, BSim and TSim are statistically close.).

By definition, BSim and TSim are identical except in the case that BSim times out. We calculate this probability. For the following, fix any choice of $\underline{b}$, Rec^*'s random coins, and initial commitment from Rec^*.

Let $B_{U,V}$ denote the event that BSim breaks from the first loop with $X = U, Y = V$. Since a timeout can only occur when $B_{U,V}$ occurs with $U \neq \varnothing$, we observe that:

$$\Pr_{\mathsf{BSim}}[\mathsf{BSim} \text{ times out}] = \sum_{V \subsetneq U \subseteq \Sigma} \Pr_{\mathsf{BSim}}[\mathsf{BSim} \text{ times out} \wedge B_{U,V}] \tag{4.3}$$

Since there are less than 2^{2k} choices for U, V, it suffices to show that each term of the summation is bounded by $2^{-\Omega(nk)}$. To do this, we relate $\Pr[B_{U,V}]$ to the following quantity:

$$\delta_{U,V} = \Pr_{S \xleftarrow{R} \mathcal{D}_{\underline{b}}}[S \subseteq U \wedge S \not\subseteq V] \tag{4.4}$$

where $\mathcal{D}_{\underline{b}}$ is as defined in Definition 4. We claim that

Lemma 9. $\Pr_{\mathsf{BSim}}[B_{U,V}] \leq \delta_{U,V}$

Let us apply this lemma to complete the proof of the lemma; we will prove the lemma later. By the lemma, all terms in Equation 4.3 satisfy either $\Pr[B_{U,V}] \leq 2^{-nk}$ or $\delta_{U,V} > 2^{-nk}$. The second case is the only interesting one, so fix such U, V. It suffices to show that $\Pr[\mathsf{BSim} \text{ times out} \mid B_{U,V}] \leq 2^{-\Omega(nk)}$.

Let T denote the timeout calculated in the simulation. Since each trial in the timeout calculation is a success with probability $\delta_{U,V}$, the expected number of trials necessary to obtain $(nk)^2$ successes is $\frac{(nk)^2}{\delta_{U,V}}$. Therefore by a standard Chernoff bound, the probability that $T < \frac{nk}{2\delta_{U,V}}$ is at most 2^{-nk}. (Here the assumption that $\delta_{U,V} > 2^{-nk}$ is important, since by definition T is limited to be at most $nk2^{nk}$.)

Conditioned on $T \geq \frac{nk}{2\delta_{U,V}}$, the probability of timeout is at most $(1-\delta_{U,V})^T \leq 2^{-nk/2}$. In total therefore $\Pr[\mathsf{BSim} \text{ times out} \mid B_{U,V}] \leq 2^{-nk} + 2^{-nk/2} < 2^{-nk/3}$. Therefore, every term in Equation 4.3 is bounded by $2^{-\Omega(nk)}$ and since there are less than 2^{2k} terms in total, the total probability of timeout is negligible. ■

We now prove Lemma 9.

Proof (of Lemma 9). Let $\overline{\alpha}$ denote a vector of $\mathsf{Send}_{\mathsf{NI}}$ commitments $(\underline{d}^j)_{j \in \Sigma}$. Let z denote a pair containing a vector of queries $\overline{\alpha}$ and a response s from Rec^*. For a fixing of $z = (\overline{\alpha}, s)$, let Z denote the set of non-aborting sessions in s.

For any z, let A_z denote the event that the simulator breaks from the first loop where, in the iteration that causes the loop to break, the query to Rec^* are the queries in z and the response received is the response in z. By definition, it holds that

$$\Pr_{\mathsf{BSim}}[B_{U,V}] \leq \sum_{z \mid Z \not\subseteq V, Z \subseteq U} \Pr_{\mathsf{BSim}}[A_z] \tag{4.5}$$

The following says that $\Pr_{\mathsf{BSim}}[A_z] = \Pr[(\overline{\alpha}, s) = z]$, where $\overline{\alpha}$ are constructed as honest commitments to $\underline{b}$ and s is Rec^*'s response (*i.e.* the same probability space as $\mathcal{D}_{\underline{b}}$). This implies that the RHS of Equation 4.5 is equal to $\delta_{U,V}$ and Lemma 9 follows.

Claim. For all z where $Z \neq \varnothing$, $\Pr_{\mathsf{BSim}}[A_z] = \Pr[(\overline{\alpha}, s) = z]$.

Proof. Let $\Sigma^* = \bigcup_{S \subseteq \mathrm{supp}(\mathcal{D}_{\underline{b}})} S$. If $Z \not\subseteq \Sigma^*$ then $\Pr[A_z] = \Pr[(\overline{\alpha}, s) = z] = 0$ and we are done, so suppose that $Z \subseteq \Sigma^*$. For any $V \subsetneq U \subseteq \Sigma^*$, let $\rho_{U,V,z}$ denote the probability A_z occurs, conditioned on BSim ever executing the first loop (Step 6) with $X = U, Y = V$ (but not necessarily breaking from the first loop with $X = U, Y = V$). We prove that

$$\rho_{U,V,z} = \Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq V] \tag{4.6}$$

where S is the set of non-aborting sessions of the response contained in s. This would imply the claim, since for any z with non-empty Z, we have

$$\Pr_{\mathsf{BSim}}[A_z] = \Pr[S \neq \varnothing] \cdot \mathbb{E}_S[\rho_{S,\varnothing,z} \mid S \neq \varnothing] = \Pr[(\overline{\alpha}, s) = z]$$

since it must be that the first iteration of the loop sampled $S \neq \varnothing$ and then conditioned on this the probability of sampling z is given by $\rho_{S,\varnothing,z}$. (This corresponds to our earlier intuition that the abstract sampling algorithm of Algorithm 3 samples the correct distribution.)

We prove Equation 4.6. If $Z \subseteq V$ then both sides of Equation 4.6 are 0. So suppose that $Z \not\subseteq V$. There are two cases:

1. Suppose that $Z \subseteq U$. Let us look at the very first sample $(\overline{\alpha}, s)$ satisfying $S \not\subseteq V$ that is obtained after the simulator sets $X = U, Y = V$. If $(\overline{\alpha}, s) \neq z$ then either BSim breaks the loop with this different query/response, or else Y is updated to be $U \cup S$ and in the subsequent iterations of the loop, Z is contained in the updated Y, and so z can no longer possibly be sampled.
   Therefore, the only contribution to the probability of z being sampled is when this first sample $(\overline{\alpha}, s) = z$. This occurs with probability $\Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq V]$.
   In particular, this shows that Equation 4.6 holds for any $V \subsetneq \Sigma^*$ when $U = \Sigma^*$.
2. Suppose that $Z \not\subseteq U$. Since the first point establishes Equation 4.6 when $U = \Sigma^*$, we may use induction and assume that it holds for all $\rho_{U',U,z}$ where $|U'| > |U|$. Since $Z \not\subseteq U$, it follows that A_z only occurs if BSim does not break the loop while $X = U$. Therefore, $\rho_{U,V,z}$ equals:

$$\begin{aligned}
&\sum_{W \not\subseteq U} \Pr[S = W \mid S \not\subseteq V] \cdot \rho_{U \cup W, U, z} \\
&\qquad = \Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq U] \sum_{W \not\subseteq U} \Pr[S = W \mid S \not\subseteq V] \\
&\qquad = \Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq U] \Pr[S \not\subseteq U \mid S \not\subseteq V] \\
&\qquad = \Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq V]
\end{aligned}$$

   where in the last step we use the fact that $Z \not\subseteq U$ and that for events A, B, C such that A implies B implies C, it holds that $\Pr[A \mid B] \Pr[B \mid C] = \Pr[A \mid C]$. ■

Proof (of Lemma 8, TSim *and* $\langle \underline{\mathsf{Send}}, \mathsf{Rec}^* \rangle$ *are identical.).*

By definition, the output of TSim and an honest interaction are identical up to the end of the initial commitment, so fix any random coins of Rec^* and fix any initial commitment transcript. As in Lemma 9, let z be any tuple of queries to and responses of Rec^* to open its initial commitments, *i.e.* z is of the form $(\overline{\alpha}, s)$. Let Z denote the set of non-aborting sessions in s.

If $Z = \varnothing$, then it is clear that the probability that TSim outputs z is identical to the probability of z being output in an honest transcript, since this can only be output in

the first iteration of the first loop in TSim and by definition this is identical to an honest interaction.

So consider z such that $Z \neq \varnothing$. As in the proof of Lemma 9, let A_z denote the probability that the TSim breaks from the first loop and the last query to and response received from Rec^* before breaking being given by the tuple z. TSim and BSim are completely identical in the first loop, so we can apply Equation 4.3 to show that $\Pr_{\mathsf{TSim}}[A_z] = \Pr[(\overline{\alpha}, s) = z]$.

Let A'_z denote the event that TSim outputs z as the query/response in the step corresponding to Step 8(b)ii.

Lemma 10. *For all z containing at least one non-aborting session, it holds that* $\Pr_{\mathsf{TSim}}[A'_z] = \Pr_{\mathsf{TSim}}[A_z]$.

This combined with Equation 4.3 imply that $\Pr_{\mathsf{TSim}}[A'_z] = \Pr[(\overline{\alpha}, s) = z]$.

If A'_z occurs then z is written to the output. From the definition of the TSim, conditioned on outputting z, the rest of the output, namely the choice of I and opening, are identical to the honest interaction conditioned on outputting z. Since the probability of outputting z is identical, this proves that $\mathsf{TSim}^{\mathsf{Rec}^*}(\underline{b})$ and $\langle \underline{\mathsf{Send}}, \mathsf{Rec}^* \rangle(\underline{b})$ are identical.

It remains to prove Lemma 10.

Proof (of Lemma 10). If A_z occurs, then it must be that TSim breaks out of the first loop for some U, V satisfying $Z \not\subseteq V$ and $Z \subseteq U$. This is exactly the event $B_{U,V}$ as defined in Lemma 9. Likewise, if A'_z occurs then $B_{U,V}$ must occur for some $Z \not\subseteq V, Z \subseteq U$. Therefore it suffices to show that for all U, V such that $Z \not\subseteq V, Z \subseteq U$ and $\Pr[B_{U,V}] > 0$, it holds that

$$\Pr_{\mathsf{TSim}}[A'_z \mid B_{U,V}] = \Pr_{\mathsf{TSim}}[A_z \mid B_{U,V}] \tag{4.7}$$

since we could apply this as follows to deduce Lemma 10:

$$\Pr[A_z] = \sum_{U,V \mid Z \not\subseteq V, Z \subseteq U} \Pr[A_z \wedge B_{U,V}] = \sum_{U,V \mid Z \not\subseteq V, Z \subseteq U} \Pr[A'_z \wedge B_{U,V}] = \Pr[A'_z]$$

We now prove Equation 4.7. The LHS is equal to $\Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq V, S \subseteq U]$ because by definition of TSim, it generates $\overline{\alpha}$ as honest commitments to $\underline{b}$ and gets a response s satisfying $S \not\subseteq V$ and $S \subseteq U$ (notice this requires the fact that there is no timeout or binding broken condition).

To evaluate the RHS, let $E_{U,V}$ denote the event of TSim ever executing the first loop with $X = U, Y = V$. By definition if $B_{U,V}$ occurs then so does $E_{U,V}$. Therefore we may develop:

$$\begin{aligned} \Pr_{\mathsf{TSim}}[A_z \mid B_{U,V}] &= \frac{\Pr_{\mathsf{TSim}}[A_z \wedge B_{U,V}]}{\Pr_{\mathsf{TSim}}[B_{U,V}]} \\ &= \frac{\Pr_{\mathsf{TSim}}[A_z \wedge B_{U,V} \mid E_{U,V}]}{\Pr_{\mathsf{TSim}}[B_{U,V} \mid E_{U,V}]} \end{aligned}$$

In the last line, we can simplify the numerator to $\Pr[A_z \mid E_{U,V}]$, because conditioned on $E_{U,V}$, A_z implies $B_{U,V}$. Since $Z \subseteq U$, it also holds that $\Pr[A_z \mid E_{U,V}] = \Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq V]$. The denominator in the last line equals $\Pr[S \subseteq U \mid S \not\subseteq V]$.

Therefore we have that

$$\Pr_{\mathsf{TSim}}[A_z \mid B_{U,V}] = \frac{\Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq V]}{\Pr[S \subseteq U \mid S \not\subseteq V]}$$

Using the fact that $(\overline{\alpha}, s) = z$ implies that $S = Z \subseteq U$, we can simplify the fraction to $\Pr[(\overline{\alpha}, s) = z \mid S \not\subseteq V, S \subseteq U]$. This proves Equation 4.7 for all U, V satisfying $Z \not\subseteq V$, $Z \subseteq U$. ■

5 Conclusion

Combined with [ORSV11, Xia11], we now have a fairly comprehensive view of commitments with selective opening attack security (under parallel composition): for statistically hiding commitments there exist 3-round protocols and these are optimal for black-box simulation [ORSV11,Xia11], and for statistically-binding commitments there exist 4-round protocols and these are optimal for black-box simulation [ORSV11,Xia11]. Interestingly, the situation is the reverse of stand-alone commitments, where we know non-interactive statistically-binding commitments yet the minimal complexity of statistically hiding commitments is two rounds (without setup assumptions).

[ORSV11] showed that their statistically-hiding commitment is not only secure under parallel composition but also under "concurrent-with-barrier" composition: the commit-phase may occur with arbitrary scheduling of the messages, but the reveal phase happens at the same time across all sessions. An interesting open question is to show whether this is possible for statistically-binding commitments.

References

Bea96. D. Beaver. Adaptive zero knowledge and computational equivocation (extended abstract). In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 629–638, New York, NY, USA, 1996. ACM.

BHY09. M. Bellare, D. Hofheinz, and S. Yilek. Possibility and Impossibility Results for Encryption and Commitment Secure under Selective Opening. In A. Joux, editor, *EUROCRYPT*, volume 5479 of *Lecture Notes in Computer Science*, pages 1–35. Springer, 2009.

BCC88. G. Brassard, D. Chaum, and C. Crépeau. Minimum Disclosure Proofs of Knowledge. *J. of Comp. and Sys. Sci.*, 37(2):156–189, Oct. 1988.

BCY89. G. Brassard, C. Crépeau, and M. Yung. Everything in NP Can Be Argued in *Perfect* Zero-Knowledge in a *Bounded* Number of Rounds. In *Eurocrypt '89*, pages 192–195, 1989. LNCS No. 434.

DPP93. I. Damgård, T. P. Pedersen, and B. Pfitzmann. On the Existence of Statistically Hiding Bit Commitment Schemes and Fail-Stop Signatures. In *In CRYPTO '93*, pages 250–265, 1993.

DPP98. I. Damgård, T. P. Pedersen, and B. Pfitzmann. Statistical Secrecy and Multibit Commitments. *IEEE Transactions on Information Theory*, 44(3):1143–1151, 1998.

DCIO98. G. Di Crescenzo, Y. Ishai, and R. Ostrovsky. Non-interactive and non-malleable commitment. In *STOC '98: Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 141–150, New York, NY, USA, 1998. ACM.

DCO99. G. Di Crescenzo and R. Ostrovsky. On Concurrent Zero-Knowledge with Preprocessing. In M. J. Wiener, editor, *CRYPTO*, volume 1666 of *Lecture Notes in Computer Science*, pages 485–502. Springer, 1999.

DNRS03. C. Dwork, M. Naor, O. Reingold, and L. Stockmeyer. Magic Functions: In Memoriam: Bernard M. Dwork 1923–1998. *J. ACM*, 50(6):852–921, 2003.

Fis01. M. Fischlin. *Trapdoor Commitment Schemes and Their Applications*. Ph.D. Thesis (Doktorarbeit), Department of Mathematics, Goethe-University, Frankfurt, Germany, 2001.

GK90. O. Goldreich and H. Krawczyk. On the Composition of Zero-Knowledge Proof Systems. *SIAM J. of Com.*, 25(1):169–192, Feb. 1996. Preliminary version appeared in ICALP' 90.

GMW86. O. Goldreich, S. Micali, and A. Wigderson. Proofs that Yield Nothing But Their Validity or All Languages in NP Have Zero-Knowledge Proof Systems. *Journal of the ACM*, 38(3):691–729, July 1991. Preliminary version in FOCS' 86.

HM96. S. Halevi and S. Micali. Practical and Provably-Secure Commitment Schemes from Collision-Free Hashing. In *In CRYPTO '96*, pages 201–215. Springer-Verlag, 1996.

ORSV11. R. Ostrovsky, V. Rao, A. Scafuro, and I. Visconti. Revisiting Lower and Upper Bounds for Selective Decommitments. Cryptology ePrint Archive, Report 2011/536, 2011. http://eprint.iacr.org/.

Xia11. D. Xiao. (Nearly) Round-Optimal Black-Box Constructions of Commitments Secure against Selective Opening Attacks. In *Proc. 8th TCC*, pages 541–558, 2011.