How to Achieve Non-Malleability in One or Two Rounds

Dakshita Khurana
UCLA
dakshita@cs.ucla.edu

Amit Sahai
UCLA
sahai@cs.ucla.edu

Abstract

Despite over 25 years of research on non-malleable commitments in the plain model, their round complexity has remained open. The goal of achieving non-malleable commitment protocols with only *one or two rounds* has been especially elusive. Pass (TCC 2013, CC 2016) captured this difficulty by proving important impossibility results regarding two-round non-malleable commitments. This led to the widespread belief that achieving two-round non-malleable commitments was impossible from standard sub-exponential assumptions. We show that this belief was false. Indeed, we obtain the following positive results:

- We construct the first two-message non-malleable commitments satisfying the strong definition of non-malleability with respect to commitment, assuming standard sub-exponential assumptions, namely: sub-exponentially hard one-way permutations, sub-exponential ZAPs, and sub-exponential DDH. Furthermore, our protocol is *public-coin*.
- We also obtain two-message *private-coin* non-malleable commitments with respect to commitment, assuming only sub-exponentially hard DDH or QR or N^{th}-residuosity.
- We bootstrap the above protocols (under the same assumptions) to obtain constant bounded-concurrent non-malleability while preserving round complexity.
- We compile the above protocols to obtain, in the simultaneous messages model, the first *one-round* non-malleable commitments, with unbounded concurrent security respect to opening, under standard sub-exponential assumptions.
  - This implies *non-interactive non-malleable commitments with respect to opening*, in a restricted model with a broadcast channel, and a-priori bounded polynomially many parties such that every party is aware of every other party in the system.
    To the best of our knowledge, this is the first protocol to achieve completely non-interactive non-malleability in *any* plain model setting from standard assumptions.
  - As an application of this result, in the simultaneous exchange model, we obtain the first two-round multi-party pseudorandom coin-flipping.

  We believe that our protocols are likely to find several additional applications.
- In order to obtain our results, we develop the first two-round black-box rewinding strategy based on standard sub-exponential assumptions, in the plain model, which may be of independent interest.

Contents

1 Introduction

The notion of non-malleability was introduced by Dolev, Dwork and Naor [DDN91] in 1991, to counter the ubiquitous problem of man-in-the-middle (MIM) attacks on cryptographic protocols. An MIM adversary participates in two or more instantiations of a protocol, trying to use information obtained in one execution to breach security in the other protocol execution. A non-malleable protocol should ensure that such an adversary gains no advantage. Let's call any interactive protocol between two parties, where both parties send at least one message to each other, a conversation. In this paper, we ask if we can provably embed non-malleability into any two-party conversation. We focus on a core non-malleable cryptographic primitive: non-malleable commitments (described below). Thus, the main question we consider in this work is,

Can we construct two-message non-malleable commitments from standard sub-exponential assumptions?

A commitment scheme is a two-party protocol between a committer and a receiver. The committer has a message m as input, while the receiver obtains no input. The two parties engage in a probabilistic interactive commitment protocol, and the receiver's view at the end of this protocol is denoted by $\mathsf{com}(m)$. Later, in the opening phase, the committer sends an opening message to the receiver, allowing the receiver to verify that the message m was really the message committed during the commitment phase.

In a (statistically) binding commitment, the receiver's view $\mathsf{com}(m)$ should be binding in the sense that with high probability, there should not exist an opening message that would convince the receiver that the committer had used any string $m' \neq m$. In short, we say that the commitment cannot be later opened to any message $m' \neq m$. A commitment should also be hiding; that is, for any pair of messages (m, m') the distributions $\mathsf{com}(m)$ and $\mathsf{com}(m')$ should be computationally indistinguishable. Finally, such a scheme is said to be *non-malleable* with respect to commitment, if for every message m, no MIM adversary, intercepting a commitment protocol $\mathsf{com}(m)$, and modifying every message sent during this protocol arbitrarily, is able to efficiently generate a commitment $\mathsf{com}(m')$ such that message m' is related to the original message m.

In the standard model, we call each message sent by any party a *round*. We will also consider the simultaneous-message model, wherein a round consists of both (or all) parties sending a single message simultaneously. Non-malleable commitments are among the core building blocks of (and therefore have a direct impact on the round complexity of) various cryptographic protocols such as coin-flipping, secure auctions, electronic voting, non-malleable proof systems and multi-party computation protocols.

The goal of achieving non-malleable commitment protocols with only *two messages* has been particularly elusive. Notably, Pass [Pas13] proved that two-message non-malleable commitments (satisfying non-malleability with respect to commitment) are impossible to construct with a black-box reduction to any polynomial falsifiable assumption. However, another claim from [Pas13] stated that two-message non-malleable commitments are impossible to construct with a black-box reduction to any *sub-exponentially* hard falsifiable assumptions, seemingly cutting off hope of achieving two-message non-malleable commitments from standard assumptions.

On the impossibility result of [Pas13]. Let us examine the impossibility result of [Pas13]: it considers the setting where there are only two identities/tags in the system, and discusses how one cannot achieve non-malleability even in this restricted setting via black-box reductions to falsifiable hardness. The impossibility builds as a counter-example, a MIM that *runs the reduction* in order to break hiding of an honest commitment and carry out a successful mauling attack. If the assumption is with regard to any polynomial-time attacker with inverse polynomial advantage, then this proof works, and the impossibility holds. It might appear that this argument should also extend to assumptions that require security against sub-exponential attackers with inverse sub-exponential advantage. However, we observe that an actual MIM only participates in at most a polynomial number of interactions and is required to break non-malleability in one of them[1], whereas a (sub-exponential) time reduction has oracle access to an adversary – and can therefore participate in sub-exponentially many interactions.

This gap between the number of sessions that the reduction can participate in, and the number of sessions in which participation is possible for any adversary that wants to "run the reduction," precludes the impossibility claim. Therefore, Theorem 5.11 as stated in [Pas16], is incorrect[2]. Indeed, we show how to contradict this statement by achieving several positive results from standard sub-exponential assumptions.

We stress that when considering a reduction that can run in sub-exponential time, a reduction that participates in sub-exponentially many sessions is no worse asymptotically than a reduction that participates in only polynomially many sessions. For example, let $\delta < \epsilon$, and suppose that we consider a reduction $\mathcal{R}$ that runs in time 2^{n^ϵ}, and participates in m sessions with an adversary MIM that runs in time 2^{n^δ}. Then observe:

- If $\mathcal{R}$ participates in $\mathsf{poly}(n)$ sessions, then the total security loss is $2^{n^\epsilon} + \mathsf{poly}(n) \cdot 2^{n^\delta} = O(2^{n^\epsilon})$.
- If $\mathcal{R}$ participates in 2^{n^δ} sessions, the security loss is $2^{n^\epsilon} + 2^{n^\delta} \cdot 2^{n^\delta} = 2^{n^\epsilon} + 2^{2n^\delta} = O(2^{n^\epsilon})$.

Thus, it makes sense asymptotically to consider reductions that can participate in sub-exponentially many sessions.

The state of the art before our work. There has been a long line of work on constructing non-malleable commitments with respect to commitment, in the plain model in as few rounds as possible (e.g.[DDN91, Bar02, PR05b, Wee10, PW10, LP, Goy11, GLOV12, GRRV14, GPR15, COSV16b, COSV16a]). In a major advance, [GPR15] showed how to construct three-message non-malleable commitments, and subsequently [COSV16b, COSV16a] obtained concurrent three-message non-malleable commitments. These results relied on super-polynomial or sub-exponential injective one-way functions to achieve general notions of non-malleability in three rounds. Thus, to the best of our knowledge, current constructions of even 3-message non-malleable commitments (with respect to commitment) require super-polynomial assumptions. In contrast, in this paper, we will construct *2-message non-malleable commitments* with respect to commitment.

In our work, we will also consider a weaker notion of malleable commitments called non-malleability with respect to opening (see below for a discussion of this definition), where our goal will be to construct *one-round non-malleable commitments* in the simultaneous-message model. Prior to our work, no one-round non-malleable commitment with respect to opening was known, for any flavor of the definition, in any communication model, without setup and based on standard assumptions. Before our work, the work of [GKS16] had the fewest rounds of interaction for non-malleable commitment with respect to opening from standard assumptions. That work showed how to construct *two-round unidirectional* non-malleable commitments achieving a form of non-malleability with respect to opening, from polynomial hardness of injective one-way functions. The

[1] Alternately, an MIM is required to maul with some inverse polynomial probability in a single interaction.

[2] We contacted Pass via personal communication, and he explicitly agreed that the impossibility result as stated in [Pas16] is incorrect. As we note above, however, the only case not ruled out by Pass is a reduction that makes super-polynomially many queries to the adversary.

model and definition in [GKS16] were carefully chosen to avoid the impossibility of [Pas13] for two rounds, even in the polynomial hardness regime. As a result, [GKS16] achieve a weaker definition of non-malleability with respect to opening than ours, achieve non-malleability only with respect to synchronizing adversaries, and require two rounds in the commit phase.

1.1 Our Results

As mentioned above, broadly speaking, there are two flavors of definitions for non-malleable commitment that have been considered in the literature, called non-malleability with respect to commitment, and non-malleability with respect to opening. We will obtain different positive results for each of these definitions.

Non-Malleable Commitments with respect to Commitment. We first consider the standard model, where each round consists of a single message from one party to another. In the standard model, we work with the stronger of the two standard definitions of non-malleability, namely non-malleability with respect to commitment (against both synchronous and asynchronous adversaries). Informally, this definition requires that non-malleability hold with respect to the underlying message as soon as the commitment phase completes. Thus, even if an adversary MIM never actually opens its commitment, nevertheless we can be assured that the message underlying his commitment did not depend on the message committed to by the honest party.

In the standard model, we obtain the following positive results from standard sub-exponential assumptions, for non-malleable commitments with respect to to commitment.

- We construct two-message *public-coin* non-malleable commitments with respect to commitment, assuming sub-exponentially hard one-way permutations, sub-exponential ZAPs, and sub-exponentially hard DDH.
- We obtain two-message *private-coin* non-malleable commitments with respect to commitment, assuming only sub-exponentially hard DDH or QR or N^{th}-residuosity.
- We bootstrap the above protocols (under the same assumptions) to obtain constant[3] bounded-concurrent non-malleability while preserving round complexity.

Another viewpoint: Non-interactive non-malleability with a tamperable CRS. If we were willing to rely on a trusted setup that generates a common random string (CRS) for all parties, constructions of non-interactive non-malleable commitments become much simpler [CIO98]. However, a major design goal of all of theoretical cryptography is to reduce global trust as much as possible. A trusted CRS is a straightforward example of the kind of global trust that we would like to avoid.

Indeed, we can interpret our result above through the lens of an *untrusted* CRS: what if the man-in-the-middle attacker can arbitrarily tamper with a CRS, and convince an honest committer to generate his commitment with respect to this tampered CRS? For all prior constructions, in this situation, all bets would be off. On the other hand, our work shows the first solution to this problem: we obtain non-interactive non-malleable commitment with respect to commitment, where the honest committer must use a *tampered CRS*.

[3] Our actual construction imposes a trade-off between the concurrent non-malleability and the tag space. Please see Section 6.4 for a discussion of this tradeoff, and the actual bounds that we have in different settings.

Non-Malleable Commitments with respect to Opening. We next consider the simultaneous-message model, where a round consists of both (or all) parties sending a single message to all other parties. We consider the standard asynchronous model with rushing adversaries.

We achieve the first one-round non-malleable commitment protocols in this model under standard sub-exponential assumptions. To achieve one-round protocols, we work with the other definition of non-malleable commitments, called non-malleability with respect to opening. Roughly speaking, this definition requires that the adversary cannot *open* his commitment to a value related to the honest party's opened value. There are several ways to formulate the definition of non-malleability with respect to opening. We formulate a simulation-based definition that is both simpler and more powerful than the recent indistinguishability-based definition of [GKS16] (in particular, our definition implies the definition of [GKS16], see Section 4.2 for more details). Furthermore, we require and obtain security against asynchronous adversaries, whereas the work of [GKS16] required an additional round and only obtained non-malleable commitments with respect to opening against synchronous adversaries.

In particular, in the simultaneous-message model, we obtain the following results from standard sub-exponential assumptions:

- We compile the previously described two-round protocols in the standard model to obtain *one-round* non-malleable commitments with respect to opening, in the simultaneous-message model. The opening phase of this protocol remains non-interactive.
- We further show how to transform this protocol to achieve fully concurrent non-malleable commitments with respect to opening, in the simultaneous-message exchange model, still using only one round. The opening phase of this transformed protocol remains non-interactive.
- We show that this implies concurrent *completely non-interactive non-malleable commitments with respect to opening*, in a model with a broadcast channel, and an a-priori fixed polynomial number of parties such that every party is aware of every other party in the system. To the best of our knowledge, this is the first protocol to achieve completely non-interactive non-malleability in *any* plain model setting from standard assumptions.

Applicability. The general applicability of non-malleable commitments within cryptography is well known; a classic and simple example is conducting sealed-bid auctions online. As mentioned above, in a setting where there are a fixed polynomial number of participants and a broadcast channel, our results give the first completely non-interactive method of conducting sealed-bid auctions based on standard sub-exponential assumptions.

Can we break round-complexity barriers in other settings as well? Indeed, consider the classic question of secure coin flipping [Blu81] in a multi-party setting, where parties wish to agree on a shared random string. Note that the standard model of interaction in this setting is the simultaneous-message model. The work of [GMPP16] establish a lower bound of 4 rounds for secure multi-party coin-flipping with black-box security from polynomial hardness assumptions (with polynomial simulation). We show that by moving to the sub-exponential regime (with sub-exponential simulation), we can cut this lower bound in half! We give the first two-round bounded multi-party secure coin flipping protocol (with sub-exponential simulation) from standard sub-exponential assumptions. Note that sub-exponential simulation also implies two-round pseudorandom coin-flipping, where the output of the coin flipping protocol is indistinguishable from random even to sub-exponential time distinguishers.

1.2 Related Work

In less than three messages, the only prior method of achieving 2-message non-malleable commitments with respect to commitment was via the assumption of adaptive one-way functions [PPV08], which essentially assumes the existence of a one-way function that already exhibits strong non-malleability properties. Such assumptions are very different in spirit from traditional hardness assumptions, and are both non-falsifiable [Nao03] and not complexity assumptions in the sense of [GK90].

We note that constructions of non-malleable commitments in two rounds were not known even based on indistinguishability obfuscation. Recently [KS17] showed how to obtain two-message non-malleable commitments with respect to commitment from an injective one-way function with very strong exponential hardness beyond $2^{n/2}$, while we obtain the same result from standard sub-exponential assumptions.

1.3 Comparison with Concurrent Independent Work of [LPS17]

In a fascinating concurrent and independent work, Lin, Pass, and Soni (LPS) [LPS17] construct two-message concurrent non-malleable commitments, and non-interactive non-malleable commitments with respect to commitment against uniform adversaries. Their work is substantially different from ours in terms of techniques as well as assumptions.

The constructions of LPS require several assumptions, most notably a novel sub-exponential variant of the Rivest-Shamir-Wagner (RSW) assumption first proposed for constructing time-lock puzzles by [RSW96]. Roughly speaking, the RSW assumption considers the Repeated Squaring Algorithm for computing $h = g^{2^n}$, and requires that the natural algorithm for computing h in time n cannot be sped up by parallel computation. The novel variant of the RSW assumption considered by [LPS17] is more complex than the original RSW assumption in that it is essentially a "two-dimensional" family of assumptions: In the new assumption, there is a security parameter n and another parameter t, and it is required that computing $h = g^{2^{2^t}}$ cannot be done by circuits of overall size 2^{n^ϵ} and depth 2^{t^δ}, for constants ϵ and δ.

For example, their assumption implies the following two specific assumptions (informally stated) as special cases.

(RSW Variant A): There exist constants ϵ, δ, and c, such that the value $g^{2^{n^c}}$ cannot be computed by circuits of size 2^{n^ϵ} and depth $\mathsf{polylog}(n)$. Note that $g^{2^{n^c}}$ can be computed in roughly time n^c.

(RSW Variant B): There exist constants ϵ and δ such that the value $g^{2^{2^n}}$ cannot be computed by circuits of size 2^{n^ϵ} and depth 2^{n^δ}.

Thus, the new assumption of [LPS17] essentially assumes that (a large family of) computations cannot be sped up via parallelism.

In contrast, standard subexponential assumptions in cryptography – including the assumptions that we make in our work – require only security against circuits of subexponential size, regardless of the depth of these circuits. In this way, the assumption of [LPS17] is non-standard, and falls outside the definition of falsifiable assumptions ruled out by Pass [Pas13]. Indeed, the authors [LPS17] themselves note that assumptions of this type were previously used only in time-release cryptography. On the other hand, the assumptions that we use in our work have been considered by many previous works constructing cryptographic protocols, including secure computation protocols.

It is also noted in [LPS17] that their 2-round protocols can be based entirely on search assumptions (note that their non-interactive protocols require additional nonstandard assumptions). However, in this case, [LPS17] also require subexponential trapdoor permutations (for building ZAPs) in addition to their novel variant of the RSW assumption.

Finally, on a quantitative level, we only require $O(\log^* n)$ levels of complexity leveraging, thereby only requiring sub-subexponential hardness assumptions as per the new definition of [LPS17].

In terms of techniques, the novel assumption on parallel complexity allows LPS to[4] construct a pair of commitment schemes Com_1 and Com_2 that are simultaneously harder than the other, in different axes. In particular, Com_2 is harder in the axis of circuit-size, in the sense that Com_1 admits an extractor of size S while Com_2 is secure against all circuits of size S; on the other hand, Com_1 is harder in the axis of circuit-depth, in the sense that it admits an extractor of depth D (and some size S) while Com_1 is hiding against all circuits with depth D (and size S). This scheme already achieves a flavor of non-malleability for two tags.

In contrast, we develop new techniques to work with a single axis of hardness, in order to rely on standard subexponential assumptions. Indeed, a lot of work in our paper goes into constructing extractable commitments that help us obtain a non-malleable commitment scheme for just two tags (please refer to Section 2 for more details).

2 Overview of Techniques

As we already discussed, we would like to build protocol that admits a security reduction that can access the (adversarial) committer a super-polynomial number of times, while an actual adversary can only interact with the honest committer in polynomially many executions. Any hope of obtaining a positive result requires us to exploit this disparity between the MIM and the reduction, otherwise our approach would succumb to the impossibility result of [Pas16].

Main Tool: Extractable Commitments. The crux of this question boils down to building a special kind of extractable commitment with just two messages. In such a commitment scheme, informally speaking, there is a black-box extractor algorithm that runs in time T', that extracts the values committed to by any malicious polynomial-time committer. Popular intuition so far has been that rewinding with only two rounds is useless: whatever the extractor can do, a malicious receiver can also do.

However, in our new kind of extractable commitment, we will require that the hiding property of the commitment scheme holds with respect to any malicious receiver that runs in time T that exceeds T'. This seemingly contradictory requirement means that a malicious receiver should not be able to run the extractor on his own.

This is the point at which we will use the disparity in the number of interactions that a malicious receiver can participate in, versus those that an extractor can participate in. Our techniques will be centered around the following question for cryptographic protocols between parties Alice and Bob:

Can extractor E with black-box access to Alice*, gain an advantage in just 2 messages,*
over (malicious) Bob *interacting with* Alice *in the actual protocol?*

As we have already discussed, we do not want to restrict the running time of Bob to be less than that of the extractor. Prior to our work, achieving black-box extraction in just 2 rounds from standard assumptions eluded all attempts at analysis.

Previous Work: Exponential Hardness Beyond $2^{n/2}$. In the exponential hardness regime, [KS17] devised one strategy to accomplish this, via the birthday paradox. They designed a reduction that would execute a committer several times, looking for birthday-style "collisions" in the transcripts

[4] The following text is largely copied directly from [LPS17].

generated by the committer. Their scheme was designed so that finding such collisions would help the reduction succeed. On the other hand, an adversary interacting with the same committer would only obtain polynomially many transcripts and would have no hope of being able to find a collision.

Unfortunately, the birthday strategy in [KS17] hits a fundamental roadblock because there can be at most a quadratic gap between the power of the reduction and the adversary – inherently limiting this approach to requiring strong and nonstandard exponential hardness assumptions beyond $2^{n/2}$ hardness.

2.1 Our Approach: Extractable Commitments

We devise a completely new simulation strategy that allows the reduction to gain an advantage over a malicious receiver potentially running in more time than the reduction itself. To understand this simulation strategy, it might be helpful to consider the following analogy of extractable commitments, to a forest invaded by a leprechaun.

Villagers in a leprechaun-haunted forest. A leprechaun is an invisible being that can steal objects without the victim's realization. We will think of every execution of the committer as being analogous to a villager taking *one* random walk in a forest. We imagine that this forest has been invaded by a single leprechaun released by the evil queen (receiver). If the villager (committer) inadvertently crosses the leprechaun, then the leprechaun is able to "steal" or extract the villager's gold coin (committed value) and hand it over to the queen. On the other hand if the villager does not cross the leprechaun, then the villager's coin (committed value) remains well-hidden. The villager only crosses the forest a few times, so the queen has only a few chances to steal the villager's coin. (This corresponds to only polynomially many interactions between a committer and a T-time malicious receiver.)

At the same time, we think of the extractor as being a powerful wizard that covets the villagers' gold coins. This wizard can force any villager to keep going back into the forest again and again, until a leprechaun is able to successfully steal the villager's coin. (This corresponds to a T' – less than T-time extractor being able to run the committer T' –greater than polynomially many times.)

We want several properties to hold for our villager-leprechaun interactions:

- We do not want that the evil queen to steal the villager's coins: In other words, we want the probability that a villager crosses a leprechaun, to be very small in any individual walk through the forest.
- On the other hand, the wizard should be able to force the villager to enter the forest so many times, that the poor villager would be sure to unknowingly encounter the leprechaun at least once.
- Why is it important that the leprechaun be invisible? Because we are rooting for the wizard: We don't want a villager to know that a leprechaun is nearby, potentially allowing the villager to substitute a brass coin for his golden one (i.e. we don't want a malicious committer to allow the distribution of extracted values to be different from the honest values).

Implementing leprechauns. We now turn to describing the construction of extractable commitments. The commitments will be hiding against T-time receivers, and yet will be extractable by T'-time extractors where T' is much smaller than T. Formally, we will write $T' \ll T$ to mean that T' is smaller than T multiplied by any polynomial in the security parameter. At this point in the technical overview, it will be useful to assume that we have two idealized technical tools. We will in fact make do with less ideal tools, as we discuss later[5]. For now, assume that we have the following two primitives that can be leveraged to be secure against T-time adversaries:

- Two-message two-party computation, against semi-honest senders and malicious receivers.
- Two-message ZK arguments, with polynomial simulation.

The leprechauns described above will be implemented using secure two-party computation for the following functionality: $\mathcal{F}\big((x, M), y\big) = \left\{ \begin{array}{cl} \perp & \text{if } x \neq y \\ \text{M} & \text{if } x = y \end{array} \right\}$

Intuitively, this functionality denotes the committer choosing path x and the leprechaun choosing path y, such that the leprechaun steals the committed message M if and only if $x = y$.

More formally, the receiver will sample a random challenge $\mathsf{ch} \xleftarrow{\$} \{0,1\}^m$ and the committer will sample another challenge $r \xleftarrow{\$} \{0,1\}^m$ independently. In order to commit to message M, the committer and receiver run secure two-party computation for $\mathcal{F}\big((r, M), \mathsf{ch}\big)$. The committer will also prove, via SPS ZK, that he correctly computed the output of the functionality.

Note that a malicious receiver, running in time T and participating in only a single execution, will have probability at most 2^{-m} of guessing the committer's challenge r. Thus, the commitment will still be computationally hiding against such a receiver.

On the other hand, an extractor that interacts with the committer super-polynomially many times, will have a good probability of obtaining at least one "extracting" transcript where $\mathsf{ch} = r$, and will thus find M after only slightly more than $T' = 2^m$ attempts. We must also ensure that the distribution over messages M output by the extractor is indistinguishable from the actual distribution of committed messages. We will exploit the security of two-party computation protocol against semi-honest senders, and additional complexity leveraging to ensure that the distribution of values committed by the committer cannot change between extracting transcripts and transcripts that don't allow extraction.

Finally, note that in this construction, the honest receiver is only required to verify the SPS ZK argument (which is public coin) – and doesn't actually need to observe the output of the two-party computation protocol. Thus, such a receiver can sample uniformly random coins to compute his message for the two-party computation protocol.

This completes an informal description of our extractable commitment, and we have the following (informal) theorem:

Informal Theorem 1. *Let n denote the security parameter. Assume sub-exponential security of DDH, together with sub-exponentially hard one-way permutations and sub-exponential ZAPs. Then there exists a statistically binding two-round public-coin extractable commitment scheme, that is hiding against malicious receivers running in time T and extractable in time $T' \ll T$.*

For technical reasons, our actual construction of extractable commitments is a slight variant of the scheme outlined above. This construction is described in Section 5, Figure 3.

In fact, this type of extractable commitment is really the main technical tool that we will use to obtain our main result on non-malleable commitments.

[5] It turns out that two round secure two-party computation with indistinguishability-based security, together with two-round zero-knowledge with super-polynomial simulation(SPS), will suffice. If uniform reductions are required, the two-round SPS ZK can be replaced with two-round strong WI [JKKR17] at the cost of requiring private coins.

2.2 Two-Message Non-Malleable Commitments w.r.t. Commitment

2.2.1 Model

Our main result is the construction of a public-coin bounded-concurrent two-message non-malleable commitment scheme with respect to commitment, assuming sub-exponentially hard ZAPs, sub-exponential one-way permutations, and sub-exponential hardness of DDH. We also get a private coin construction assuming only sub-exponential DDH or QR or N^{th} residuosity.

Very roughly, non-malleability requires that a man-in-the-middle adversary participating in two executions, acting as a receiver interacting with an honest committer in a "left" execution, and acting as committer interacting with an honest receiver in a "right" execution, is unable to commit to a message $\widetilde{m}$ on the right, that is nontrivially related to the message m committed by the honest committer on the left.

We require non-malleability against both synchronous and asynchronous adversaries. A synchronous MIM adversary observes an honest receiver message on the right, and then generate its own (malicious) receiver message for the left execution. Then, on obtaining an honestly generated left commitment, it generates a (malicious) right commitment. An asynchronous adversary is one that completes the entire left commitment, before generating its own right commitment. Typically (and this will especially be true in our situation), it is more difficult to prove security against synchronous adversaries than against asynchronous adversaries.

In this paper, we consider a setting where parties have identities or tags, typically in $[2^n]$ and only require non-malleability to hold when the tag used by the adversary is different from the tag used by an honest party. We note that this can be compiled in a standard way (using one-time signatures) to a notion without tags that requires the MIM's committed message to be independent from that of the honest committer, unless the MIM copies the entire left transcript [DDN91].

We now discuss a basic scheme, secure in a restricted setting where there are only two tags in the system, and the MIM's tag is guaranteed to be different from the honest committer's tag.

2.2.2 A basic scheme for two tags

The impossibility in [Pas13] is stated for the setting of just two tags, therefore overcoming it using sub-exponential assumptions is already non-trivial. As stated in the introduction, this will require us to exploit the gap between the number of executions available to the MIM versus those available to the reduction.

Recall that we achieved two-round extractable commitments with $T_{\mathsf{hid}} \gg T_{\mathsf{Ext}}$, that is secure against malicious receivers running in time T_{hid}, while extractable by extractors running in time $T_{\mathsf{Ext}} \ll T_{\mathsf{hid}}$. Having achieved such an extractable commitment scheme, in order to obtain a non-malleable commitment scheme for just two tags, we rely on the recent ideas of [KS17] (who needed exponential hardness). We adapt their ideas to our setting of standard subexponential assumptions.

Let us first consider a one-sided non-malleable commitment: Suppose there are two tags 0 and 1. Then a one-sided non-malleable commitment would guarantee that the commitment with tag 1 cannot depend on a commitment with tag 0, but it would potentially enable arbitrary malleability in the other direction. Pass and Wee [PW] demonstrated how to obtain a *one-sided* non-malleable commitment in this setting, based on sub-exponential assumptions.

We now illustrate how the gap between extraction and hiding of our two-round extractable commitment scheme can be used to enable two-sided non-malleable commitments, by appropriately leveraging hardness to exploit this gap. We use a two-round extractable commitment $\mathsf{ext\text{-}com}$ with security parameter n, that is extractable in time T_{Ext} and hiding against adversaries running in time $T_{\mathsf{hid}} \gg T_{\mathsf{Ext}}$. We also make use of a non-interactive commitment com leveraged so that it is hiding

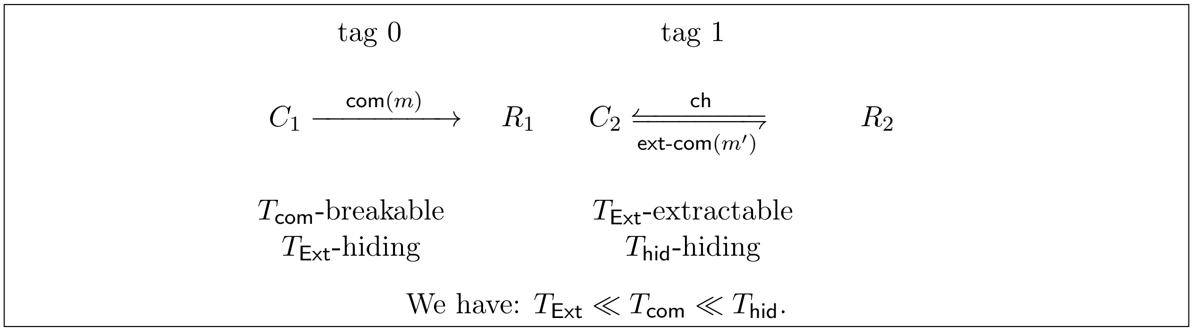

tag 0
tag 1
C_1
$\mathsf{com}(m)$
R_1
C_2
ch
$\mathsf{ext\text{-}com}(m')$
R_2
T_{com}-breakable
T_{Ext}-hiding
T_{Ext}-extractable
T_{hid}-hiding
We have: $T_{\mathsf{Ext}} \ll T_{\mathsf{com}} \ll T_{\mathsf{hid}}$.

Figure 1: A scheme for two tags

against adversaries running in time T_{Ext}, and trivially breakable in time T_{com}. We set parameters such that $T_{\mathsf{hid}} \gg T_{\mathsf{com}} \gg T_{\mathsf{Ext}}$. Then consider the following protocol:

- If $\mathsf{tag} = 0$, commit to the message m using the non-interactive commitment scheme com.
- If $\mathsf{tag} = 1$, commit to the message m using the extractable commitment scheme $\mathsf{ext\text{-}com}$.

This scheme is represented in Figure 1. We consider two representative settings, one where the man-in-the-middle (MIM) is the receiver on the left, and the committer on the right (thus, $R_1 = C_2$), and second, where the MIM is the receiver on the right, and committer on the left (thus, $R_2 = C_1$).

First, we consider the case where an honest committer uses tag 0 to commit to message m, while the MIM uses tag 1. A challenger against the hiding of the non-interactive commitment com, can obtain $\mathsf{com}(0)$ or $\mathsf{com}(m)$ externally, and then exploit the extractability of $\mathsf{ext\text{-}com}$ that is being used by the MIM, to extract the value committed by the MIM, in time T_{Ext}.

However, the non-interactive commitment is hiding against adversaries running in time T_{Ext}. Thus, if the MIM's commitment is related to m, such a challenger can break hiding of com, by extracting the value committed by the MIM, which contradicts the T_{Ext}-hiding of the non-interactive commitment.

Next, let us consider the complementary case where an honest committer uses tag 1 to commit to message m, while the MIM uses 0. A challenger against the hiding of the extractable commitment $\mathsf{ext\text{-}com}$, can obtain $\mathsf{ext\text{-}com}(0)$ or $\mathsf{ext\text{-}com}(m)$ externally, and then break com that is being used by the MIM, via brute-force to extract the value committed by the MIM, in time T_{com}.

However, $\mathsf{ext\text{-}com}$ is hiding against adversaries running in time $T_{\mathsf{hid}} \gg T_{\mathsf{com}}$. Thus, if the MIM's commitment is related to m, such a challenger can break hiding of $\mathsf{ext\text{-}com}$, by breaking the commitment of the MIM using com, and extracting the value committed, in time only $T_{\mathsf{com}} \ll T_{\mathsf{hid}}$, contradicting the hiding of $\mathsf{ext\text{-}com}$. We must now extend the above construction for two tags, all the way to tags in $[2^n]$. Pass and Wee [PW] noted that assuming sub-exponential hardness, it is possible to obtain $O(\frac{\log n}{\log\log n})$ levels of hardness. Thus, simple complexity leveraging, even if it could be used in some way, would not help us directly go beyond $O(\frac{\log n}{\log\log n})$ tags. As a first step, we describe how the construction above can be extended to a constant number of tags.

2.2.3 A construction for constant number of tags

Note that the 2-tag construction relied on extractability of $\mathsf{ext\text{-}com}$ to achieve non-malleability when the adversary uses $\mathsf{tag} = 1$. Implicit in the description above, was a crucial reliance on the *non-interactivity* of the other (non-extractable) commitment.

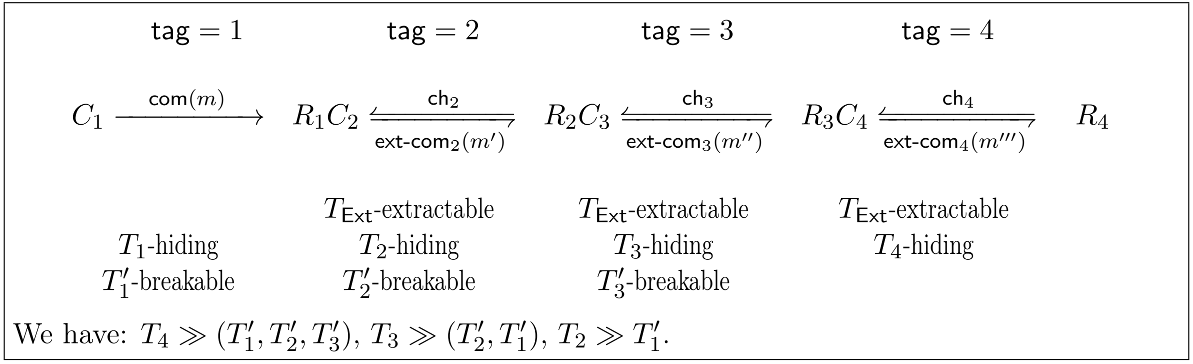
Indeed, a problem arises when using $\mathsf{ext\text{-}com}$ on both sides: the extractor that extracts from the MIM on the right, naturally needs to rewind the MIM. This may result in the MIM implicitly rewinding the honest committer, possibly causing extraction even from the honest committer. If the honest commitment is non-interactive, this is not a problem because it is possible to send the *same externally obtained string* to the MIM, every time the honest committer interaction is rewound. In other words, there is no rewinding allowed in the left interaction. However, if the honest interaction consists of two rounds, then the initial challenge of the MIM to the honest committer may change, and require a new response on the left from the honest committer. How should we simulate this response?

Let us illustrate this issue more concretely: A natural way of extending our 2-tag construction to a constant number of tags is illustrated in Figure 2, with parameters of various extractable commitment schemes adjusted (via leveraging, like in Figure 1) to ensure that:

1. For every pair of tags $\mathsf{tag} > \mathsf{tag}'$, the commitment for tag is hiding with respect to the time it takes to brute-force break the commitment for tag'.

2. The commitment associated with each tag is extractable in time less than the time with respect to which hiding is guaranteed all the tags: thus when $\mathsf{tag} < \mathsf{tag}'$ we will extract the commitment for tag' while trying to rely on the hiding of tag.

In the figure, by T-breakable, we always mean that the underlying commitment in $\mathsf{ext\text{-}com}$ is breakable using brute-force in time T.



We have: $T_4 \gg (T'_1, T'_2, T'_3)$, $T_3 \gg (T'_2, T'_1)$, $T_2 \gg T'_1$.

Figure 2: An illustrative natural (but incomplete) extension to four tags

We recall (from the two-tag case) that an extractor has two possible strategies, depending on whether the honest tag is larger or smaller than the MIM's tag. If the MIM's tag is smaller than the honest tag, then it is possible to argue non-malleability by breaking (via brute-force) the commitment generated by the MIM. This part of the argument goes through exactly as in the two-tag case.

However, the proof runs into the subtle issue mentioned above when the MIM's tag is larger than the honest tag. In this case, the reduction must run the extractor on the commitment generated by the MIM. However, every time the MIM is rewound by the extractor (using different challenges for the $\mathsf{ext\text{-}com}$), the MIM may generate its own fresh challenges for the honest commitment. Therefore, while extracting from the MIM, we may end up inadvertently also be extracting from the honest commitment – which would not let us achieve any contradictions. Recall that the entire point of this experiment was to extract from the man-in-the-middle while preserving hiding in the commitment generated by the honest committer.

Our Solution. Our main idea to solve this problem is as follows: We set our parameters in such a way that we can "modulate" the extractability of the commitment scheme. In other words, when the MIM's tag is larger than the honest tag, the MIM's commitment will be extractable in time $T_{\mathsf{Ext},\mathsf{tag}'}$ that is *much smaller than* the time taken to extract from the honest commitment $T_{\mathsf{Ext},\mathsf{tag}}$.

In a nutshell, we will set challenge spaces (for extraction) so that, when the MIM's tag is larger than the honest tag, the MIM's challenge space is also exponentially larger than the honest challenge space. This is accomplished, in particular, just by setting the length of ch corresponding to tag, to be $(\mathsf{tag} \times p(n))$, where $p(n)$ is some fixed (small) polynomial in the security parameter n.

Not only this, we will in fact require that the honest commitment corresponding to tag be *hiding* even under $T_{\mathsf{Ext},\mathsf{tag}'}$ attempts to extract from it. This will be achieved by leveraging the advantage of the adversaries in SPS ZK and secure two-party computation appropriately. We will still be careful so that time taken for *any extraction* will be much smaller than the time required to break hiding of any of the commitments. The flexibility of our construction of extractable commitments ensures that we can set parameters appropriately.

Bounded-Concurrent Security. We also prove a stronger security guarantee about the scheme outlined above, that is, we consider a setting where the MIM participates in $\ell(n)$ sessions with honest receiver(s) in which he acts as malicious committer, while obtaining a single commitment from an honest committer. We require that the *joint distribution* of the view and value committed by the MIM is unrelated to the message committed by the honest committer[6].

We prove $\ell(n)$-bounded-concurrent non-malleability of the scheme described above for polynomial $\ell(n) \ll m$, where m denotes the length of the challenge string ch for extraction. We need to set parameters appropriately for bounds $\ell(n)$. To ensure $\ell(n)$-bounded non-malleability, in the sessions where MIM commits to messages, we require an an extractor that extracts the *joint distribution* of messages committed by the MIM committer.

However, upon careful observation, our extraction strategy turns out to have the following problem: The extractor extracts the value from some "rare" transcripts, and when the MIM generates multiple transcripts simultaneously, the rare transcripts that the extractor is able to extract from, may not occur simultaneously at all. We therefore need to modify the extraction strategy to keep running until it succeeds in simultaneously extracting from all of the MIM's transcripts. Note that this only happens when the extractor is able to guess *all* the challenges generated by the MIM in all its commitment sessions.

In order for such an extractor to contradict non-malleability, we need to set the parameters large enough so that hiding of the challenge commitment holds even against adversaries running in time $\mathcal{T}$, where $\mathcal{T}$ is the time taken to extract from *all* the MIM's sessions simultaneously. This helps prove bounded-concurrent non-malleability.

Our techniques for handling a constant number of tags as well as bounded-concurrent non-malleability are novel and very specific to our construction. Next, we bootstrap a (sub-exponentially secure) non-malleable commitment scheme for just 4 tags into a scheme for all tags, in a way that only requires two rounds, and preserves bounded-concurrent non-malleability. Before this, we will review a new technical tool that will help in our two-round tag amplification scheme.

2.2.4 Two-round Zero-Knowledge with Superior Superpolynomial Simulation

Standard constructions of two round zero-knowledge arguments with superpolynomial simulation can be described as follows: the verifier generates a challenge that is hard to invert by adversaries

[6] This notion is called one-many non-malleability (with a bounded number of right executions), and implies many-many non-malleability [PR05b, LPV].

running in time T, then the prover proves (via a ZAP) that either the statement being proven is in the language, or that he knows the inverse of the challenge used by the verifier. This ZAP is such that the witness used by the prover can be extracted (via brute-force) in time $T' \ll T$. Naturally, this restricts the argument to be zero-knowledge against verifiers that run in time $T_{\mathsf{zk}} \ll T' \ll T$.

Thus, if a prover generates an accepting proof for a false statement, the ZAP can be broken in time T' to invert the challenge, leading to a contradiction. On the other hand, there exists a simulator that runs in time $T_{\mathsf{Sim}} \gg T$ to invert the receiver's challenge and simulate the proof (alternatively, such a simulator can non-uniformly obtain the inverse of the receiver's challenge). Thus, we have $T_{\mathsf{Sim}} \gg T_{\mathsf{zk}}$.

We define Zero-Knowledge with Superior Superpolynomial Simulation (SPSS-ZK) as ZK with super-polynomial simulation, such that $T_{\mathsf{Sim}} \ll T_{\mathsf{zk}}$. At first glance such a primitive may seem impossible to realize, but let us revisit the construction of SPS ZK described above, through the lens of non-malleability.

In order to ensure soundness, what we actually require is that a cheating prover, be unable to "maul" the challenge sent by the verifier, into a witness for his own ZAP. A simple way to do this is to use complexity leveraging to get one-sided non-malleability, which is what the construction described above achieves.

However, this *constrains* $T' \ll T$, which in turn constrains $T_{\mathsf{Sim}} \gg T_{\mathsf{zk}}$. We would like to look for a different way of achieving non-malleability, which potentially allows $T' \gg T$. In other words, we would like a more efficient way of extracting the witness from the NIWI than directly breaking it via brute force. This is *exactly* the kind of non-malleability that is supported our basic construction of two-sided non-malleable commitments for two tags from Section 2.2.2.

Specifically, we will just let the verifier use a non-interactive non-malleable commitment corresponding to $\mathsf{tag} = 0$, whereas the prover will use a two-message non-malleable (extractable) commitment corresponding to $\mathsf{tag} = 1$. We can now set parameters such that $T \ll T'$, which allows $T_{\mathsf{Sim}} \ll T_{\mathsf{zk}}$. On the other hand, in order to ensure soundness, we rely on the *extractability* of the prover's commitment in time $T_{\mathsf{Ext}} \ll T$.

We will use this primitive in the next subsection, while performing tag amplification while preserving bounded-concurrent non-malleability. We also believe that this primitive may be of independent interest. The construction and analysis can be found in Section 5.2.

2.2.5 Two-round tag amplification from 4 tags

Suppose we had a non-malleable commitment scheme for tags in $[2n]$. The popular DDN [DDN91] encoding, and a recent work [KS17] suggest a method of breaking a large tag T^j (say, in $[2^n]$) into n small tags $t_1^j, t_2^j, \ldots t_n^j$, such that for two different large tags $T^1 \neq T^2$, there exists at least one index i such that $t_i^2 \not\in \{t_1^1, t_2^1, \ldots t_n^1\}$. As in other tag amplification schemes [Wee10, LP], we will recursively apply an encoding with the property specified above. The scheme in [KS17] allows us to start with a one-one secure scheme for just 4 tags and amplify to obtain a (one-one secure) scheme for tags in $[2^n]$. However, we would like to amplify tags in such a way that we are able to bootstrap from a bounded-concurrent non-malleable commitment scheme for 4 tags to a bounded-concurrent non-malleable commitment scheme for all tags. To achieve this, we use ideas from [KS17], but describe the entire construction here for completeness.

Given the property of the encoding scheme in [DDN91, KS17], we consider the following construction (which is a round-compressed version of several constructions in prior work): To commit to a value with large tag T, commit to the value multiple times with small tags $t_1, t_2, \ldots t_n$ corresponding to T. Simultaneously, provide a 2-round ZK proof that all commitments are to the same value. We require the proof to be ZK against adversaries running in time T, where T is the time required to brute-force break (all components of) the underlying non-malleable commitment scheme for small tags.

In order to prove $\ell(n)$-bounded-concurrent non-malleability of the resulting scheme, we will focus on the index i_j in the MIM's j^{th} commitment, for $j \in \ell(n)$, such that the tag $\tilde{t}_{i_j} \notin \{t_1^1, t_2^1, \dots t_n^1\}$. In the real interaction, by soundness of the ZK argument, the value committed by the MIM is identical to the value committed using $\tilde{t}_{i_j}$. Thus, it suffices to argue that this value is generated independent of the honest committer's value. Because the argument is ZK against adversaries running in time T (that is, $T_{\mathsf{zk}} \gg T$), where T is the time required to brute-force break (all components of) the non-malleable commitment with $\tilde{t}_{i,j}$, the value committed remains indistinguishable even when a challenger generates the honest commitment by simulating the ZK proof.

Next, it is possible to switch commitments using tags $t_1^1, t_2^1, \dots t_n^1$ one by one, while the joint distribution of the values committed using tag $\tilde{t}_{i_j}$ does not change, because of $\ell(n)$-bounded concurrent non-malleability of the underlying commitment scheme. Note that here we are running in super-polynomial time T_{Sim}, so we require non-malleability to hold even against T_{Sim}-time adversaries. By our constraint on the ZK property of the argument, we will end up requiring that $T_{\mathsf{Sim}} \ll T_{\mathsf{zk}}$. This is exactly where our two-round SPSS ZK helps.

We note that this amplification can be applied recursively, several times, until non-malleability is obtained for all tags in $[2^n]$. The resulting protocol for tags in $[2^n]$ still only uses $\mathsf{poly}(n)$ commitments with small tags. We note that at each recursion, the ZK proof we use will require stronger parameters. However, since the tag space grows exponentially, starting with a constant number of tags, recursion only needs to be applied $O(\log^* n)$ times. Thus, we only require $O(\log^* n)$ levels of security for the ZK and for the non-malleable commitments, which can be obtained based on sub-exponential hardness, as was also shown by Pass and Wee [PW]. Apart from minor technical modifications to ensure that the resulting protocol remains efficient, this is essentially how we construct non-malleable commitments for larger tags. Our construction is formally described and proved in Section 6, and we have the following informal theorem.

Informal Theorem 2. *Assume sub-exponential security of DDH, together with sub-exponentially hard one-way permutations and sub-exponential ZAPs. Then there exists a constant bounded-concurrent statistically binding two-round public-coin non-malleable commitment scheme with respect to commitment.*

2.2.6 Instantiating the primitives

Throughout the discussion above, we assumed some idealized 2-round primitives, most notably a 2-round ZK argument, and 2-round secure two party computation. We note that everywhere above, the 2-round ZK argument can be instantiated with the work of Pass [Pas03] that builds 2-round super-polynomial simulation ZK arguments. At the same time, however, it turns out that our proofs only need a distinguisher-dependent notion of simulation called weak ZK. Recently, a construction of such weak ZK arguments (albeit with private coins) was given in [JKKR17], and by using this recent construction we also enjoy the ability to instantiate this 2-round weak ZK argument from any of the subexponential assumptions given in the set $\mathcal{Y}$ referenced in our informal theorem statements above.

Obtaining 2-round secure two-party computation is simpler: We can use 2-round OT, secure against malicious receivers, together with garbled circuits to implement this; OT security guarantees hiding of the receiver input against semi-honest senders. We additionally rely on leveraging to ensure that the sender input is chosen independently of the receiver input. To argue sender input-indistinguishability, we require a way to extract the OT receiver's choice bits, so that we can invoke the security of the garbled circuit scheme. Since we only require an indistinguishability-based guarantee, we could simply rely on non-uniformity to extract the OT receiver's choice bits and obtain a reduction to the security of garbled circuits. Another option is to adapt the proof strategy in [JKKR17] to provide distinguisher-based polynomial extraction of the OT choice bits that suffice in the circumstances where we need sender input-indistinguishability.

2.3 One Round Non-Malleable Commitments w.r.t. Opening, with Simultaneous Messages

Reordering Non-Malleable Commitments. We note that the two-message commitment schemes described so far indeed required the committer to generate a message depending on the receiver's challenge message. As such, compressing the protocol into a single round appears to be a difficult task. At the very least, we would like to force the committer to be bound to his message by the end of the first round.

We observe that the extractable commitments described in Section 2.1 can be deconstructed into two sub-protocols that occur in parallel: one sub-protocol (which we will call the commitment sub-protocol) is used to generate the actual commitment, and the other sub-protocol (which we will call the extraction sub-protocol) consists of the two-party computation together with proof of correct computation. The extraction sub-protocol is carried out purely to assist the extractor. Furthermore, the sub-protocol that generates the commitment can be made completely non-interactive by using a non-interactive statistically binding commitment based on injective one-way functions.

Moreover, the relative ordering of messages between these sub-protocols can be arbitrarily altered without affecting security. More specifically, we can reorder the extractable commitment, into the following different (still, two-round) extractable commitment in the simultaneous exchange model: In the first round of simultaneous exchange, the committer sends the commitment sub-protocol, whereas the receiver sends the first message of the extraction sub-protocol. In the second round of simultaneous exchange, the committer responds to the receiver's message for the extraction sub-protocol. This reordered scheme satisfies the same extraction properties as the previously considered scheme. In fact, in the simultaneous exchange model, this reordered scheme has an additional property: the committer is bound to his message by the end of the first round.

The non-malleable commitment scheme described previously can be similarly reordered, as we illustrate in more detail in Section 7. At this point, we have a two round non-malleable commitment scheme $\mathsf{NM}-\mathsf{Com}$, with respect to commitment, in the simultaneous exchange model, that is binding in the first round.

Non-Malleability with respect to Opening. We define non-malleability with respect to opening by requiring that the joint distribution of the view (including both the commit and opening phase) and the value *committed* by the MIM remain indistinguishable between real and simulated executions. Of course, in the real experiment, the MIM obtains the honest committer's opening once the commit phase is over, and therefore, the simulator is also given the honest committer's opening. This definition is similar to several previously considered definitions, with the main exception being that it allows super-polynomial simulation (this restriction is because of the two-round setting). In particular, our definition implies the recent indistinguishability-based definition in [GKS16]. We refer the reader to Section 4.2 for more details.

The natural next step, after obtaining a non-malleable commitment scheme in the simultaneous message model, that is binding in the first round, is to try and push the second message of the non-malleable commitment into the opening phase, and send this message together with an opening. However, we must ensure that the scheme is binding, and also that a man-in-the-middle is unable to create arbitrary malleations after obtaining an opening. In order to achieve this, we will again use SPSS ZK in a crucial way.

We accomplish this by setting up the opening phase in a specific way, additionally making use of an SPSS ZK argument, with low T_{Sim} (lower than other parameters of the $\mathsf{NM}-\mathsf{Com}$) and high T_{zk} (higher than other parameters of the $\mathsf{NM}-\mathsf{Com}$). The receiver sends the first message of the SPSS ZK argument in the first round, together with the first receiver message of $\mathsf{NM}-\mathsf{Com}$. Simultaneously, the committer sends the first round commitment message of $\mathsf{NM}-\mathsf{Com}$. This marks the end of the commitment phase.

In order to open, the committer sends the second message of the commitment $\mathsf{NM}-\mathsf{Com}$, together with the message committed (but not the randomness), and an SPSS ZK argument that the message opened actually corresponds to the committed value. Why is this construction secure?

We consider a simple sequence of hybrid experiments: In the first hybrid, the challenger starts simulating the SPSS ZK argument (of opening), and since T_{zk} is higher than the parameters of $\mathsf{NM}-\mathsf{Com}$, we note that the value committed by the $\mathsf{NM}-\mathsf{Com}$ (jointly with the overall view of the MIM in the commitment and opening phases) remains indistinguishable. Next, the challenger changes the value of the $\mathsf{NM}-\mathsf{Com}$ from committing to the honest committer's message to committing to 0, while still opening to M and simulating T_{zk}. Because the $\mathsf{NM}-\mathsf{Com}$ scheme is non-malleable against adversaries running in time T_{Sim}, the joint distribution of the value committed by the MIM and the view of the MIM (including the opening phase) remain indistinguishable. This is exactly the simulated experiment.

In particular, since the simulator does not have access to the honest committer's message in the commitment phase, and yet must open to this message in the opening phase, the simulator is required to equivocate in some way. Naturally, the commitment is required to be (computationally) binding in the real execution. However, since the joint distribution of the view and committed value remain indistinguishable between the real and simulated worlds – this means that the MIM remains computationally bound to his committed value *even while the simulator equivocates.*

This property ends up being useful from an application point of view, as can be observed in our construction of two round multi-party coin tossing. The multi-party coin tossing protocol, formally described in Section 7.3 is simple to construct: it only requires each party to (non-malleably) commit to a random input in the first round, and then open this commitment in the second round. Naturally, this protocol is round optimal.

We also obtain *fully concurrent two round non-malleable commitments with respect to commitment* in the simultaneous message setting (where the MIM can participate as malicious committer and malicious receiver in an unbounded number of sessions), full details of which are provided in Appendix A. We use these to obtain *fully concurrent one-round non-malleable commitments with respect to opening* in the simultaneous exchange setting. These protocols make a more central use of SPSS ZK, in fact they work by first modifying the SPSS ZK to obtain a variant of simulation soundness, and then using techniques similar to those of [LPV09] to obtain concurrent non-malleability. We believe that our round-optimal non-malleable protocols will find several other interesting applications, to low-round secure computation. We conclude with the following informal theorem.

Informal Theorem 3. *Assume sub-exponential security of DDH, together with sub-exponentially hard one-way permutations and sub-exponential ZAPs. Then there exists a fully concurrent statistically binding two-round public-coin non-malleable commitment scheme with respect to commitment, in the simultaneous exchange model. Furthermore, there exists a one round fully concurrent statistically binding public-coin non-malleable commitment scheme with respect to opening, in the simultaneous exchange model.*

3 Preliminaries

Here, we recall some preliminaries that will be useful in the rest of the paper. We will typically use n to denote the security parameter. We will say that $T_1(n) \gg T_2(n)$ if $T_1(n) > T_2(n) \cdot n^c$ for all constants c. When we say (sub-exponentially secure) one-way permutations exist, in fact it suffices to assume a family of (sub-exponentially secure) onto one-way functions where the specification of the function is public coin. We note that we only require this assumption to obtain public-coin variants of our protocols.

3.1 ZK With Superpolynomial Simulation.

We will use two message ZK arguments with superpolynomial simulation (SPS) [Pas04].

Definition 1 (Two Message $(T_{\mathsf{Sim}}, T_{\mathsf{zk}}, \delta_{\mathsf{zk}})$-ZK Arguments With Superpolynomial Simulation)**.** *We say that an interactive proof (or argument) $\langle P, V\rangle$ for the language $L \in \mathsf{NP}$, with the witness relation R_L, is $(T_{\mathsf{Sim}}, T_{\mathsf{zk}}, \delta_{\mathsf{zk}})$-simulatable if for every T_{zk}-time machine V^* exists a probabilistic simulator $\mathcal{S}$ with running time bounded by T_{Sim} such that the following two ensembles are $T_{\mathsf{zk}}, \delta_{\mathsf{zk}}$)-computationally indistinguishable (when the distinguishing gap is a function in $n = |x|$):*

- $\{(\langle P(y), V^*(z)\rangle(x))\}_{z\in\{0,1\}^*, x\in L}$ *for arbitrary* $y \in R_L(x)$
- $\{\mathcal{S}(x, z)\}_{z\in\{0,1\}^*, x\in L}$

That is, for every probabilistic algorithm D running in time polynomial in the length of its first input, every polynomial p, all sufficiently long $x \in L$, all $y \in R_L(x)$ and all auxiliary inputs $z \in \{0,1\}^*$ it holds that

$$\Pr[D(x, z, (\langle P(y), V^*(z)\rangle(x)) = 1] - \Pr[D(x, z, S(x, z)) = 1] < \delta_{\mathsf{zk}}(n)$$

Definition 2. *We say that a two-message $(T_{\mathsf{Sim}}, T_{\mathsf{zk}}, \delta_{\mathsf{zk}})$-SPS ZK argument satisfies non-uniform simulation (for delayed statements) if we can write the simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ where $\mathcal{S}_1(V^*(z))$, which outputs σ, runs in T_{Sim}-time, but where $\mathcal{S}_2(x, z, \sigma)$, which outputs the simulated view of the verifier V^*, runs in only polynomial time.*

3.2 Special Two-Party Computation

Definition 3 (Special Two-Party Computation)**.** *Special two-message two-party secure computation involves a protocol Π between a sender $\mathcal{S}$ with input x, and receiver $\mathcal{R}$ with input y, who obtains the output $f(x, y)$. The first message is sent by $\mathcal{R}$ as a function of y and receiver's randomness r_R, and we will denote this message by $\tau_1 = \mathsf{2PC}_R(1^n, y; r_R)$. The second message is sent by $\mathcal{S}$ as a function of (τ_1, x, r_S), which we will denote by $\tau_2 = \mathsf{2PC}_S(\tau_1, x; r_S)$. We will denote by $\mathsf{View}_{\mathcal{R}}(x)$ the tuple (x, τ_1, τ_2) where τ_1 is generated by $\mathcal{R}$, and τ_2 is generated by $\mathcal{S}$ with uniform randomness r_S.*

We will require the following properties:

- **(Perfect) Correctness:** *For all x, y, and honest $\mathcal{S}$ and $\mathcal{R}$, we have that the output obtained by $\mathcal{R}$ equals $f(x, y)$.*
- **Receiver Input-Hiding against T-time Senders:**
  For any T-time distinguisher $\mathcal{D}$, for all y_1, y_2, over the random choice of r_R:

$$|\Pr[\mathcal{D}(\mathsf{2PC}_R(1^n, y_1; r_R)) = 1] - \Pr[\mathcal{D}(\mathsf{2PC}_R(1^n, y_2; r_R)) = 1]| \leq 1/T.$$

- **Sender Input-Indistinguishability against T'-time Receivers:**
  There exists a constant $c > 0$ such that for large enough n, and for any T'-time malicious receiver $\mathcal{R}^*$, and T'-time distinguisher $\mathcal{D}$ that obtains the view of the receiver, for all f and all distributions $(\mathcal{X}_1, \mathcal{X}_2)$ such that for any y, if over random choice of $x_1 \overset{\$}{\leftarrow} \mathcal{X}_1$ and $x_2 \overset{\$}{\leftarrow} \mathcal{X}_2$, we have $\Pr[f(x_1, y) = f(x_2, y)] \geq 1 - \epsilon(n)$, then following is true when $\mathcal{S}$ and $\mathcal{R}^*$ run Π for f:
  $$|\Pr[\mathcal{D}(\mathsf{View}_{\mathcal{R}^*}(x_1)) = 1] - \Pr[\mathcal{D}(\mathsf{View}^*_{\mathcal{R}}(x_2)) = 1]| \leq (\epsilon(n) + 1/T') \cdot n^c.$$

Definition 4 (T'-Oblivious Special Two-Party Computation)**.** *A special two-message two-party secure computation protocol Π is said to have the T'-obliviousness property if the following holds. There exists a procedure $\mathsf{oSamp}(1^n)$ such that for any T'-time distinguisher $\mathcal{D}$, we have $\Pr[\mathcal{D}(r' \leftarrow \mathsf{oSamp}(1^n; r_O)) = 1 | r_O \overset{\$}{\leftarrow} \{0,1\}^*] - \Pr[\mathcal{D}(\mathsf{2PC}_R(1^n, y; r_R)) = 1 | (y, r_R) \overset{\$}{\leftarrow} \{0,1\}^*] \leq 1/T'$. Furthermore, there exists an efficient algorithm $\mathsf{Explain}$ such that $\mathsf{oSamp}(1^n; \mathsf{Explain}(\mathsf{2PC}_R(1^n, y; r_R))) = \mathsf{2PC}_R(1^n, y; r_R)$ for all values of y, r_R for sufficiently large n.*

Remark 1. *Since we only require indistinguishability-based security, secure two-message two party computation with super-polynomial simulation, where the receiver's message can be obliviously sampled, already implies this definition. This can be directly constructed [BGI$^+$17] using oblivious transfer and garbled circuits, where OT is secure against malicious receivers [NP01, HK12]. The security reduction can also be made uniform via the techniques of [JKKR17]. Our protocols can be instantiated with any special two-party computation satisfying the required properties. Finally, we note that the oblivious sampling property is only required to ensure that the resulting protocol is public coin.*

4 Definitions

We define a T-time machine as a non-uniform Turing Machine that runs in time at most T. All honest parties in definitions below are by default uniform interactive Turing Machines, unless otherwise specified.

4.1 Non-Malleability w.r.t. Commitment

Throughout this paper, we will use n to denote the security parameter, and $\mathsf{negl}(n)$ to denote any function that is asymptotically smaller than $\frac{1}{\mathsf{poly}(n)}$ for any polynomial $\mathsf{poly}(\cdot)$. We will use PPT to describe a probabilistic polynomial time machine. We will also use the words "rounds" and "messages" interchangeably.

We follow the definition of non-malleable commitments introduced by Pass and Rosen [PR05b] and further refined by Lin et al [LPV] and Goyal [Goy11] (which in turn build on the original definition of [DDN91]). In the real interaction, there is a man-in-the-middle adversary MIM interacting with a committer $\mathcal{C}$ (where $\mathcal{C}$ commits to value v) in the left session, and interacting with receiver $\mathcal{R}$ in the right session. Prior to the interaction, the value v is given to C as local input. MIM receives an auxiliary input z, which might contain a-priori information about v. Then the commit phase is executed. Let $\mathsf{MIM}_{\langle C,R \rangle}(\mathsf{value}, z)$ denote a random variable that describes the value $\widetilde{\mathsf{val}}$ committed by the MIM in the right session, jointly with the view of the MIM in the full experiment. In the simulated experiment, a PPT simulator $\mathcal{S}$ directly interacts with the MIM. Let $\mathsf{Sim}_{\langle C,R \rangle}(1^n, z)$ denote the random variable describing the value $\widetilde{\mathsf{val}}$ committed to by $\mathcal{S}$ and the output view of $\mathcal{S}$. If the tags in the left and right interaction are equal, the value $\widetilde{\mathsf{val}}$ committed in the right interaction, is defined to be $\perp$ in both experiments.

Concurrent non-malleable commitment schemes consider a setting where the MIM interacts with committers in polynomially many (a-priori unbounded) left sessions, and interacts with receiver(s) in upto $\ell(n)$ right sessions. If any of the tags (in any right session) are equal to any of the tags in any left session, we set the value committed by the MIM to $\perp$ for that session. The we let $\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)^{\mathsf{many}}$ denote the joint distribution of all the values committed by the MIM in all right sessions, together with the view of the MIM in the full experiment, and $\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)^{\mathsf{many}}$ denotes the joint distribution of all the values committed by the simulator $\mathcal{S}$ (with access to the MIM) in all right sessions together with the view.

Definition 5 (Non-malleable Commitments w.r.t. Commitment)**.** *A commitment scheme $\langle C, R\rangle$ is said to be non-malleable if for every PPT MIM, there exists a PPT simulator $\mathcal{S}$ such that the following ensembles are computationally indistinguishable:*

$$\{\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*} \text{ and } \{\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*}$$

Definition 6 ($\ell(n)$-Concurrent Non-malleable Commitments w.r.t. Commitment)**.** *A commitment scheme $\langle C, R\rangle$ is said to be $\ell(n)$-concurrent non-malleable if for every PPT MIM, there exists a PPT simulator $\mathcal{S}$ such that the following ensembles are computationally indistinguishable:*

$$\{\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)^{\mathsf{many}}\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*} \text{ and } \{\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)^{\mathsf{many}}\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*}$$

We say that a commitment scheme is fully concurrent, with respect to commitment, if it is concurrent for any a-priori unbounded polynomial $\ell(n)$.

4.2 Non-Malleability w.r.t. Opening

We also consider a strong notion of non-malleability w.r.t. opening, where we consider a superpolynomial time simulator $\mathcal{S}$ that interacts with the MIM. We will only consider non-malleability w.r.t. opening for a special type of commitments, that we call semi-statistically binding commitments, which we now define.

Definition 7 (Semi-Statistically Binding Commitments)**.** *We call a commitment scheme semi-statistically binding if upon completion of the commitment phase, the commitment can be opened in two modes: the first mode is statistically binding and the second mode is computationally binding against PPT committers. Naturally, the binding property requires that for any commitment transcript, the values opened (by any malicious PPT committer) in both modes are identical with overwhelming probability over the choice of transcripts. The value "committed" will refer to the value that is determined by the statistically binding mode.*

In actual applications, we will always use the semi-statistically binding commitment scheme in computationally binding mode. The statistically binding mode will only be considered for the purpose of security definitions, specifically, to define a notion of correctness of the committed value. We now proceed to our definition of non-malleability w.r.t. opening.

In the real interaction, there is a man-in-the-middle adversary MIM interacting with a committer $\mathcal{C}$ (where $\mathcal{C}$ commits to value v) in the left session, and interacting with receiver $\mathcal{R}$ in the right session. Prior to the interaction, the value v is given to C as local input. MIM receives an auxiliary input z, which might contain a-priori information about v. Then, the commitment **and decommitment** phase is executed. Whenever the MIM provides an invalid opening, his opening is defined to be $\perp$.

We consider semi-statistically binding commitments (that are always opened in computationally binding mode) and we define the value **committed** as the value that is determined by the statistically binding mode. However, if the MIM fails to provide a valid opening in any execution, the value committed is replaced with $\perp$ for the given execution.

Let $\mathsf{MIM}_{\langle C,R\rangle,\mathsf{open}}(\mathsf{value}, z)$ denote a random variable that describes the value $\widetilde{\mathsf{val}}$ **committed** by the MIM in the right session, jointly with the view of the MIM in the full experiment (this view includes the opening of the honest committer and the MIM). We note that the MIM could also potentially be equivocating, which is why we include the joint distribution of the committed and opened value.

In the simulated experiment, the super-polynomial time simulator $\mathcal{S}$ directly interacts with the MIM. $\mathcal{S}$ obtains the value v only after the commit phase is over. Let $\mathsf{Sim}_{\langle C,R\rangle,\mathsf{open}}(1^n, z)$ denote the random variable describing the value $\widetilde{\mathsf{val}}$ **committed** to by $\mathcal{S}$ (that is, the value corresponding to the statistically binding mode) and the output view of $\mathcal{S}$.

If the tags in the left and right interaction are equal, or if the MIM fails to open, the value $\widetilde{\mathsf{val}}$ committed in the right interaction, is set to $\perp$ in the above experiments.

Definition 8 (Non-malleable Commitments w.r.t. Opening)**.** *A commitment scheme $\langle C, R\rangle$ is said to be non-malleable with respect to opening if for every PPT* MIM*, there exists a PPT simulator $\mathcal{S}$ such that the following ensembles are computationally indistinguishable:*

$$\{\mathsf{MIM}_{\langle C,R\rangle,\mathsf{open}}(\mathsf{value}, z)\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*} \textit{ and } \{\mathsf{Sim}_{\langle C,R\rangle,\mathsf{open}}(1^n, z)\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*}$$

Concurrent non-malleable commitments with respect to opening naturally generalize the above to a setting where the MIM interacts with committers in polynomially many (a-priori unbounded) left sessions, and interacts with receiver(s) in upto $\ell(n)$ for some polynomial $\ell(\cdot)$) right sessions. If any of the tags (in any right session) are equal to any of the tags in any left session, or if the MIM doesn't provide a valid opening in any session, we set the value committed by the MIM to $\perp$ for that session. The we let $\mathsf{MIM}^{\mathsf{many}}_{\langle C,R\rangle,\mathsf{open}}(\mathsf{value}, z)$ denote the joint distribution of all the values committed by the MIM in all right sessions, together with the view of the MIM in the full experiment, and $\mathsf{Sim}^{\mathsf{many}}_{\langle C,R\rangle,\mathsf{open}}(1^n, z)$ denotes the joint distribution of all the values committed by the simulator $\mathcal{S}$ (with access to the MIM) in all right sessions together with the view. In this case, the simulator $\mathcal{S}$ obtains all the values for the honest executions after the commit phase is over.

Definition 9 ($\ell(n)$-Concurrent Non-malleable Commitments w.r.t. Opening)**.** *A commitment scheme $\langle C, R\rangle$ is said to be $\ell(n)$-concurrent non-malleable with respect to opening if for every PPT* MIM*, there exists a PPT simulator $\mathcal{S}$ such that the following ensembles are computationally indistinguishable:*

$$\{\mathsf{MIM}^{\mathsf{many}}_{\langle C,R\rangle,\mathsf{open}}(\mathsf{value}, z)\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*} \textit{ and } \{\mathsf{Sim}^{\mathsf{many}}_{\langle C,R\rangle,\mathsf{open}}(1^n, z)\}_{n\in\mathbb{N},v\in\{0,1\}^n,z\in\{0,1\}^*}$$

We say that a commitment scheme is fully concurrent, with respect to opening, if it is concurrent for any a-priori unbounded polynomial $\ell(n)$.We give some additional remarks about the above definitions:

- **Computational Binding for the** MIM **even in the Simulated Execution.** In Definition 8 and Definition 9, we require that the joint distribution of the view (including the MIM's opening) and value committed by the MIM (for the statistically binding mode) be indistinguishable between the real and ideal(simulated) executions. Note that in the real world, by the computational binding property of the scheme, the MIM's committed and opened values are *identical.* Therefore, if the committed and opened values are not identical in the ideal world, the two distributions described above are distinguishable. Thus, we have that even though the simulator may be equivocating in the ideal world, the MIM's commitment still remains binding[7].

- **Comparison with Prior Definitions.** Definition 8 is similar to several previously considered definitions of non-malleability with respect to opening, [PR05a, PR05b, OPV09, GKS16], except that it allows super-polynomial simulation. Also, some prior definitions did not take into account the joint distribution of the view and value opened, and unlike ours, did not compose well.

  In particular, Definition 8 implies the weaker (indistinguishability-based) definition of non-malleability with respect to opening given in [GKS16]. The definition in [GKS16] is especially suited to the uni-directional setting, where a simulation-based definition such as Definition 8 is impossible to achieve. They prove that for all messages m_0, m_1, the joint distribution of the view (excluding the honest opening) and *value opened* by a man-in-the-middle remains indistinguishable between the following two experiments: one, where the honest committer commits and opens to message m_0, and the second where an honest committer commits and opens to message m_1. In order to account for situations where a man-in-the-middle adversary adaptively chooses to abort based on whether the opening was to m_0 or m_1, [GKS16] allow an (unbounded) replacer to replace openings where the man-in-the-middle aborts with valid openings. Our definition can be directly seen to imply this definition, where the replacer strategy is as follows: run the simulator twice, providing openings both with respect to m_0 and m_1. By the binding property described in the previous bullet, the two openings of the MIM will not be different, with overwhelming probability. Thus, the replacer simply outputs any valid opened value produced by the MIM, or $\perp$ if the MIM aborts in both cases.

- **Using Non-Malleable Commitments w.r.t. Opening According to Definition 8.** Definition 8 is extremely versatile from the point of view of utility. Unlike some prior definitions of non-malleability with respect to opening, our definition allows for composability. Our simulation-based definition of one-one non-malleability *implies* one-many non-malleability, by a direct hybrid argument over the honest commitments, similar to the case of non-malleability with respect to commitment [LPV]. Furthermore, in fact, Definition 8 implies that *even the value committed by the* MIM – not just the value opened – does not change between executions in which the MIM provides a valid opening. This captures the strengthened intuition that as long as the MIM is able to provide a valid opening, the value committed by the MIM doesn't change between executions.

  These features help use non-malleable commitments with respect to opening satisfying Definition 8, to obtain low round protocols such as two-round multi-party coin tossing: In the first round, all parties commit to random coins via a non-malleable commitment with respect to opening. In the second round, parties open their committed coins and output the XOR of the coins of all parties. Prior to our work, such a two-round coin flipping scheme, where both parties obtain the same shared output, was not known even for the case of two parties. See Section 7.3 for the construction and proof of security.

[7] This property turns out to be important to achieve non-malleability. Indeed, if we only required that the value opened by the MIM (possibly, jointly with the view of the MIM) remain indistinguishable between real and simulated worlds, then the MIM could potentially be freely equivocating its own commitment when the simulator equivocates. Such definition would say nothing about non-malleability in the real (non-simulated) execution: and will be vacuously satisfied by any equivocal commitment scheme.

4.3 Extractable Commitments

We first (re)define commitment schemes while introducing some useful notation. Let n denote the security parameter.

Definition 10 (Commitment Scheme)**.** *A commitment scheme $\langle \mathcal{C}, \mathcal{R} \rangle$ is a two-phase protocol between a committer $\mathcal{C}$ and a receiver $\mathcal{R}$. At the beginning of the protocol, $\mathcal{C}$ obtains input m. Next, $\mathcal{C}$ and $\mathcal{R}$ execute the* Commit *phase. At the end of the* Commit *phase, $\mathcal{R}$ outputs* 0 *or* 1*.*

An execution of the Commit *phase between $(\mathcal{C}, \mathcal{R})$ with committer input m is denoted by $\langle \mathcal{C}, \mathcal{R} \rangle(m)$. The view of the receiver (including its coins and the transcript) at the end of this phase is denoted by $\mathsf{View}_{\mathcal{R}}\left(\langle \mathcal{C}, \mathcal{R} \rangle(m)\right)$.*

Next, only if $\mathcal{R}$ outputs 1 *in the* Commit *phase, $\mathcal{C}$ and $\mathcal{R}$ possibly engage in another interactive* Decommit *phase, at the end of which $\mathcal{R}$ outputs $\perp$ or a message $\widetilde{m}$. A commitment scheme should satisfy two security properties:*

- (T, δ)**-Hiding.** *For every message $M \in \{0,1\}^p$, for every probabilistic T-time receiver $\mathcal{R}^*$, every honest committer $\mathcal{C}$, and every T-time distinguisher $\mathcal{D}$ that obtains the view of the receiver,*

$$\Pr[\mathcal{D}\left(\mathsf{View}_{\mathcal{R}^*}\left(\langle \mathcal{C}, \mathcal{R}^* \rangle(\mathcal{M})\right)\right) = 1] - \Pr[\mathcal{D}\left(\mathsf{View}_{\mathcal{R}^*}\left(\langle \mathcal{C}, \mathcal{R}^* \rangle(0^n)\right)\right) = 1] \leq \delta(n)$$

- **Statistical Binding.** *There exists an (unbounded) extractor $\mathcal{E}_{\mathsf{Ideal}}$ such that the following holds: for every unbounded committer $\mathcal{C}^*$, let τ be the transcript generated by the interaction $\langle \mathcal{C}^*, \mathcal{R} \rangle(\cdot)$ in the* Commit *phase. Then $\mathcal{E}_{\mathsf{Ideal}}(\tau)$ outputs $\widetilde{m}_{\mathsf{Ideal}}$ such that the following holds: After $\mathcal{C}^*$ and $\mathcal{R}$ complete the* Decommit *phase, the probability that $\mathcal{R}$ outputs any value that is not $\perp$ or $\widetilde{m}_{\mathsf{Ideal}}$ is negligible.*

Definition 11 ($(\mathcal{T}, \mathcal{T}', T_C, \delta)$-Extractable Commitment Scheme)**.** *We say that a statistically binding, $\mathcal{T}_{\mathsf{hid}}$-hiding commitment scheme is $(\mathcal{T}, \mathcal{T}', T_C, \delta)$-extractable if there exists a $\mathcal{T}$-time uniform oracle machine $\mathcal{E}_{\mathsf{Real}}$ such that the following holds against all T_C-time adversarial committers $\mathcal{C}^*$:*
Ideal World. *In the ideal world, there is a sampling procedure* Samp *with black-box access to $\mathcal{C}^*$ that samples a committer view $\mathsf{View}_{\mathsf{Ideal}}$, which includes the committer's random coins and the transcript $\langle \mathcal{C}^*, \mathcal{R} \rangle(\cdot)$, together with some auxiliary output* aux *generated by the committer $\mathcal{C}^*$, using uniform random coins for the committer and receiver. Let $\mathcal{E}_{\mathsf{Ideal}}$ be the extractor for the transcript of $\mathsf{View}_{\mathsf{Ideal}}$ that outputs $\perp$ for any transcript not accepted by the verifier, and otherwise outputs the message embedded in the commitment guaranteed by statistical binding. Let $\widetilde{m}_{\mathsf{Ideal}}$ denote the message output by $\mathcal{E}_{\mathsf{Ideal}}$. The output of the ideal world $\mathsf{Exp}_{\mathsf{Ideal}}$ equals the joint distribution $(\mathsf{View}_{\mathsf{Ideal}}, \widetilde{m}_{\mathsf{Ideal}})$.*
Real World. *$\mathcal{E}_{\mathsf{Real}}$ obtains input committer views, denoted by random variable $\mathsf{View}_{\mathsf{Real}}$ via black-box access to $\mathcal{C}^*$, using uniform random coins for the committer and the receiver. The view $\mathsf{View}_{\mathsf{Real}}$ consists of the committer's random coins and the transcript $\langle \mathcal{C}^*, \mathcal{R} \rangle(\cdot)$, together with some auxiliary output* aux *generated by the committer $\mathcal{C}^*$. $\mathcal{E}_{\mathsf{Real}}^{\mathcal{C}^*}$ outputs $(\mathsf{View}_{\mathsf{Real}}, \widetilde{m}_{\mathsf{Real}})$. Let τ_{Real} denote the transcript from $\mathsf{View}_{\mathsf{Real}}$, and let $\mathsf{Exp}_{\mathsf{Real}} = (\mathsf{View}_{\mathsf{Real}}, \widetilde{m}_{\mathsf{Real}})$ be the output of the real world.*

Then, we require that two conditions hold:

- ***Correctness:***

$$\Pr[\mathcal{E}_{\mathsf{Ideal}}(\tau_{\mathsf{Real}}) = \widetilde{m}_{\mathsf{Real}}] = 1 - \delta(n)$$

- *$\mathcal{T}'$**-Indistinguishability:*** *For all $\mathcal{T}'$-time distinguishers $\mathcal{D}$,*

$$\left|\Pr[\mathcal{D}(\mathsf{Exp}_{\mathsf{Ideal}}) = 1] - \Pr[\mathcal{D}(\mathsf{Exp}_{\mathsf{Real}}) = 1]\right| \leq \delta(n)$$

Remark 2. *In this paper, we will also consider commitment schemes with simultaneous messages, that is, schemes where the committer and receiver simultaneously send messages to each other in the same round. The messages for any round can only depend on messages sent in previous rounds, and we only consider security against rushing adversaries, that wait for all honest messages being sent in a round before generating their own message for the same round.*

4.4 SPSS ZK

Definition 12 ($(T_\Pi, T_{\mathsf{Sim}}, T_{\mathsf{zk}}, T_L, \delta_{\mathsf{zk}})$-SPSS Zero Knowledge Arguments)**.** *We call an interactive protocol between a PPT prover P with input $(x, w) \in R_L$ for some language L, and PPT verifier V with input x, denoted by $\langle P, V\rangle(x, w)$, a super-polynomial superior simulation (SPSS) zero-knowledge argument if it satisfies the following properties and $T_\Pi \ll T_{\mathsf{Sim}} \ll T_{\mathsf{zk}} \ll T_L$:*

- **Completeness.** *For every $(x, w) \in R_L$, $\Pr[V \text{ outputs } 1 | \langle P, V\rangle(x, w)] \geq 1 - \mathsf{negl}(n)$, where the probability is over the random coins of P and V.*
- **T_Π-Adaptive-Soundness.** *For any language L that can be decided in time at most T_L, every x, every $z \in \{0, 1\}^*$, and every poly-non-uniform prover P^* running in time at most T_Π that chooses x adaptively after observing verifier message, $\Pr[\langle P^*(z), V\rangle(x) = 1 \ \wedge \ x \notin L] \leq \mathsf{negl}(n)$, where the probability is over the random coins of $\mathcal{V}$.*
- **$T_{\mathsf{Sim}}, T_{\mathsf{zk}}, \delta_{\mathsf{zk}}$-Zero Knowledge.** *There exists a (uniform) simulator $\mathcal{S}$ that runs in time T_{Sim}, such that for every x, every non-uniform T_{zk}-verifier V^* with advice z, and every T_{zk}-distinguisher $\mathcal{D}$: $\left|\Pr[\mathcal{D}(x, z, \mathsf{View}_{V^*}[\langle P, V^*(z)\rangle(x, w)]) = 1] \ - \Pr[\mathcal{D}(x, z, \mathcal{S}^{V^*}(x, z)) = 1]\right| \leq \delta_{\mathsf{zk}}(n)$*

4.5 Secure Computation

We will first define multi-party simulatable coin-tossing (with superpolynomial simulation), as well as pseudorandom multi-party coin tossing. The definition of simulatable coin tossing will follow [KOS03], except allowing for a super-polynomial time simulator.

Definition 13 (Multi-party Simulatable Coin-Tossing with Superpolynomial Time Simulation [KOS03])**.** *An $\mathcal{N}$-party protocol prot (for $\mathcal{N} = \mathsf{poly}(n)$), is a simulatable coin-flipping protocol if it is an $(n-1)$-secure protocol (with superpolynomial time simulation) realizing the coin-flipping functionality. That is, there exists some time bound T and a T-time simulator $\mathcal{S}$ such that for every PPT adversary $\mathcal{A}$ corrupting at most $(n-1)$ parties, that the (output of the) following experiments $\mathsf{REAL}(1^n, 1^\lambda)$ and $\mathsf{IDEAL}(1^n, 1^\lambda)$ are indistinguishable. Here we parse the result of running protocol prot with an adversary $\mathcal{A}$ (denoted by $\mathsf{REAL}(1^n, 1^\lambda)$) as a pair $(c, \mathsf{View}_\mathcal{A})$ where $c \in \{0, 1\}^\lambda \cup \{\bot\}$ is the outcome and $\mathsf{View}_\mathcal{A}$ is the view of the adversary $\mathcal{A}$.*

- *Experiment* $\mathsf{REAL}(1^n, 1^\lambda)$*:*
  1. $(c, \mathsf{View}_\mathcal{A}) \leftarrow \mathsf{REAL}_{\mathsf{prot},\mathcal{A}}(1^n, 1^\lambda)$
  2. *Output* $(c, \mathsf{View}_\mathcal{A})$
- *Experiment* $\mathsf{IDEAL}(1^n, 1^\lambda)$*:*
  1. $c' \leftarrow \{0, 1\}^\lambda$
  2. $\tilde{c}, \mathsf{View}_\mathcal{S} \leftarrow \mathcal{S}^\mathcal{A}(c', 1^n, 1^\lambda)$
  3. *If* $\tilde{c} = c', \bot$ *then output* $(\tilde{c}, \mathsf{View}_\mathcal{S})$

4. *Else output* fail

Definition 14 (Multi-party Pseudorandom Coin-Tossing). *An $\mathcal{N}$-party protocol* prot *(for $\mathcal{N} = \mathsf{poly}(n)$), is a pseudorandom coin-flipping protocol if it is a protocol where every party obtains output c (or the adversary aborts), and the following is true. For any adversary $\mathcal{A}$, we let $\mathcal{C}^*_{\mathcal{A}}$ denote the distribution of c generated in $\mathsf{REAL}(1^n, 1^\lambda)$ (where $c = \bot$ when the adversary aborts), we have for any PPT distinguisher $\mathcal{D}$ that obtains auxiliary input z,*

$$|\Pr[\mathcal{D}(r \xleftarrow{\$} \{0,1\}^\lambda, z) = 1] - \Pr[\mathcal{D}(c \xleftarrow{\$} \mathcal{C}^*_{\mathcal{A}}, z) = 1]| \le \Pr[\mathcal{A} \textit{ aborts}] + \mathsf{negl}(n)$$

5 Extractable Commitments

5.1 Construction

We describe our two-round extractable commitment scheme in Figure 3, where n denotes the security parameter. We use the following primitives:

- Let $\mathsf{com} = \mathsf{com}_1, \mathsf{com}_2$ denote a two-message statistically binding commitment scheme (which can be constructed from one-way functions). This commitment must be $(T_{\mathsf{com}}, \delta_{\mathsf{com}})$-hiding (this notation means that any T_{com}-time machine has advantage at most δ_{com} in breaking the hiding of the commitment scheme).

- Let Π_1, Π_2 denote the messages of a two-round zero-knowledge argument with super-polynomial time simulation (SPS-ZK), with non-uniform simulation (for delayed statements). This argument must be sound against adversaries running in time T_Π, except with probability δ_Π. Brute-force extraction of the witness from the argument should be possible in time T_{wext}. Zero-knowledge should hold such that adversaries running in time T_{zk} have advantage at most δ_{zk}. Finally, the running time of the simulator is T_{Sim}. Known constructions of SPS-ZK allow us to set these parameters as long as $T_{\mathsf{zk}} \ll T_{\mathsf{wext}} \ll T_\Pi \ll T_{\mathsf{Sim}}$.

- Let $(\mathcal{S}, \mathcal{R})$ be a special two-message two-party computation protocol for a function f that we will define in Figure 3. We require that the protocol achieve receiver input-hiding such that malicious receivers running in time T_{MalR} have advantage at most δ_{MalR}, and we require that the protocol achieves sender input-indistinguishability such that malicious senders running in time T_{MalS} have advantage at most δ_{MalS}. We will set these parameters so that $T_{\mathsf{MalR}} \ll T_{\mathsf{MalS}}$, which is a setting of parameters supported by known protocols.

Parameter Setting. The parameters m and p denote the challenge space and the length of message being committed respectively, and will be set according to our applications. For this section, it is useful, but not necessary, to assume that $m = p = n$.

We can set our parameters in any way so that $2^m \ll (T_{\mathsf{zk}}, T_{\mathsf{MalR}}, T_{\mathsf{com}}) \ll T_{\mathsf{wext}} \ll (T_\Pi, T_{\mathsf{MalS}}) \ll T_{\mathsf{Sim}}$.

We will now prove the following main theorem.

Theorem 1. *Assuming sub-exponentially hard DDH, sub-exponential one-way permutations and sub-exponential ZAPs, the scheme in Figure 3 is an extractable commitment scheme according to Definition 11.*

Lemma 1. *The scheme in Figure 3 is statistically binding.*

Proof. This follows from statistical binding of the underlying commitment scheme com. □

Function f: The 2-party function $f(x,y)$ is defined as follows. Parse $y = \mathsf{ch}$. Parse $x = (b, M, \widetilde{r}, r')$. If $b = 0$, output $\perp$. If $b = 1$, then check if $r' = \mathsf{ch}$. If so, output $(M, \widetilde{r})$. Otherwise, output $\perp$.
Language L: We define $L = \{(\mathsf{com}_1, \tau_1, c, \tau_2) : \exists r', \widetilde{r}, M, R : c = \mathsf{com}_2(m, \widetilde{r}, \mathsf{com}_1), \tau_2 = \mathsf{2PC}_S(\tau_1, x = (1, M, \widetilde{r}, r'); R)\}$.
Committer Input: Message $M \in \{0,1\}^p$.

Commit Stage: Receiver Message.

- Pick challenge string $\mathsf{ch} \xleftarrow{\$} \{0,1\}^m$.
- Compute $\tau_1 = \mathsf{2PC}_\mathsf{R}(1^n, \mathsf{ch})$.
- Compute Π_1 as first message of Π for language L, com_1 as first message of com, and send $\widehat{\mathsf{ch}} = (\tau_1, \Pi_1, \mathsf{com}_1)$ to the committer $\mathcal{C}$.

Committer Message.

- Sample randomness $\widetilde{r} \xleftarrow{\$} \{0,1\}^N$, and output $c = \mathsf{com}_2(M; \widetilde{r})$.
- Compute $r' \xleftarrow{\$} \{0,1\}^m$. Sample randomness R of length sufficient to compute $\tau_2 = \mathsf{2PC}_S(\tau_1, x = (1, M, \widetilde{r}, r'); R)$.
- Generate Π_2 as the second message of Π, proving that $(\mathsf{com}_1, \tau_1, c, \tau_2) \in L$.
- Send (c, τ_2, Π_2) to the receiver.

Verification. The receiver accepts the commitment if and only if Π_2 verifies.
Reveal Stage: The committer reveals randomness $\widetilde{r}$ to the receiver. The receiver accepts the decommitment to message M if and only if $c = \mathsf{com}(m, \widetilde{r})$.

Figure 3: Two-Round Extractable Commitments

Lemma 2. *The scheme in Figure 3 satisfies $(T_{\mathsf{hid}}, \delta_{\mathsf{hid}})$-hiding if: $T_{\mathsf{hid}} \ll T_{\mathsf{zk}}, T_{\mathsf{MalR}}, T_{\mathsf{com}}$, and $2^m \ll T_{\mathsf{MalR}}$, and $\delta_{\mathsf{hid}} \gg 2^{-m}, \delta_{\mathsf{zk}}, \delta_{\mathsf{com}}$.*

Proof. Suppose, for sake of contradiction, that there exist messages $M_0 \neq M_1$, and malicious T_{hid}-time receiver $\mathcal{R}^*$ and T_{hid}-time distinguisher $\mathcal{D}$, such that:

$$\Pr[\mathcal{D}\left(\mathsf{View}_{\mathcal{R}^*}\left(\langle \mathcal{C}, \mathcal{R}^* \rangle (M_0)\right)\right) = 1] - \Pr[\mathcal{D}\left(\mathsf{View}_{\mathcal{R}^*}\left(\langle \mathcal{C}, \mathcal{R}^* \rangle (M_1)\right)\right) = 1] > \delta_{\mathsf{hid}}(n)$$

We first consider a hybrid experiment Hybrid_1 where the committer uses message M_0, but instead of generating the message Π_2 using knowledge of the witness for the statement $(\mathsf{com}_1, \tau_1, c, \tau_2) \in L$, it executes the SPS ZK simulator to produce this statement. After this the experiment Hybrid_1 outputs the view of the receiver $\mathcal{R}^*$.

We now argue that

Claim 1.

$$\Pr[\mathcal{D}\left(\mathsf{View}_{\mathcal{R}^*}\left(\langle \mathcal{C}, \mathcal{R}^* \rangle (M_0)\right)\right) = 1] - \Pr[\mathcal{D}\left(\mathsf{Hybrid}_1\right) = 1] \leq \delta_{\mathsf{zk}}(n)$$

Proof of Claim. Suppose not. First, non-uniformly fix the first message of the receiver $\mathcal{R}^*$, and the output of the SPS ZK simulator $\mathcal{S}_1(\Pi_1) = \sigma$, in order to maximize the distinguishing advantage of $\mathcal{D}$. Now, suppose we obtain Π_2 as honestly generated using the prover's algorithm – this yields $\mathsf{View}_{\mathcal{R}^*}(\langle \mathcal{C}, \mathcal{R}^* \rangle(M_0))$. On the other hand, if we obtain Π_2 as the output of the (polynomial-time) simulator $\mathcal{S}_2$, then this yields Hybrid_1. By invoking the SPS ZK property, and because $T_{\mathsf{hid}} \ll T_{\mathsf{zk}}$, the claim follows. □

Observe that in Hybrid_1, because of ZK simulation, no secrets are needed to compute Π_2.

We next consider a hybrid experiment Hybrid_2 that works the same as Hybrid_1, except that the committer computes $\tau_2 = \mathsf{2PC}_S(\tau_1, x = (0, 0^p, 0^N, 0^m); R)$ instead of $\tau_2 = \mathsf{2PC}_S(\tau_1, x = (1, M_0, \widetilde{r}, r'); R)$.

We now argue that

Claim 2. *There exists a constant $c > 0$ such that*

$$\Pr[\mathcal{D}(\mathsf{Hybrid}_1) = 1] - \Pr[\mathcal{D}(\mathsf{Hybrid}_2)) = 1] \le 2^{-m} \cdot n^c$$

Proof of Claim. Fix any $y \in \{0,1\}^m$. Now, let $\mathcal{X}_1$ be the distribution that always outputs $x = (0, 0^p, 0^N, 0^m)$. Let $\mathcal{X}_2$ be the distribution that outputs $x = (1, M_0, \widetilde{r}, r')$ where $\widetilde{r} \xleftarrow{\$} \{0,1\}^n$ and $r' \xleftarrow{\$} \{0,1\}^m$. Then it follows immediately that

$$\Pr[f(\mathcal{X}_1, y) = f(\mathcal{X}_2, y)] = 1 - 2^{-m}$$

As in the proof of the previous claim, non-uniformly fix the first message of the receiver $\mathcal{R}^*$, and the output of the SPS ZK simulator $\mathcal{S}_1(\Pi_1) = \sigma$, in order to maximize the distinguishing advantage of $\mathcal{D}$. Recall again that the running time of the simulator $\mathcal{S}_2$ is polynomial-time. Now, suppose we obtain $\tau_2 = \mathsf{2PC}_S(\tau_1, x = (0, 0^p, 0^N, 0^m))$ – this yields Hybrid_2. On the other hand, if we obtain $\tau_2 = \mathsf{2PC}_S(\tau_1, x = (1, M_0, \widetilde{r}, r'))$, then this yields Hybrid_1.

By invoking the Sender Input-Indistinguishability property of the Special Two-Party Computation protocol, and because $T_{\mathsf{hid}} \ll T_{\mathsf{MalR}}$ and $2^m \ll T_{\mathsf{MalR}}$, the claim follows. □

Now observe that the only dependence on M_0 in Hybrid_2 is in the computation of the commitment $c = \mathsf{com}_2(M_0)$, and no steps in Hybrid_2 depend on the randomness used to generate this commitment.

Thus, we define a hybrid experiment Hybrid_3 that works the same as Hybrid_2, except that the committer computes $c = \mathsf{com}_2(M_1)$ instead of $c = \mathsf{com}_2(M_0)$ as in Hybrid_2. We now have:

Claim 3.

$$\Pr[\mathcal{D}(\mathsf{Hybrid}_2) = 1] - \Pr[\mathcal{D}(\mathsf{Hybrid}_3)) = 1] \le \delta_{\mathsf{com}}(n)$$

Proof of Claim. As in the proof of the previous claims, non-uniformly fix the first message of the receiver $\mathcal{R}^*$, and the output of the SPS ZK simulator $\mathcal{S}_1(\Pi_1) = \sigma$, in order to maximize the distinguishing advantage of $\mathcal{D}$. Recall again that the running time of the simulator $\mathcal{S}_2$ is polynomial-time. The claim then follows immediately from the definition of $(T_{\mathsf{com}}, \delta_{\mathsf{com}})$-hiding and the fact that $T_{\mathsf{com}} \gg T_{\mathsf{hid}}$. □

The lemma follows by carrying out the hybrid steps in reverse order with the message fixed to M_1.

□

Lemma 3. *The scheme in Figure 3 is a $(T_{\mathsf{Ext}}, T'_{\mathsf{Ext}}, T_C, \delta_{\mathsf{Ext}})$-extractable commitment scheme, where $T_{\mathsf{Ext}} = 2^m \cdot \mathsf{poly}(n) \cdot (1/\delta_{\mathsf{Ext}}) \cdot T_C$, and $(T'_{\mathsf{Ext}}, T_C, T_{\mathsf{wext}}) \ll (T_{\mathsf{MalS}}, T_\Pi)$, and $\delta_{\mathsf{Ext}} \gg 2^m \delta_{\mathsf{MalS}}$.*

Proof. We begin by describing the extractor. The running time of the extractor (as an oracle machine) will be $T_{\mathsf{Ext}} = 2^m \cdot (1/\delta_{\mathsf{Ext}}) \cdot \log(1/\delta_{\mathsf{Ext}}) \cdot m \cdot n^c$ for some constant $c > 0$. Here δ_{Ext} will be the extraction error; for most of our applications it will suffice to set $\delta_{\mathsf{Ext}} \leq \mathsf{negl}(n)$. The extractor proceeds as follows:

- Execute the following once. If Verification fails, output the view of the malicious sender and output $\perp$ as the extracted value. Otherwise, repeat the following up to $2^m \cdot (1/\delta_{\mathsf{Ext}}) \cdot \log(1/\delta_{\mathsf{Ext}}) \cdot m \cdot n^4$ times, until success. We call an individual performance of the following steps a *trial*:

  1. Choose $\mathsf{ch} \xleftarrow{\$} \{0,1\}^m$. Compute $\tau_1 = \mathsf{2PC_R}(1^n, \mathsf{ch}; R')$ using uniform randomness R'. Carry out the rest of the steps needed to compute the honest receiver's first message.
  2. Query the malicious sender $\mathcal{C}^*$ with this message, and obtain the sender's response (c, τ_2, Π_2).
  3. Verify the proof Π_2. As written above, if this is the first attempt, if verification fails, the experiment will not continue. However, if this is not the first attempt, we consider this a failure and move on to the next iteration. If verification succeeds, use R' and τ_2 to compute the output of the special two-party computation.
  4. If the output is $\perp$, the extractor considers this a failure and moves on to the next attempt.
  5. If the output is $(M^*, \widetilde{r})$, then the extractor considers this a success, and outputs the view of the malicious sender together with the message M^* (for some applications, we also require the extractor to output $\widetilde{r}$). (Note that the extractor does not bother to check if $c = \mathsf{com}_2(M^*, \widetilde{r})$. Looking ahead, it will rely on the soundness of the ZK argument to ensure that this happens often enough.)

We now analyze our extraction. We consider two cases:

- Case 1. Suppose $\Pr[(\mathcal{C}^*, \mathcal{R}) \text{ aborts}] > 1 - \delta_{\mathsf{Ext}}$. In this case, observe that the extractor also outputs aborting views with $\perp$ as the extracted message with probability greater than $1 - \delta_{\mathsf{Ext}}$, as would happen in the Ideal experiment. Therefore, we have both correctness and indistinguishability.

- Case 2. Suppose $\Pr[(\mathcal{C}^*, \mathcal{R}) \text{ aborts}] \leq 1 - \delta_{\mathsf{Ext}}$.

  Note that below, we will give the analysis conditioned on a non-aborting trial being output by the extractor. The general case will follow for correctness because the probability of a non-aborting trial exceeds δ_{Ext}.

  We first argue correctness, then we will argue indistinguishability.

  We have that the probability of seeing at least one non-aborting view is at least $1 - (\frac{\delta_{\mathsf{Ext}}}{n^2 \cdot 2^m})$ after $q = (1/\delta_{\mathsf{Ext}}) \cdot \log(1/\delta_{\mathsf{Ext}}) \cdot m \cdot n^3$ independent trials, by a Chernoff bound. Since the extractor performs $q \cdot 2^m \cdot n$ trials, by a union bound, except with probability $1 - \delta_{\mathsf{Ext}}/n$, the extractor obtains at least $2^m \cdot n$ non-aborting views with independently chosen ch.

  Now we prove the following claim:

  Claim 4. *In any individual trial, conditioned on the trial not aborting, the probability that the output of the special two-party computation is not $\perp$ is at least $2^{-m}(1-\delta)$, where $\delta = \delta_{\mathsf{Ext}} \cdot \mathsf{negl}$ and $\delta_{\mathsf{Ext}} \gg 2^m \delta_{\mathsf{MalS}}$.*

Proof of Claim. Suppose not. Recall that the special two-party computation produces a non-$\perp$ output when the r' chosen by the committer is equal to the ch chosen by the extractor in the trial. By soundness of Π, then, we have that the probability that $r' = \mathsf{ch}$ is below $2^{-m}(1-\delta) + \delta_{\Pi}$, where r' is obtained by running the witness extractor for Π, and ch is the challenge chosen in the extraction trial.

We will use this to contradict the Receiver Input-Hiding Security of the Special Two-Party Computation protocol. To do so, we describe our reduction algorithm $\hat{A}$ and distinguisher $\hat{\mathcal{D}}$ for the Receiver-Input Hiding Security game.

The reduction $\hat{A}$ simply chooses two challenges $\mathsf{ch}_1, \mathsf{ch}_2 \overset{\$}{\leftarrow} \{0,1\}^m$ at random, and creates auxiliary information consisting of these two challenges.

Now, our distinguisher $\hat{\mathcal{D}}$ will obtain as input either $\tau_1 = \mathsf{2PC}_R(1^n, \mathsf{ch} = \mathsf{ch}_1; r_R)$ or $\tau_1 = \mathsf{2PC}_R(1^n, \mathsf{ch} = \mathsf{ch}_2; r_R)$. The distinguisher now does the following:

- It generates the first message of the SPS ZK system Π_1 and the first message of the commitment com_1 honestly.
- It then runs the malicious sender $\mathcal{C}^*$ on the message $(\tau_1, \Pi_1, \mathsf{com}_1)$, obtaining the view View of the sender.
- If the proof message Π_2 within View does not verify, the distinguisher aborts.
- It then runs the ZK witness extraction procedure to obtain (r', M) from the proof message Π_2 within View. If the witness extraction fails (which can happen with probability δ_{Π}), the distinguisher aborts and outputs $\perp$.
- If $r' = \mathsf{ch}_1$, it outputs 1. Otherwise, it aborts and outputs $\perp$.

Let us now analyze the probability that $\hat{\mathcal{D}}$ outputs 1 in the two cases of $\mathsf{ch} = \mathsf{ch}_1$ and $\mathsf{ch} = \mathsf{ch}_2$. If $\mathsf{ch} = \mathsf{ch}_1$, then by assumption, we have that, conditioned on a non-aborting trial, $\Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_1] < 2^{-m}(1-\delta) + 2\delta_{\Pi}$. On the other hand, if $\mathsf{ch} = \mathsf{ch}_2$, then no information about ch_1 is given to the distinguisher $\mathcal{D}$. Therefore, conditioned on a non-aborting trial, $\Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_2] \geq 2^{-m} - \delta_{\Pi}$. This is a contradiction, if $|3\delta_{\Pi} - 2^{-m} \cdot \delta| > \delta_{\mathsf{MalS}}$. In particular, we have a contradiction if $2^{-m}\delta \gg \delta_{\mathsf{MalS}}$ and $\delta_{\Pi} \ll \delta_{\mathsf{MalS}}$. Note that we also used $(T_{\mathsf{MalS}}, T_{\Pi}) \gg (T_{\mathsf{wext}}, T_C)$, where T_C is the running time of the committer. □

Correctness then follows from a Chernoff bound and the facts that $2^m\delta_{\Pi} \ll 2^{-m}$ and $2^m/T_{\mathsf{MalS}} \ll 2^{-m}$.

Now we proceed to the proof that the extraction produces views and extracted messages that are indistinguishable from views and extracted messages drawn from the real distribution. Note that below, we will give the analysis conditioned on a non-aborting trial being output by the extractor. The general case will follow simply because the aborting views output by the extractor are identically distributed to aborting views output in the Ideal experiment, and therefore no distinguisher can gain an advantage on seeing aborting views. Let's suppose that there is a distinguisher that distinguishes between the Real and Ideal experiments for extraction. In other words, we have a T_C-time committer $\mathcal{C}^*$ and distinguisher $\mathcal{D}$ such that:

$$\left|\Pr[\mathcal{D}(\mathsf{Exp}_{\mathsf{Ideal}}) = 1] - \Pr[\mathcal{D}(\mathsf{Exp}_{\mathsf{Real}}) = 1]\right| > \delta_{\mathsf{Ext}}(n)$$

Without loss of generality, we can assume that there exists a probability p such that:

$$\Pr[\mathcal{D}(\mathsf{Exp}_{\mathsf{Ideal}}) = 1] = p - \delta$$

and

$$\Pr[\mathcal{D}(\mathsf{Exp}_{\mathsf{Real}}) = 1] = p$$

where $\delta > \delta_{\mathsf{Ext}}(n)$ (if $\delta < -\delta_{\mathsf{Ext}}(n)$, we can flip the output of the distinguisher).

We will use this distinguisher to contradict the Receiver Input-Hiding Security of the Special Two-Party Computation protocol. To do so, we first describe our reduction algorithm $\hat{A}$ and distinguisher $\hat{\mathcal{D}}$ for the Receiver-Input Hiding Security game.

The reduction $\hat{A}$ simply chooses two challenges $\mathsf{ch}_1, \mathsf{ch}_2 \xleftarrow{\$} \{0,1\}^m$ at random, and creates auxiliary information consisting of these two challenges.

Now, our distinguisher $\hat{\mathcal{D}}$ will obtain as input either $\tau_1 = \mathsf{2PC}_R(1^n, \mathsf{ch} = \mathsf{ch}_1; r_R)$ or $\tau_1 = \mathsf{2PC}_R(1^n, \mathsf{ch} = \mathsf{ch}_2; r_R)$. In a nutshell, $\hat{\mathcal{D}}$ will extract (via brute-force) the underlying committed value, and feed this to distinguisher $\mathcal{D}$ together with the view. However, since receiver input-hiding holds even against $\hat{\mathcal{D}}$ that breaks hiding of the commitment scheme, we will get a contradiction.

The distinguisher now does the following:

- ◦ It generates the first message of the SPS ZK system Π_1 and the first message of the commitment com_1 honestly.
- ◦ It then runs the malicious sender $\mathcal{C}^*$ on the message $(\tau_1, \Pi_1, \mathsf{com}_1)$, obtaining the view View of the sender.
- ◦ If the proof message Π_2 within View does not verify, the distinguisher aborts.
- ◦ It then runs the ZK witness extraction procedure to obtain (r', M) from the commitment message within View. If the witness extraction fails (which can happen with probability δ_Π), the distinguisher aborts and outputs $\perp$.
- ◦ If $r' = \mathsf{ch}_1$, it outputs $\mathcal{D}(\mathsf{View}, M)$. Otherwise, it aborts and outputs $\perp$.

Now, we analyze two cases.

Suppose $\mathsf{ch} = \mathsf{ch}_1$. Observe that if $r' = \mathsf{ch}_1$, then by correctness of extraction, this is the distribution that corresponds to the output of $\mathsf{Exp}_{\mathsf{Real}}$. In this case, conditioned on a non-aborting run, by Claim 4, we have that $\Pr[r' = \mathsf{ch}_1 | \mathsf{ch} = \mathsf{ch}_1] \geq 2^{-m}(1 - \delta_{\mathsf{Ext}} \cdot \mathsf{negl})$. Thus, $\Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_1] \geq p \cdot 2^{-m}(1 - \delta_{\mathsf{Ext}} \cdot \mathsf{negl}) - \delta_\Pi$.

Suppose $\mathsf{ch} = \mathsf{ch}_2$. Observe that this is the distribution that corresponds to the output of $\mathsf{Exp}_{\mathsf{Ideal}}$. In this case, no information about ch_1 is given to the adversary. Thus, conditioned on a non-aborting run, we have that $\Pr[r' = \mathsf{ch}_1 | \mathsf{ch} = \mathsf{ch}_2] \leq 2^{-m}$. Thus, $\Pr[\hat{\mathcal{D}} = 1 | \mathsf{ch} = \mathsf{ch}_2] \leq (p - \delta) \cdot (2^{-m}) + \delta_\Pi$.

As long as $2^{-m}\delta_{\mathsf{Ext}} \gg \delta_{\mathsf{MalS}}$, and $\delta_\Pi \ll \delta_{\mathsf{Ext}}$, and $(T_C, T'_{\mathsf{Ext}}, T_{\mathsf{wext}}) \ll (T_{\mathsf{MalS}}, T_\Pi)$, we reach a contradiction, and the lemma follows. □

Remark 3 (Uniform Reduction for Hiding)**.** *The SPS ZK used in the construction can be replaced by two-message delayed-input strong WI [JKKR17] in order to obtain a uniform reduction that proves hiding of the extractable commitment, however then the protocol no longer remains public-coin.*

Remark 4 (Public Coins)**.** *Assuming that Π is public-coin, and that $\mathsf{2PC}$ is a T'-oblivious special two-party computation protocol, the receiver can use oSamp to sample uniformly random coins in order to generate the first message for $\mathsf{2PC}$. This results in the protocol in Figure 3 being public coin.*

5.1.1 Modified Extractable Commitment

We remark that the extractable commitment scheme specified above, can be modified so that the commitment com is a non-interactive, statistically binding commitment (that can be based on injective one-way functions).

If required, the SPS ZK can also be replaced with SPSS ZK (described in the next section, Section 5.2)[8]. Replacing SPS ZK with SPSS ZK in the construction of extractable commitments, would yield a uniform simulation strategy that runs in time T_{Sim} (which is superpolynomial, yet less than all other parameters in the system) for proving hiding of the commitment scheme. Essentially the same proof of extraction goes through for , except that witness extraction from the ZK argument is performed using the extractable commitment that is within the SPSS ZK argument.

These modifications are not necessary for Section 6, but will be used in Section 7.

5.2 SPSS Zero Knowledge from Extractable Commitments

Our SPSS ZK protocol is described in Figure 4. We let n denote the security parameter.

We assume the existence of a two-round extractable commitment scheme, according to Definition 11, that is $(T_{\mathsf{hid}}, \delta_{\mathsf{hid}})$-hiding and $(T_{\mathsf{Ext}}, T'_{\mathsf{Ext}}, T_C, \delta_{\mathsf{Ext}})$-extractable, where $T_C \ll T_{\mathsf{Ext}}$, $T_{\mathsf{Ext}} \ll T_{\mathsf{hid}} \ll T'_{\mathsf{Ext}}$, and $\delta_{\mathsf{Ext}} = \mathsf{negl}(n)$. We denote its messages by $\mathsf{ext\text{-}com}_1, \mathsf{ext\text{-}com}_2(m; r)$.

We also assume the existence of a two-round witness-indistinguishable proof, denoted by zap such that adversaries running in time T_{wi} have advantage at most δ_{wi}.

Finally, we assume the existence of a non-interactive commitment scheme com. The public-coin version of our protocol assumes a non-interactive commitment scheme from one-way permutations, such that every string corresponds to a valid commitment. We only describe the public-coin version for simplicity. The private coin version can be obtained by replacing the commitment sent by the receiver with two commitments, along with a NIWI proof that one of the two is well-formed. We assume that com is hard to invert by adversaries running in time $T_{\mathsf{hid-com}}$, and can be broken (via brute-force) in time T_{com}, where $T_{\mathsf{hid-com}} \ll T_{\mathsf{com}}$.

It is straightforward to observe (by correctness of the zap) that the protocol satisfies correctness. We prove the following theorems about soundness and ZK properties of the protocol. We will leverage parameters so that $T_\Pi \ll T_{\mathsf{Sim}} \ll T_{\mathsf{zk}} \ll T_L$ in the following theorems, which will imply SPSS ZK. We prove the following two theorems.

Theorem 2 (T_Π-Adaptive-Soundness)**.** *For any language L that can be decided in time at most T_L, every x, every $z \in \{0,1\}^*$, and every poly-non-uniform prover P^* running in time at most T_Π that chooses x adaptively after observing verifier message,* $\Pr[\langle P^*(z), V\rangle(x) = 1 \ \wedge \ x \notin L] \leq \mathsf{negl}(n)$, *where the probability is over the random coins of $\mathcal{V}$; assuming that $(T_\Pi \cdot T_{\mathsf{Ext}}) \ll T_{\mathsf{hid-com}}$, and $T_L \ll T'_{\mathsf{Ext}}$.*

Proof. Suppose there exists a T_Π-time prover P^* that chooses $x \notin L$ in the last round, and over the random choice of $x \notin L$, $\Pr[\langle P^*(z), V\rangle(x) = 1] > \frac{1}{\mathsf{poly}(n)}$.

Then, consider reduction $\mathcal{R}$ to the hiding of com, which obtains com for the first message externally (as a commitment to M, where M is either $r_1 \xleftarrow{\$} \{0,1\}$ or $r_2 \xleftarrow{\$} \{0,1\}$) and then constructs committer $\mathcal{C}^*$ using the malicious prover, with the receiver and sender messages for $\mathsf{ext\text{-}com}$ and with the rest of the proof transcript as auxiliary information aux.

[8] We point out that SPSS ZK itself makes use of an extractable commitment. This means that SPS ZK is used to build $\mathsf{ext\text{-}com}_1$, which will be used to construct SPSS ZK, which in turn would be used to construct $\mathsf{ext\text{-}com}_2$.

Prover Input: Instance x, witness w such that $R(x, w) = 1$.
Verifier Input: Instance x.

1. Verifier V sends $e_1 = \mathsf{ext\text{-}com}_1$, $e_1' = \mathsf{ext\text{-}com}_1'$ to V, together with $c = \mathsf{com}(s; r)$ for $s \xleftarrow{\$} \{0,1\}^n, r \xleftarrow{\$} \{0,1\}^*$ and zap_1.
2. Verifier V picks $s' \xleftarrow{\$} \{0,1\}^n, (r', \widetilde{s}) \xleftarrow{\$} \{0,1\}^*$, computes $e_2 = \mathsf{ext\text{-}com}_2(s'; r')$ and $e_2' = \mathsf{ext\text{-}com}_2'(w; r'')$. It also computes zap_2 proving:
$$\exists w, r'' \text{ such that } w \text{ is a witness for } x \in L \ \wedge \ e_2' = \mathsf{ext\text{-}com}_2'(w; r'')\mathsf{OR}$$
$$\exists (s', r', r) \text{ such that } e_2 = \mathsf{ext\text{-}com}_2(s'; r') \ \wedge \ c = \mathsf{com}(s'; r).$$
It then sends (e_2, zap_2) to P.
3. **Verification.** The verifier accepts (outputs 1) if and only if zap verifies.

Figure 4: Two Round SPSS ZK Arguments

Since, with probability at least $\frac{1}{\mathsf{poly}(n)}$, the T_Π-time prover P^* outputs an accepting transcript for $x \notin L$, by soundness of the zap, we have that with probability at least $\frac{1}{\mathsf{poly}(n)}$, P^* generates an extractable commitment to $s' = M$.

$\mathcal{R}$ runs the extractor for $\mathsf{ext\text{-}com}$ on $\mathcal{C}^*$ – this takes time at most $T_\Pi \cdot T_{\mathsf{Ext}}$. The extractor outputs $(\mathsf{aux}, \mathsf{View}_P, P)$ that is indistinguishable except with negligible advantage (by T_{Ext}-time distinguishers) from the joint distribution of the view and value generated by the prover in $\mathsf{ext\text{-}com}$ in a random execution.

Since $T'_{\mathsf{Ext}} = T_L$, this also implies that with probability at least $\frac{1}{\mathsf{poly}(n)}$, the transcripts indeed have $x \notin L$ (as otherwise a T_L-time distinguisher would distinguish the real extracted values from ideal extracted values), thus the value output by the extractor is identical to M with significant probability.

This gives the reduction $\mathcal{R}$ non-negligible advantage in guessing the external challenge commitment com. Since $(T_\Pi \cdot T_{\mathsf{Ext}}) \ll T_{\mathsf{hid-com}}$, this is a contradiction to the hiding of the commitment scheme com. □

Theorem 3 (($T_{\mathsf{zk}}, \delta_{\mathsf{zk}}$)-Zero-Knowledge)**.** *There exists a simulator $\mathcal{S}$ that runs in time T_{Sim} such that for every x, every T_{zk} verifier V^* and every T_{zk} distinguisher $\mathcal{D}$,*

$$\left| \Pr[\mathcal{D}(x, z, \mathsf{View}_{V^*}[\langle P, V^*(z)\rangle(x, w)]) = 1] - \Pr[\mathcal{D}(x, z, \mathcal{S}^{V^*}(x, z)) = 1] \right| \leq \delta_{\mathsf{zk}}(n)$$

assuming that $T_{\mathsf{com}} \leq T_{\mathsf{Sim}} \ll T_{\mathsf{zk}} \ll T_{\mathsf{hid}}$, $T_{\mathsf{zk}} \ll T_{\mathsf{wi}}$, $\delta_{\mathsf{hid}} \geq \delta_{\mathsf{zk}}, \delta_{\mathsf{zap}} \geq \delta_{\mathsf{zk}}$.

Proof. The simulator Sim works as follows: it runs in time T_{com} to break (via brute-force) the string c and extract randomness (s, r). It then generates the prover message by picking randomness r', generating $e_2 = \mathsf{ext\text{-}com}_2(s; r')$, and generating zap_2 using (s, r, r') as witness. Thus, the running time of the simulator, $T_{\mathsf{Sim}} \geq T_{\mathsf{com}}$.

In order to prove zero-knowledge, we consider the following hybrids.

Hybrid_0 : This corresponds to an honest execution where the prover uses the witness to compute the ZK argument.

Hybrid_1 : In this hybrid, the simulator runs in time T_{com} to break (via brute-force) the string c and extract randomness (s, r). It then generates the prover message by picking randomness r', generating $e_2 = \mathsf{ext\text{-}com}_2(s; r')$, but still generates zap_2 using the real witness w for $x \in L$. Here, the running time of the simulator is $T_{\mathsf{com}} \ll T_{\mathsf{hid}}$. By $(T_{\mathsf{hid}}, \delta_{\mathsf{hid}})$-hiding of the $\mathsf{ext\text{-}com}$, this hybrid is indistinguishable from Hybrid_0 with advantage at most δ_{hid} against T_{hid}-time distinguishers.

Hybrid_2 : This hybrid corresponds to the simulator strategy, which is the same as Hybrid_1, except that the simulator generates zap_2 using (s, r, r') as witness. Here, the running time of the simulator is $T_{\mathsf{com}} \ll T_{\mathsf{wi}}$. By $(T_{\mathsf{wi}}, \delta_{\mathsf{wi}})$-witness indistinguishability of the zap, this hybrid is indistinguishable from Hybrid_1 with advantage at most δ_{wi} against T_{wi}-time distinguishers.

This proves that no T_{zk}-time verifier V^* and distinguisher $\mathcal{D}$ that distinguish the real view from the simulated view with advantage better than δ_{wi}, if $T_{\mathsf{zk}} \ll T_{\mathsf{wi}}$ and $T_{\mathsf{zk}} \ll T_{\mathsf{hid}}$. □

We note that this construction of SPSS ZK allows us to set parameters such that $T_{\mathsf{wext}} \ll T_{\Pi} \ll T_{\mathsf{Sim}} \ll T_{\mathsf{zk}} \ll T_L$, where T_{wext} is the time taken to extract the distribution of witnesses by extracting from the extractable commitment. Furthermore, we can use this SPSS ZK safely with any languages that can be decided in some a-priori bounded time, by setting T_L to be large enough. For the rest of this paper, while using SPSS ZK, we will assume that T_L is always set large enough.

6 Non-malleable Commitments

6.1 Non-Malleable Commitments for Two Tags

We begin by describing the first (simpler) construction of non-malleable commitments for two tags.

Here, in addition to the assumptions required for extractable commitments, we assume the existence of a non-interactive statistically binding commitment scheme, with security parameter κ, that can be broken (via brute-force) in time 2^{κ}, and whose security holds against adversaries running in time $2^{\kappa^{\epsilon}}$ for some $\epsilon > 0$. Such a scheme exists assuming sub-exponential injective one-way functions. We set parameters for the protocol as follows.

- If $\mathsf{tag} = 0$, we will use the extractable commitment scheme from Figure 3 which satisfies Definition 11 that has:
  - Security parameter n
  - Hiding against malicious receivers running in time at most $2^{n^{\epsilon}} \cdot \mathsf{poly}(n)$ for constant ϵ
  - An extractor that runs in time $2^{2m(n)}$ for $m(n) = \frac{n^{\epsilon^3}}{2}$
- If $\mathsf{tag} = 1$, we will use a non-interactive statistically binding commitment scheme that has:
  - Security parameter $\widetilde{n} = n^{\epsilon}$: thus can be broken via brute-force in time $2^{\widetilde{n}}$
  - Hiding against malicious receivers running in time at most $2^{\widetilde{n}^{\epsilon}} \cdot \mathsf{poly}(n)$ for constant ϵ

Statistical binding and computational hiding of the scheme follow from the statistical binding and computational hiding properties of the underlying extractable commitment scheme.

We will now sketch the proof of non-malleability (for intuition). Formal proofs can be found in Section 6.2. We let tag denote the tag used by an honest committer, participating in the left execution, and let $\mathsf{tag}' \neq \mathsf{tag}$ denote the tag used by the MIM participating in a right execution of the protocol. We only discuss the synchronous case here, since extractability makes the proof trivial in the asynchronous setting.

The simulator strategy is as follows: the simulator Sim generates c an honest commitment to 0, and outputs the transcript generated by the man-in-the-middle MIM on input this commitment c.

Proof Sketch: Non-malleability when $\mathsf{tag} = 0, \mathsf{tag}' = 1$. Our parameters are carefully aligned so that in this situation, the commitment scheme for $\mathsf{tag} = 0$ is hiding against malicious receivers running in time at most $2^{n^\epsilon} \cdot \mathsf{poly}(n)$ (thus such receivers have advantage at most $\mathsf{negl}(n)$). On the other hand, the commitment scheme for $\mathsf{tag} = 1$ can be broken via brute-force in time at most 2^{n^ϵ}.

Thus, we consider a reduction $\mathcal{R}$ that obtains (externally) for $\mathsf{tag} = 0$, a string c which is a commitment to msg, where msg is either M or 0, and runs the (PPT) MIM to obtain the view $\mathsf{View}_{\mathsf{MIM}}$ generated by the MIM. $\mathcal{R}$ then runs in time at most 2^{n^ϵ} to extract (via brute-force) the value $\mathsf{val}_{\mathsf{MIM}}$ committed by the MIM in $\mathsf{View}_{\mathsf{MIM}}$. Then, if there exists a PPT distinguisher $\mathcal{D}$ such that:

$$\Pr[\mathcal{D}(\mathsf{View}_{\mathsf{Real}}, \mathsf{val}_{\mathsf{Real}}) = 1] - \Pr[\mathcal{D}(\mathsf{View}_{\mathsf{Ideal}}, \mathsf{val}_{\mathsf{Ideal}}) = 1] \geq \mathsf{negl}(n),$$

$\mathcal{R}$ just echoes the output of $\mathcal{D}$ such that:

$$\Pr[\mathcal{D} = 1 | \mathsf{msg} = M] - \Pr[\mathcal{D} = 1 | \mathsf{msg} = 0] \geq \mathsf{negl}(n)$$

Since the running time of $\mathcal{R}$ is at most $2^{n^\epsilon} \cdot \mathsf{poly}(n)$, this contradicts hiding of the commitment scheme for $\mathsf{tag} = 0$.

Proof Sketch: Non-malleability when $\mathsf{tag} = 1, \mathsf{tag}' = 0$. Our parameters are carefully aligned so that in this situation, the commitment scheme for $\mathsf{tag} = 1$ is hiding against malicious receivers running in time at most $2^{n^{\epsilon^2}} \cdot \mathsf{poly}(n)$ (thus such receivers have advantage at most $\mathsf{negl}(n)$). On the other hand, the commitment scheme for $\mathsf{tag} = 1$ is extractable via an extractor that runs in time at most $2^{n^{\epsilon^3}}$.

Then, we consider the following reduction $\mathcal{R}$, that obtains (externally) for $\mathsf{tag} = 1$, a string c which is a commitment to msg, where msg is either M or 0, and uses the MIM to construct committer $\mathcal{C}^*$ that on input receiver message, runs the MIM, with auxiliary input c, and outputs the commitment generated by the MIM together with auxiliary information c.

$\mathcal{R}$ then runs the extractor in time at most $2^{n^{\epsilon^3}}$ on the (PPT) committer $\mathcal{C}^*$ to obtain output $(\mathsf{View}_{\mathsf{MIM}}, \mathsf{val}_{\mathsf{MIM}})$ that includes c as the commitment generated with $\mathsf{tag} = 1$. We remark that it is important that extraction from MIM can be done using the single externally obtained challenge as auxiliary input c.

Then, if there exists a PPT distinguisher $\mathcal{D}$ such that:

$$\Pr[\mathcal{D}(\mathsf{View}_{\mathsf{MIM}}, \mathsf{val}_{\mathsf{MIM}}) = 1 | \mathsf{msg} = M] - \Pr[\mathcal{D}(\mathsf{View}_{\mathsf{MIM}}, \mathsf{val}_{\mathsf{MIM}}) = 1 | \mathsf{msg} = 0] \geq \mathsf{negl}(n),$$

$\mathcal{R}$ just echoes the output of $\mathcal{D}$ such that:

$$\Pr[\mathcal{D} = 1 | \mathsf{msg} = M] - \Pr[\mathcal{D} = 1 | \mathsf{msg} = 0] \geq \mathsf{negl}(n)$$

Since the running time of $\mathcal{R}$ is at most $2^{n^{\epsilon^3}} \cdot \mathsf{poly}(n)$, this contradicts hiding of the commitment scheme for $\mathsf{tag} = 1$.

6.2 Non-Malleable Commitments for Four Tags

We now describe and formally prove security of a construction of non-malleable commitments for four tags.

Besides the assumptions required for extractable commitments, we assume the existence of a non-interactive statistically binding commitment scheme, with security parameter κ, whose security holds against adversaries running in time 2^{κ^ϵ} for some $\epsilon > 0$. Such a scheme exists assuming sub-exponential injective one-way permutations. We set parameters for the protocol as follows.

1. If $\mathsf{tag} \in [1,3]$, we will use the extractable commitment scheme from Figure 3 which satisfies Definition 11 and has:
   - Security parameter $n_{\mathsf{tag}} = n^{\epsilon^{\mathsf{tag}}}, m_{\mathsf{tag}} = \mathsf{tag} \cdot n^{\epsilon^{100}}$.
   - Can be broken via brute-force in time $2^{n_{\mathsf{tag}}}$.
   - $(2^{{n_{\mathsf{tag}}}^{\epsilon} \cdot \mathsf{poly}(n)}, 2^{-m_{\mathsf{tag}}} \cdot n^c)$- hiding against malicious receivers, for some c, ϵ.
   - An extractor that runs in time $2^{m_{\mathsf{tag}}} \cdot n^c$ for some c.
2. If $\mathsf{tag} = 4$, we will use a non-interactive statistically binding commitment scheme that has:
   - Security parameter $n_4 = n^{\epsilon^4}, m_4 = 4 \cdot n^{\epsilon^{100}}$.
   - Can be broken via brute-force in time 2^{n_4}.
   - $(2^{{n_4}^{\epsilon}} \cdot \mathsf{poly}(n), 2^{-m_4} \cdot n^c)$- hiding against malicious receivers, for some c, ϵ.

Figure 5: Non-malleable Commitments for Four Tags

Lemma 4. *The commitment scheme in Figure 5 is statistically binding and computationally hiding.*

Proof. The statistical binding and computational hiding properties (against PPT adversaries) follow from the statistical binding and computational hiding of the underlying extractable commitment scheme (for $\mathsf{tag} \in [1,3]$) or non-interactive commitment scheme (for $\mathsf{tag} = 4$). □

Theorem 4. *The scheme in Figure 5 is a non-malleable commitment scheme according to Definition 5, for 4 tags.*

Proof. We let tag denote the tag used by an honest committer, participating in the left execution, and let $\mathsf{tag}' \neq \mathsf{tag}$ denote the tag used by the MIM participating in a right execution of the protocol. We only discuss the synchronous case here, which is strictly harder to prove than the asynchronous case (extractability makes the proof trivial in the latter case).

Let $\mathsf{view}_{\mathsf{Real}}(M)$ denote the view and $\mathsf{val}_{\mathsf{Real}}(M)$ denote the value committed by the MIM in the real execution when the honest committer generates a commitment to some message M in the real world.

The simulator Sim generates an honest commitment to 0 with randomness r, and outputs the view generated by the MIM on input the honest commitment to 0. This corresponds to the simulated view in the ideal world. Let $\mathsf{view}_{\mathsf{Ideal}}$ denote the view and $\mathsf{val}_{\mathsf{Ideal}}$ denote the value committed by the MIM in the ideal execution in the ideal world.

Then for any M and any PPT distinguisher $\mathcal{D}$ that obtains input the view and value committed by the MIM, we will show that: $\Pr[\mathcal{D}(\mathsf{view}_{\mathsf{Real}}(M), \mathsf{val}_{\mathsf{Real}}(M)) = 1] - \Pr[\mathcal{D}(\mathsf{view}_{\mathsf{Ideal}}, \mathsf{val}_{\mathsf{Ideal}}) = 1] \leq \mathsf{negl}(n)$

The rest of our analysis is split into two cases.

Case I: Non-malleability when $\mathsf{tag} < \mathsf{tag}'$. Our parameters are carefully aligned such that the commitment scheme for tag is $(2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n), 2^{-m_{\mathsf{tag}}} \cdot n^c)$-hiding against malicious receivers, while the commitment scheme for tag' can be broken (via brute-force) in time $2^{n^{\epsilon^{\mathsf{tag}'}}}$ to extract the underlying committed message. Thus, the proof of non-malleability of this case follows from a complexity leveraging argument.

We consider a reduction $\mathcal{R}$ that obtains a commitment c with tag from an external challenger, to either M or to 0, and $\mathcal{R}$ will use MIM and the non-malleability distinguisher $\mathcal{D}$, to break hiding of the commitment scheme for tag as described below.

$\mathcal{R}$ runs the PPT MIM, with the left commitment substituted by the externally obtained challenge, and obtains the view $\mathsf{View}_{\mathsf{MIM}}$ generated by the MIM. $\mathcal{R}$ then runs in time at most $2^{n^{\epsilon^{\mathsf{tag}'}}}$ to extract (via brute-force) the value committed in the commitment generated by the MIM in $\mathsf{View}_{\mathsf{MIM}}$. Then, if there exists a PPT distinguisher $\mathcal{D}$ such that:

Then, if there exists a PPT distinguisher $\mathcal{D}$ such that:

$$\Pr[\mathcal{D}(\mathsf{View}_{\mathsf{Real}}(M), \mathsf{val}_{\mathsf{Real}}(M)) = 1] - \Pr[\mathcal{D}(\mathsf{View}_{\mathsf{Ideal}}, \mathsf{val}_{\mathsf{Ideal}}) = 1] \geq \frac{1}{\mathsf{poly}(n)},$$

$\mathcal{R}$ just echoes the output of $\mathcal{D}$ such that:

$$\Pr[\mathcal{D} = 1 | \mathsf{msg} = M] - \Pr[\mathcal{D} = 1 | \mathsf{msg} = 0] \geq \frac{1}{\mathsf{poly}(n)}$$

Since $\mathsf{tag}' \geq \mathsf{tag} + 1$, the running time of $\mathcal{R}$ is at most $2^{n^{\epsilon^{\mathsf{tag}'}}} \cdot \mathsf{poly}(n) \leq 2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n)$, $\mathcal{R}$ can use $\mathcal{D}$ and the MIM to break $(2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n), 2^{-m_{\mathsf{tag}}} \cdot n^c)$-hiding of the commitment scheme for tag.

This proves that the joint distribution of the view and value committed by the MIM is indistinguishable in the real and ideal worlds, in the case when $\mathsf{tag} < \mathsf{tag}'$.

Case II: Non-malleability when $\mathsf{tag} > \mathsf{tag}'$. Our parameters are carefully aligned such that the commitment scheme for tag is $(2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n), 2^{-m_{\mathsf{tag}}} \cdot n^c)$-hiding against malicious receivers, while commitment scheme for tag' is extractable via an extractor that runs in time at most $2^{m_{\mathsf{tag}'}} \cdot n^{c'}$. Thus, the proof of non-malleability of this case follows from extractability of the commitment scheme for tag'.

We will describe the reduction $\mathcal{R}$ that proves non-malleability, but we first describe an intermediate committer $\mathcal{C}^*$ on which $\mathcal{R}$ will run the extractor of the extractable commitment scheme.

The intermediate committer $\mathcal{C}^*$. We use the MIM to construct committer $\mathcal{C}^*$ that does the following. $\mathcal{C}^*$ on input a receiver message for tag', runs the MIM. If the MIM generates a receiver message corresponding to tag, $\mathcal{C}^*$ queries the reduction $\mathcal{R}$ to obtain an external commitment for tag, corresponding to the receiver message generated by the MIM. On input this external commitment, the MIM outputs its own commitment for tag'. $\mathcal{C}^*$ then outputs the commitment generated by the MIM corresponding to tag' as $\mathsf{View}_{\mathcal{C}^*}$, and as auxiliary information $\mathsf{aux}_{\mathcal{C}^*}$, it outputs the commitment for tag.

The reduction $\mathcal{R}$. Next, consider a reduction $\mathcal{R}$ against $(2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n), 2^{-m_{\mathsf{tag}}} \cdot n^c)$-hiding of the commitment scheme for tag. $\mathcal{R}$ runs the extractor on malicious committer $\mathcal{C}^*$, while answering the queries of $\mathcal{C}^*$. The extractor runs in time at most $2^{m_{\mathsf{tag}'}} \cdot n^{c'}$, and thus $\mathcal{C}^*$ may make at most $2^{m_{\mathsf{tag}'}} \cdot n^{c'}$ queries to $\mathcal{R}$. $\mathcal{R}$ responds to these queries by invoking the challenger for the hiding of the commitment scheme for tag, everytime a query is issued.

Finally, the extractor outputs a $(\mathsf{View}', \mathsf{Value}', \mathsf{aux}')$

Then, if there exists a PPT distinguisher $\mathcal{D}$ such that:

$$\Pr[\mathcal{D}(\mathsf{View}_{\mathsf{Real}}(M), \mathsf{val}_{\mathsf{Real}}(M)) = 1] - \Pr[\mathcal{D}(\mathsf{View}_{\mathsf{Ideal}}, \mathsf{val}_{\mathsf{Ideal}}) = 1] \geq \frac{1}{\mathsf{poly}(n)},$$

$\mathcal{R}$ just echoes the output of $\mathcal{D}$ on input the joint distribution $(\mathsf{View}', \mathsf{aux}', \mathsf{Value}')$.

The challenger. This challenger sample $b \xleftarrow{\$} \{0,1\}$, and if $b = 0$, it sets $\mathsf{msg} = 0$, else it sets $\mathsf{msg} = M$. Then on input a (malicious) receiver message outputs a commitment (using fresh randomness) to msg. It repeats this $2^{m_{\mathsf{tag}'}} \cdot n^{c'}$ times, thereby providing the receiver $2^{m_{\mathsf{tag}'}} \cdot n^{c'}$ different commitments to the same msg. Since the commitment scheme for tag is $(2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n), 2^{-m_{\mathsf{tag}}} \cdot n^{c})$-hiding, a simple hybrid argument accross all commitments provided by the challenger implies that no malicious receiver running in time at most $2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n)$ has advantage better than $(2^{m_{\mathsf{tag}'}} \cdot n^{c'}) \cdot (2^{-m_{\mathsf{tag}}} \cdot n^{c}) = 2^{m_{\mathsf{tag}'} - m_{\mathsf{tag}}} \cdot n^{c'-c} \leq 2^{n^{\epsilon^{100}}} \cdot n^{c-c'}$.

Putting things together. Note that the joint distribution $(\mathsf{View}_{\mathcal{C}^*}, \mathsf{aux}_{\mathcal{C}^*}, \mathsf{Value}_{\mathcal{C}^*})$ is exactly the distribution $(\mathsf{View}_{\mathsf{Real}}(M), \mathsf{val}_{\mathsf{Real}}(M))$ iff $b = 1$. And, the joint distribution $(\mathsf{View}_{\mathcal{C}^*}, \mathsf{aux}_{\mathcal{C}^*}, \mathsf{Value}_{\mathcal{C}^*})$ is exactly the distribution $(\mathsf{View}_{\mathsf{Ideal}}, \mathsf{val}_{\mathsf{Ideal}})$ iff $b = 0$.

Thus, the PPT distinguisher $\mathcal{D}$ is such that:

$$\Pr[\mathcal{D}(\mathsf{View}_{\mathcal{C}^*}, \mathsf{aux}_{\mathcal{C}^*}, \mathsf{Value}_{\mathcal{C}^*} | b = 1) = 1] - \Pr[\mathcal{D}(\mathsf{View}_{\mathcal{C}^*}, \mathsf{aux}_{\mathcal{C}^*}, \mathsf{Value}_{\mathcal{C}^*} | b = 0) = 1] \geq \frac{1}{\mathsf{poly}(n)}$$

By correctness of extraction, we also have that the joint distribution $(\mathsf{View}', \mathsf{Value}', \mathsf{aux}')$ is indistinguishable (such that all PPT distinguishers have at most $\mathsf{negl}(n)$ distinguishing advantage) from the joint distribution $(\mathsf{View}_{\mathcal{C}^*}, \mathsf{Value}_{\mathcal{C}^*}, \mathsf{aux}_{\mathcal{C}^*})$.

Thus, the PPT distinguisher $\mathcal{D}$ is such that:

$$\Pr[\mathcal{D}(\mathsf{View}', \mathsf{aux}', \mathsf{Value}' | b = 1) = 1] - \Pr[\mathcal{D}(\mathsf{View}', \mathsf{aux}', \mathsf{Value}' | b = 0) = 1] \geq \frac{1}{\mathsf{poly}(n)}$$

Therefore, $\mathcal{R}$ runs in time at most $2^{m_{\mathsf{tag}'}} \cdot n^{c'} \ll 2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n)$ to generate the joint distribution $(\mathsf{View}', \mathsf{aux}', \mathsf{Value}')$, and then echoes the output of $\mathcal{D}$ on input this distribution, we have that:

$$\Pr[\mathcal{R} = 1 | b = 1] - \Pr[\mathcal{R} = 1 | b = 0] \geq \frac{1}{\mathsf{poly}(n)}.$$

This is a contradiction to the fact that no malicious receiver running in time at most $2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n)$ has advantage better than $2^{n^{\epsilon^{100}}} \cdot n^{c-c'}$ in guessing the bit b.

This proves that the joint distribution of the view and value committed by the MIM is indistinguishable in the real and ideal worlds, even in the case when $\mathsf{tag} > \mathsf{tag}'$.

We reiterate that even though we rely on a sub-exponential time reduction, our final simulator is only polynomial time, and specifically, generates the required transcript by honestly committing to 0 corresponding to tag. $\square$

6.3 Bounded-Concurrent Non-malleability for Four Tags

In this section, we describe how to extend the previous scheme to obtain bounded concurrent non-malleability for four tags.

Let $\ell(n)$ be a polynomial that denotes an upper bound on the number of sessions in which the MIM participates as committer. It will suffice to show one-many non-malleability, that is, we will consider the setting where the MIM interacts with the honest committer in only one execution, and generates at most $\ell(n)$ commitments to honest receiver(s). This already implies many-many non-malleability even when the MIM interacts with honest committer(s) in an unbounded polynomial number of executions, and generates at most $\ell(n)$ commitments to honest receiver(s).

We consider the same scheme as Figure 5, except that we set $m_{\mathsf{tag}} = \mathsf{tag} \cdot \ell(n)^{\mathsf{tag}} \cdot n^{\epsilon^{100}}$. For this section, we assume that $\ell(n)^4 \ll n^{\epsilon^{100}}$, but we can change parameters such that $\ell(n)^4$ is small enough. [9]

Theorem 5. *The scheme in Figure 5, with $m_{\mathsf{tag}} = \mathsf{tag} \cdot \ell(n) \cdot n^{\epsilon^{100}}$, is an $\ell(n)$-bounded-concurrent non-malleable commitment scheme according to Definition 6.*

Proof. In this case, the MIM participates in $\ell(n)$ right sessions, and we let $\{\mathsf{tag}'_1, \mathsf{tag}'_2, \ldots \mathsf{tag}'_{\ell(n)}$ denote the set of tags used by the man-in-the-middle in all these right sessions. We also assume, w.l.o.g., that $\mathsf{tag} \not\in \{\mathsf{tag}'_1, \mathsf{tag}'_2, \ldots \mathsf{tag}'_{\ell(n)}\}$.

Then, we let S_{small} denote the subset of right sessions such that $\mathsf{tag}'_i < \mathsf{tag}$ iff $i \in S_{\mathsf{small}}$. And we let S_{big} denote the subset of right sessions such that $\mathsf{tag}'_i > \mathsf{tag}$ iff $i \in S_{\mathsf{big}}$.

Our parameters are carefully aligned such that the commitment scheme for tag is $(2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n), 2^{-m_{\mathsf{tag}}} \cdot n^c)$-hiding against malicious receivers. On the other hand, the commitment schemes for $\mathsf{tag}' \in S_{\mathsf{small}}$ are extractable via an extractor that runs in time at most $2^{m_{\mathsf{tag}'}} \cdot n^{c'}$. And the commitment schemes for $\mathsf{tag}' \in S_{\mathsf{big}}$ can be broken (via brute-force) in time $2^{n^{\epsilon^{\mathsf{tag}'}}}$ to extract the underlying committed message.

Thus, the reduction strategy will be to run the extractors for the extractable commitment schemes in parallel, for sessions where $\mathsf{tag}' \in S_{\mathsf{small}}$. On the other hand, the reduction extracts the committed value value brute-force, from sessions where $\mathsf{tag}' \in S_{\mathsf{big}}$. Note that the reduction must extract the *joint distribution* of values committed by the MIM together with the joint view, while trying to contradict hiding of $\mathsf{com}_{\mathsf{tag}}$ using an MIM that succesfully carries out a malleation attack.

Unfortunately the extraction strategy for the extractable commitments, when executed on one committer generating a single commitment with respect to some tag, outputs *some* commitment transcript together with the underlying committed value. Thus, executing the extraction strategy separately on all right sessions where $\mathsf{tag}' \in S_{\mathsf{small}}$ is not guaranteed to extract the joint distribution of values committed by the MIM (and may potentially be extracting values for different tag' from different executions).

However, it is easy to observe that the extraction strategy in Section 5 can be extended, to run in time $2^{K(n)}$, where $K(n) = \Sigma_{i \in S_{\mathsf{small}}} m_{\mathsf{tag}'_i} \leq \ell(n) m_{\mathsf{tag}-1}$, to *simultaneously extract* the values committed in all the MIM's right sessions corresponding to $\mathsf{tag}' \in S_{\mathsf{small}}$. The extended extractor simply waits for a situation where all of the MIM's commitment transcripts for $\mathsf{tag}' \in S_{\mathsf{small}}$ get extracted together. The reduction then uses this extended extraction strategy to jointly extract

[9] Our later tag amplification procedures require roughly $O(\log^* n)$ levels of hardness above 2^m, and therefore impose a stricter bound on $\ell(n)$. In general, we can handle any $\ell(n) \leq O(n^{\epsilon^T})$, where T are the number of levels of tag amplification. In order to achieve non-malleability for all tags in $[2^n]$, we would end up requiring $T = \log^* n$ (see Section 6.4), thus we require $\ell(n) \leq O(n^{\epsilon^{(\log^* n)}})$. On the other hand, for any constant number of tags, we can handle any a-priori fixed polynomial $\ell(n)$, by suitably increasing other parameters in the scheme.

all values from sessions where $\mathsf{tag}' \in S_{\mathsf{small}}$ in time at most $2^{\ell(n)\cdot m_{\mathsf{tag}-1}} \ll 2^{m_{\mathsf{tag}}}$. It simultaneously extracts the joint distribution of values committed by the MIM in sessions where $\mathsf{tag}' \in S_{\mathsf{big}}$, thereby using a successful MIM to contradict $(2^{n^{\epsilon^{\mathsf{tag}+1}}} \cdot \mathsf{poly}(n), 2^{-m_{\mathsf{tag}}} \cdot n^c)$-hiding of the commitment scheme. □

Remark 5. *We note that the resulting bounded-concurrent scheme can easily be made non-malleable against adversaries running in time $\widetilde{T}$, where $2^{m_{\mathsf{tag}}} \ll \widetilde{T} \ll T_{\mathsf{hid}}$ for all m_{tag}, by setting parameters so that $\widetilde{T} \cdot 2^{m_{\mathsf{tag}}} \ll T_{\mathsf{hid}}$, where T_{hid} refers to the hiding parameters of commitments.*

6.4 Round-Preserving Tag Amplification

In this section, we present a round-preserving amplification technique that helps bootstrap any $\ell(n)$-bounded-concurrent non-malleable commitment scheme for 4 tags into an $\ell(n)$-bounded-concurrent non-malleable commitments for all tags/identitites in $[2^n]$.

We now describe a compiler from a two-round non-malleable commitment scheme denoted by $\mathsf{com}_{1,\mathsf{tag}}, \mathsf{com}_{2,\mathsf{tag}}(m;r)$ for tags in $[t]$, into a non-malleable commitment scheme for tags in $[\binom{t}{t/2}]$.

We assume that the input two-round non-malleable commitment scheme $\mathsf{com}_{\mathsf{tag}}(m;r)$ for tags in $[t]$ can be broken (via brute-force) in time at most T (In other words, $T = 2^n$ where n is the maximum security parameter out of the security parameters of *all components* of the non-malleable commitment for $\mathsf{tag} \in [t]$.) We also assume the existence of two-message SPSS ZK for delayed-input statements, such that $T_{\mathsf{zk}} \gg T \gg T_{\mathsf{sim}}$. Finally, we require the underlying two-round non-malleable commitment scheme $\mathsf{com}_{1,\mathsf{tag}}, \mathsf{com}_{2,\mathsf{tag}}(m;r)$ for tags in $[t]$ to be non-malleable against adversaries running in time T_{sim}. In particular, this also means that the ZK arguments used in the input two-round non-malleable commitment scheme $\mathsf{com}_{1,\mathsf{tag}}, \mathsf{com}_{2,\mathsf{tag}}(m;r)$ are sound against adversaries running in time T_{sim}.

Then the compiler in Figure 6 gives a two round scheme that is $\ell(n)$-bounded-concurrent non-malleable for tags in $[\mathcal{T}]$, where $\mathcal{T} = \binom{t}{t/2}$. This compiler can be applied iteratively $(\log^* n)$ times, starting with a scheme for 4 tags, to obtain a scheme for $\mathsf{tag} \in [2^n]$. The resulting scheme can easily be made to have polynomial running time and polynomial communication complexity.

Claim 1. *The protocol in Figure 6 is a statistically binding, computationally hiding commitment.*

Proof. Statistical binding of the protocol in Figure 6 follows directly from the statistical binding property of the underlying commitment scheme. Computational hiding follows by the hiding of the underlying commitments, and the SPS ZK property of Π, via a sequence of hybrids.

In order to prove computational hiding, we consider the following series of hybrid experiments: Hybrid_0 : The output of this hybrid is the receiver's view when the committer sends a commitment to message M.

Hybrid_1 : The output of this hybrid is the receiver's view when the challenger sends a commitment to message M the same way as Hybrid_0, except that it sends a simulated SPSS ZK argument. The view is indistinguishable from Hybrid_0 by the simulation security of the SPSS ZK argument.

Hybrid_2 : In this hybrid, the challenger proceeds the same way as Hybrid_1 except that it generates $c_1 = \mathsf{com}_{2,s_1}(0;r_1)$. The view is indistinguishable from Hybrid_1 by the hiding of the underlying commitment com_{s_1}. The challenger proceeds the same way across $\mathsf{Hybrid}_3, \ldots \mathsf{Hybrid}_{t/2+1}$, replacing $c_i = com_{2,s_i}(0;r_i)$ for $i \in [t/2]$.

Language L: We define $L = \{\{c_i, \mathsf{com}_{s_i}\}_{i\in[t/2]} : \exists M, r_i : c_i = \mathsf{com}_{s_i}(M; r_i)\}$.

Committer Input: Message $M \in \{0,1\}^p$, tag $\mathsf{tag} \in [1, T]$, where $T = \binom{t}{t/2}$.

Receiver Input: Tag tag.

Commit Stage:

1. Let $\mathbb{T}$ denote the ordered set of all possible subsets of $[t]$, of size $t/2$. Pick the i^{th} element in set $\mathbb{T}$, for $i = \mathsf{tag}$. Let this element be denoted by $(s_1, \ldots s_{t/2})$.

2. **Receiver Message.** Send Π_1 as the first message of Π for language L, and $\mathsf{com}_{1,s}$ for $s \in [s_1, s_2, \ldots s_{t/2}]$ as the first messages of the non-malleable commitment scheme for small tags.

3. **Committer Message.** For $i \in [t/2]$, sample randomness $r_i \xleftarrow{\$} \{0,1\}^*$ and send $c_i = \mathsf{com}_{2,s_i}(M; r_i)$ to $\mathcal{R}$. Also, send Π_2 proving that:
$$\{c_i, \mathsf{com}_{s_i}\}_{i\in[t/2]} \in L$$

4. The receiver accepts the commitment if Π verifies and all $t/2$ commitments are accepting.

Reveal Stage: The committer reveals randomness $r_1, r_2, \ldots r_{t/2}$ to the receiver. The receiver verifies that all the commitments were correctly decommitted.

Figure 6: Round-Preserving Tag Amplification

$\mathsf{Hybrid}_{t/2+1}$: In this hybrid, the challenger generates $c_i = \mathsf{com}_{2,s_i}(0; r_i)$ for $i \in [t/2]$, while simulating the SPSS ZK proof. The view is indistinguishable from Hybrid_2 by the hiding of the underlying commitments.

$\mathsf{Hybrid}_{t/2+2}$: In this hybrid, the challenger generates $c_i = \mathsf{com}_{2,s_i}(0; r_i)$ for $i \in [t/2]$ and then general the SPSS ZK argument honestly. The view is indistinguishable from $\mathsf{Hybrid}_{t/2+1}$ by the simulation security of the SPSS ZK argument.
This hybrid also represents an honestly generated commitment to 0, thus, we have that a commitment to m is computationally indistinguishable from a commitment to 0. □

We have the following main theorem for tag amplification.

Theorem 6. *Assuming the existence of (sub-exponentially secure) two-round SPSS ZK for delayed-input statements, there exists a compiler that compiles a (sub-exponentially secure) bounded-concurrent non-malleable commitment scheme for* $\mathsf{tag} \in [4]$*, into a bounded-concurrent non-malleable commitment scheme for* $\mathsf{tag} \in [2^n]$*.*

This theorem is implied by the following lemma for tag amplification, Lemma 5, which proves that the compiler obtains a bounded-concurrent non-malleable commitment scheme for $\mathsf{tag} \in [\binom{t}{t/2}]$ on input a bounded-concurrent non-malleable commitment scheme for $\mathsf{tag} \in [t]$. The smallest tag t such that $T = \binom{t}{t/2} > t$, is $t = 4$, where $T = 6$.

Thus, starting at $t = 4$, we repeatedly use the protocol in Figure 6 to amplify tags, each time choosing a large enough security parameter for the outer SPSS ZK proof.

This parameter is chosen, such that $T_{\mathsf{zk}} \gg T_{\mathsf{break-com}}$, where $T_{\mathsf{break-com}}$ is the time required to break (via brute force) all internal commitments (via brute-force). Furthermore, since $T_L \gg T_{\mathsf{zk}} \gg T_{\mathsf{break-com}}$, we have that the language can be decided in time at most T_L, thus soundness holds.

We will also need that the underlying non-malleable commitment scheme is secure against adversaries running in time T_{Sim} (this can be achieved by leveraging the inner commitment scheme).

Finally, $T_{\mathsf{soundness'}} \gg T_{\mathsf{Sim}}$ where $T_{\mathsf{soundness'}}$ is such that all inner proofs (parts of all internal commitments) are sound against provers running in time $T_{\mathsf{soundness'}}$, and T_{Sim} is the running time of the simulator Sim. This means that even when an outer proof is simulated, it is possible to ensure that all inner proofs still remain sound.

Applying this compiler repeatedly requires $O(\log^* n)$ iterations, and results in a protocol where the committer and receiver run in time at most $\mathsf{poly}(n)$ in order to generate non-malleable commitments for $\mathsf{tag} \in [2^n]$. Note that sub-exponential assumptions on the SPSS ZK allow us to obtain $O(\log^* n)$ levels of complexity leveraging [PW], by setting the parameters for the 4-tag scheme at $\log^2 N$, where N denotes the overall security parameter. We note that this does not interfere in any way with our hardness assumptions for the 4-tag scheme, as we are only scaling down all parameters of that scheme simultaneously.

Lemma 5. *Assuming* com *is* $\ell(n)$*-concurrent non-malleable for tags in* $[t]$*, the scheme in Figure 6 is such that for every* $\ell(n)$*-concurrent PPT* MIM*, and for every* $\mathsf{tag}, \mathsf{tag}' \in [T]$*, where* $T = \binom{t}{t/2}$ $\mathsf{tag}' \neq \mathsf{tag}$*, there exists a PPT simulator* $\mathcal{S}$ *such that the following ensembles are computationally indistinguishable:*

$$\{\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)\}_{n\in\mathbb{N}, v\in\{0,1\}^n, z\in\{0,1\}^*} \text{ and } \{\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)\}_{n\in\mathbb{N}, v\in\{0,1\}^n, z\in\{0,1\}^*}$$

Proof. Suppose the MIM participates in $\mathcal{L}(n) \leq \ell(n)$ executions on the right (that is, with honest receiver(s)). The simulator $\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)$ generates $\{\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)\}_{n\in\mathbb{N}, v\in\{0,1\}^n, z\in\{0,1\}^*}$ by picking $r \xleftarrow{\$} \{0,1\}^*$ and generating $\mathsf{com}(0, r)$ with tag tag on the left, and outputs the transcript generated, and the view of the MIM on the right. Let $\mathsf{tag}'_1, \mathsf{tag}'_2, \ldots \mathsf{tag}'_{\mathcal{L}(n)}$ denote the tags used by the MIM.

We will now prove that the joint distribution of the view and values committed by the MIM is indistinguishable between the real and simulated executions. We consider a sequence of hybrid experiments, starting with the real execution and proceeding towards the simulated execution, proving that the joint distribution of the view and values committed by the MIM is indistinguishable between consecutive hybrids.

Before proceeding with the sequence of hybrids, we note that if $\mathsf{tag} \in \{\mathsf{tag}'_1, \mathsf{tag}'_2, \ldots \mathsf{tag}'_{\mathcal{L}(n)}\}$, the experiment aborts – and we must only prove non-malleability when $\mathsf{tag} \notin \{\mathsf{tag}'_1, \mathsf{tag}'_2, \ldots \mathsf{tag}'_{\mathcal{L}(n)}\}$.

Then for every $j \in \mathcal{L}(n)$, there exists at least one index $i'_j \in [t/2]$, such that small tag $s'_{i'_j,j} \notin \{s_1, s_2, \ldots s_{t/2}\}$, where $s'_{i'_j,j}$ denotes the i'^{th} small tag in the j^{th} session on the right, and s_i denotes the i^{th} small tag on the left. Looking ahead, we will focus on the set of right (MIM) commitments indexed by $\{i'_j, j^{th}\}_{j\in\mathcal{L}(n)}$, and extract the joint value committed in these sub-commitments, while simulating the left (honest) commitment.

By the soundness of the proof Π, in at least $1 - \mathsf{negl}(n)$ of all accepting right commitment transcripts, in both $\{\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)\}_{n\in\mathbb{N}, v\in\{0,1\}^n, z\in\{0,1\}^*}$ and $\{\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)\}_{n\in\mathbb{N}, v\in\{0,1\}^n, z\in\{0,1\}^*}$, the joint distribution of values $\{\mathsf{value}_j\}_{j\in\mathcal{L}(n)}$ committed by the MIM is identical to the joint distribution of values $\{\mathsf{value}_{i'_j,j}\}_{j\in\mathcal{L}(n)}$ where $\mathsf{value}_{i'_j,j}$ is committed in the j^{th} session using the sub-commitment $\mathsf{com}_{s'_{i'_j},j}$ for the index $i'_j \in [t/2]$, such that small tag $s'_{i'_j} \notin \{s_1, s_2, \ldots s_{t/2}\}$. Therefore, it will suffice to show that the joint distribution of the view of the MIM and the values comitted using $\mathsf{com}_{s'_{i'_j},j}$ for $j \in \mathcal{L}(n)$ is indistinguishable between the real and simulated worlds. We use the random variable $\{\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)\}_{i'_j}$ to denote this joint distribution in the real world, and $\{\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)\}_{i'_j}$ to denote this joint distribution in the ideal world.

We now formally describe the hybrid experiments:

Hybrid_0 : The output of this experiment is the distribution $\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_0}$ of the view and values committed by the MIM using the ${i'_j}^{th}$ commitments for $j \in \mathcal{L}(n)$, when the committer commits to value in the real world.

Hybrid_1 : In this experiment, the challenger generates a left commitment to value in the same way as Hybrid_0, except that it starts simulating the proof Π. Let $\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_1}$ denote the joint distribution of the view and the values committed using the ${i'_j}^{th}$ commitments for $j \in \mathcal{L}(n)$, by the MIM, in this hybrid.

We consider a reduction R against the simulation security of the proof Π against T_{zk}-time adversaries: $\mathcal{R}$ runs the simulator in time T_{sim} and externally obtains either a real proof or a simulated proof, and then obtains the right transcript of the MIM. Next, it breaks (via brute force) in time at most $T \cdot \ell(n)$ the commitments $\mathsf{com}_{s'_{i'_j},j}$ and extracts the $\mathsf{value}_{i'_j}$ for $j \in \mathcal{L}(n)$. Then, if there exists a PPT distinguisher $\mathcal{D}$ such that: $\big|\Pr[\mathcal{D}(\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_1}) = 1] - \Pr[\mathcal{D}(\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_0}) = 1]\big| \geq \frac{1}{\mathsf{poly}(n)}$ $\mathcal{R}$ can run this PPT distinguisher on joint distribution of the transcript generated, together with the extracted values $\mathsf{value}_{i'_j}$, and echo the output in order to distinguish the real from the simulated proof in time $T \cdot \ell(n) \ll T_{\mathsf{zk}}$, which is a contradiction.

$\mathsf{Hybrid}_{1,1}$: In this experiment, the challenger behaves the same was as Hybrid_1, except that it generates $\mathsf{com}_{s_1}(0; r)$ for $r \xleftarrow{\$} \{0,1\}^*$. Let $\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_2}$ denote the joint distribution of the view and the values $\mathsf{value}_{i'_j}$ committed using the ${i'_j}^{th}$ commitments, by the MIM, in this hybrid.

We consider a reduction R against the $\ell(n)$-concurrent non-malleability of com for small tags against T_{Sim}-time adversaries. $\mathcal{R}$ obtains $\mathsf{com}_{s_1}(0; r)$ externally (while still simulating the SPS ZK argument in time T_{Sim}). Since $s'_{i'_j} \neq s_1$ and $\mathcal{L}(n) \leq \ell(n)$, it obtains the joint distribution of the view and value $\mathsf{value}_{i'_j}$ for $j \in \mathcal{L}(n)$. Then, if there exists a PPT distinguisher $\mathcal{D}$ such that:

$$\big|\Pr[\mathcal{D}(\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_2}) = 1] - \Pr[\mathcal{D}(\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_1}) = 1]\big| \geq \frac{1}{\mathsf{poly}(n)}$$

$\mathcal{R}$ can run this PPT distinguisher on the transcript generated, together with the $\mathsf{value}_{i'_j}$, and echo the output in order to break non-malleability of com for small tags in time T_{sim}, which is a contradiction.

Similarly, we have the following sequence of hybrids for $\tilde{i} \in [2, t/2]$:

$\mathsf{Hybrid}_{1,\tilde{i}}$: In this experiment, the challenger behaves the same was as Hybrid_1, except that it generates $\mathsf{com}_{s_{\tilde{j}}}(0; r)$ for $r \xleftarrow{\$} \{0,1\}^*$ and $\tilde{j} \in [\tilde{i}]$. Let $\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_{1,\tilde{i}}}$ denote the joint distribution of the view and the values $\mathsf{value}_{i'_j}$ committed using the ${i'_j}^{th}$ commitment, by the MIM, in this hybrid.

We consider a reduction R against the $\ell(n)$-concurrent non-malleability of com for small tags against T_{Sim}-time adversaries. $\mathcal{R}$ obtains $\mathsf{com}_{s_{\tilde{i}}}(0; r)$ externally (while still simulating the SPSS ZK proof in time T_{Sim}). Since $s'_{i'_j} \neq s_{\tilde{i}}$ and $\mathcal{L}(n) \leq \ell(n)$, $\mathcal{R}$ obtains the joint distribution of the view and values $\mathsf{value}_{i'_j}$ for $j \in \mathcal{L}(n)$. Then, if there exists a PPT distinguisher $\mathcal{D}$ such that:

$$\left|\Pr[\mathcal{D}(\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_{1,\tilde{i}}}) = 1] - \Pr[\mathcal{D}(\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_{1,\tilde{i}-1}}) = 1]\right| \geq \frac{1}{\mathsf{poly}(n)}$$

$\mathcal{R}$ can run this PPT distinguisher on the transcript generated, together with the values $\mathsf{value}_{i'_j}$ for $j \in \mathcal{L}(n)$, and echo the output in order to break non-malleability of com for small tags in time T_{Sim}, which is a contradiction.

Hybrid_2 : In this experiment, the challenger generates a left commitment to 0 in the same way as $\mathsf{Hybrid}_{1,t/2}$, except that it generates the proof Π honestly. Note that this is possible because in this hybrid, all left commitments with small tags are valid commitments to 0. Let $\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)_{i'_j}$ denote the joint distribution of the view and the values committed using the ${i'_j}^{th}$ small commitments for $j \in \mathcal{L}(n)$, by the MIM, in this hybrid.

We consider a reduction R against the simulation security of the proof Π against T_{zk} adversaries: it externally obtains either a real proof or a simulated proof, and then obtains the right transcript of the MIM. Next, it breaks (via brute force) the commitments $\mathsf{com}_{s'_{i'_j}}$ and extracts the values $\mathsf{value}_{i'_j}$ where $\mathsf{value}_{i'_j}$ denotes the i'^{th} value committed in the j^{th} session on the right – this takes time at most $T \cdot \ell(n)$. Then, if there exists a PPT distinguisher $\mathcal{D}$ such that: $\left|\Pr[\mathcal{D}(\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)_{i'_j}) = 1] - \Pr[\mathcal{D}(\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)_{i'_j,\mathsf{Hybrid}_{1,t/2}}) = 1]\right| \geq \frac{1}{\mathsf{poly}(n)}$, $\mathcal{R}$ can run this PPT distinguisher on the transcript generated, together with the values $\mathsf{value}_{i'_j}$, and echo the output in order to distinguish the real from the simulated proof in time $T \cdot \ell(n) \ll T_{zk}$, which is a contradiction.

Thus, we have that for any PPT distinguisher $\mathcal{D}$,

$$\left|\Pr[\mathcal{D}(\{\mathsf{MIM}_{\langle C,R\rangle}(\mathsf{value}, z)\}) = 1] - \Pr[\mathcal{D}(\{\mathsf{Sim}_{\langle C,R\rangle}(1^n, z)\} = 1]\right| \leq \mathsf{negl}(n)$$

This completes the proof of the lemma. □

Remark 6. *We note that if the initial scheme (for* 4 *tags) was non-malleable against adversaries running in time* $\widetilde{T} \gg 2^m$ *for* $m = \mathsf{maximum}\{m_{\mathsf{tag}}\}_{\mathsf{tag} \in [4]}$*, then the resulting scheme (after applying the compiler) remains non-malleable against adversaries running in time* $\widetilde{T}$ *if* $\widetilde{T} \ll T_{\Pi}$*, where* T_{Π} *denotes the soundness parameter (which is also the weakest parameter) of the SPSS ZK. We also note that the resulting commitment scheme (after applying this compiler), continues to have good extraction properties, that is,* $T_{\mathsf{hid}} \gg T_{\mathsf{Ext}}$*.*

Obtaining $\mathsf{poly}(n)$ Communication and Computation Complexity of the Resulting Scheme for All Tags. The final scheme results after $O(\log^* n)$ applications of the above scheme, and the computation and communication complexity grows at every iteration.

The complexity of the commitment comp_i at iteration i, where tags go from $n_{i-1} \rightarrow n_i = \binom{n_{i-1}}{n_{i-1}/2}$, can be written as a function of the complexity comp_{i-1} at iteration $(i-1)$ as follows:

$\mathsf{comp}_i = n_{i-1} \cdot \mathsf{comp}_{i-1} + \mathsf{comp}_{\mathsf{zk}}$, where $\mathsf{comp}_{\mathsf{zk}}$ denotes the complexity of the zero-knowledge argument. Unfortunately, the statement being proved by the ZK argument has complexity $\mathsf{poly}(n, \mathsf{comp}_{i-1})$, (where n is the security parameter) thus if the protocol is executed trivially, the complexity becomes exponential in $O(\log^* n)$ iterations.

In order to fix this, we modify the ZK argument so that: instead of proving that all commitments com at stage $(i-1)$ commit to the same value, we only prove that *one* of the sub-commitments (that is, a basic commitment for $\mathsf{tag} \in [4]$ which is the leaf node) within each commitment at stage $(i-1)$ commits to the same value. The resulting modified statement has complexity only $\mathsf{poly}(n) \cdot \mathsf{comp}_0 = \mathsf{poly}(n)$, thus we have that $\mathsf{comp}_i = n_{i-1} \cdot \mathsf{comp}_{i-1} + \mathsf{poly}(n)$. We note that this expression converges such that the complexity of the resulting protocol after $O(\log^* n)$ iterations is at most $\mathsf{comp}_{\log^* n} = \mathsf{poly}(n)$.

7 One Round Non-Malleable Commitments w.r.t Opening

In this section, we construct one-round concurrent non-malleable commitments with respect to opening, in the simultaneous message model. Our main observation is that the commitment part (com) of the $\mathsf{NM} - \mathsf{Com}$ constructed in Section 6, doesn't need to depend on the receiver's message, and can therefore be sent by the committer simultaneously with the receiver's message in the first round. The remaining part of the commitment message is sent in the second round. This results in a scheme, which requires one round of simultaneous exchange followed by another round in which only the committer sends a message. The resulting scheme is statistically binding by the end of the first round. We will begin by proving non-malleability of the resulting scheme, in the following section.

After that, we will describe how to use SPSS ZK together and some additional complexity leveraging to obtain a non-malleable commitment scheme that has a single round of simultaneous exchange in the commitment phase, and then a single message in the opening phase. Very roughly, this will be achieved by pushing the second round of the non-malleable with respect to commitment scheme, into the opening round, while preserving non-malleability.

7.1 Reordering Two-Round Non-Malleable Commitments w.r.t. Commitment

We begin by reordering the two-round bounded-concurrent non-malleable commitment scheme from Section 6, into a *two-round* commitment scheme the simultaneous message model, where the first message of the scheme is statistically binding. We prove that the resulting scheme also satisfies *concurrent non-malleability with respect to commitment.*

This is achieved by simply reordering the messages in the commitment schemes constructed in Section 6.

Reordering the basic scheme In Figure 7, we describe how to reorder the basic scheme for 4 tags from Section 6.2.

Recall that the commitment scheme for 4 tags consists of either a non-interactive commitment com, or an extractable commitment. The extractable commitment (refer Section 5.1.1) itself consists of receiver message $\mathsf{2PC}_\mathsf{R}(1^n, \mathsf{ch})$ together with the first message Π_1 of SPSS ZK. This message is used for setting up the trapdoor that allows an extractor to extract the committed value. We denote this message by $\mathsf{td}_R = (\mathsf{2PC}_\mathsf{R}(1^n, \mathsf{ch}), \Pi_1)$. The sender message consists of a non-interactive commitment com generated by the committer, together with a committer message $\mathsf{td}_S = (\tau_1 = \mathsf{2PC}_\mathsf{R}(1^n, \mathsf{ch}), \Pi_1)$. We split the scheme for 4 tags into two parts, one part consisting of the commit message com,

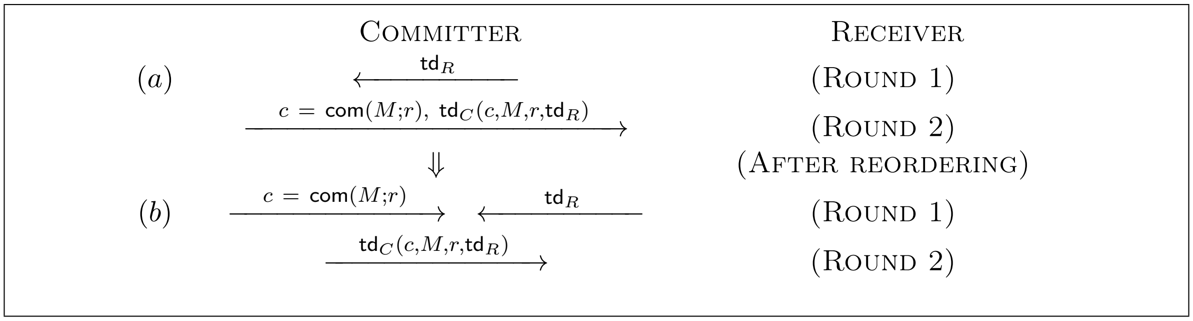


Figure 7: Reordering the scheme for four tags

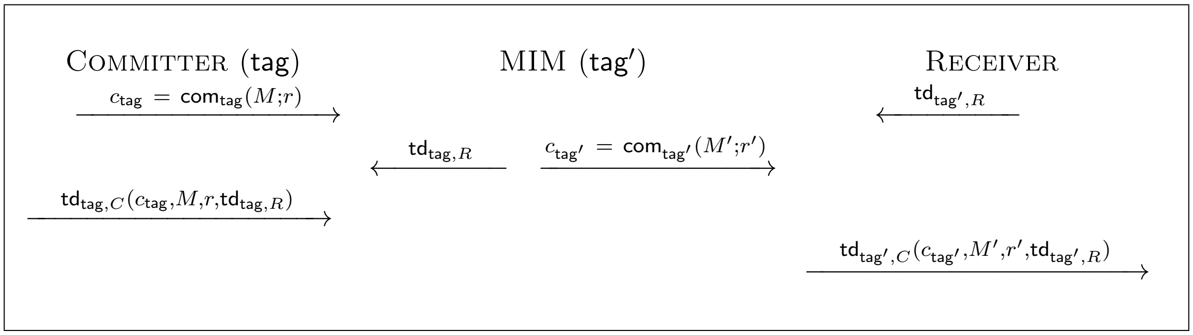

Committer (tag)
MIM (tag′)
Receiver
$c_{\mathsf{tag}} = \mathsf{com}_{\mathsf{tag}}(M;r)$
$\mathsf{td}_{\mathsf{tag}',R}$
$\mathsf{td}_{\mathsf{tag},R}$
$c_{\mathsf{tag}'} = \mathsf{com}_{\mathsf{tag}'}(M';r')$
$\mathsf{td}_{\mathsf{tag},C}(c_{\mathsf{tag}},M,r,\mathsf{td}_{\mathsf{tag},R})$
$\mathsf{td}_{\mathsf{tag}',C}(c_{\mathsf{tag}'},M',r',\mathsf{td}_{\mathsf{tag}',R})$

Figure 8: Message scheduling for a MIM adversary in the scheme 7(b) for four tags

and the second part consisting of the trapdoor that allows for extraction. This is denoted by $\mathsf{td}_R = (\mathsf{2PC}_S(\tau_1, x = (1, M, \widetilde{r}, r')), \Pi_2)$, where $\mathsf{2PC}$ is performed for the functionality $\mathcal{F}$. Given this, the scheme in Figure 7 (a) depicts the commitment schemes for 4 tags from Section 6.

In Figure 8, we describe the only non-trivial message scheduling for a (rushing) man-in-the-middle adversary participating in an execution of the protocol, that will be relevant for our main result on non-malleable commitments with respect to opening. This scheduling considers an adversary that participates in the protocol in rounds, such that it obtains all honest messages for a particular round before generating its own message. While we only illustrate the one-one setting, this can be directly extended to the one-many setting.

We will now prove that the resulting scheme retains (bounded-concurrent) non-malleability w.r.t. commitment, against rushing adversaries in the simultaneous message model, even when messages are reordered according to Figure 7 (b).

Lemma 6. *The one-one (resp. bounded concurrent) non-malleable commitment scheme in Figure 5, with messages reordered according to Figure 7, remains one-one (resp. bounded concurrent) non-malleable against a man-in-the-middle that schedules messages according to Figure 8.*

Proof. In the sessions where $\mathsf{tag} < \mathsf{tag}'$, the proof of non-malleability of the scheme in Figure 5 relied on complexity leveraging such that the MIM's commitment com, can be broken (via brute force) in time less than T_{hid}, whereas the commitment scheme for tag is hiding against adversaries running in time T_{hid}. When messages are reordered according to Figure 7, exactly the same proof as Theorem 4 and Theorem 5, Case I goes through to show that the scheme is (bounded-concurrent) secure against an MIM that schedules messages according to Figure 8.

In the sessions where $\mathsf{tag} > \mathsf{tag}'$, the proof of non-malleability of the scheme in Figure 5 relies on using the extractability of all the commitments for $\mathsf{tag}' < \mathsf{tag}$ in time roughly $2^{m'_{\mathsf{tag}}}$ using roughly $2^{m'_{\mathsf{tag}}}$ queries: while relying on the fact that the commitment for tag is more than $(2^{m'_{\mathsf{tag}}}, 2^{-m'_{\mathsf{tag}}})$-hiding. Because of SPSS ZK, the proof of hiding of the commitment scheme for tag uses only uniform simulation (with a simulator that runs in low super-polynomial time). This means that re-ordering the honest messages so that c_{tag} is sent *before* $\mathsf{td}_{\mathsf{tag},R}$ is generated does not affect hiding of the commitment scheme. Again, essentially the same proof of non-malleability as Theorem 4 and Theorem 5, Case II goes through to show that the scheme is secure against an MIM that schedules messages according to Figure 8. This completes the proof of the lemma. □

Tag Amplification for the Reordered Scheme. Our tag amplification protocol remains identical to the tag amplification procedure for two-message non-malleable commitments (Figure 6), except that the underlying commitment for small tags is now replaced with a reordered commitment for small tags.

We consider an identical tag amplification process as Section 6.4, such that the committer and receiver execute multiple parallel (reordered) commitments for different small tags according to the tag encoding scheme of Section 6.4. In parallel, the committer and receiver execute a two-message SPSS ZK argument that all commitments for small tags, commit to the same value. The first message of the SPSS ZK argument is sent by the receiver in the first round, and the second message is sent by the committer in the second round. For completeness, the protocol is described in Figure 9.

We compile from a two-round reordered non-malleable commitment scheme. This scheme will be denoted by $\mathsf{com}_{1,C,\mathsf{tag}}(M;r), \mathsf{com}_{1,R,\mathsf{tag}}, \mathsf{com}_{2,\mathsf{tag}}$ for input M, randomness r and tags in $[t]$. We obtain a reordered non-malleable commitment scheme for tags in $[\binom{t}{t/2}]$. We assume that the input non-malleable commitment scheme $\mathsf{com}_{\mathsf{tag}}(m;r)$ for tags in $[t]$ can be broken (via brute-force) in time at most T (In other words, $T = 2^n$ where n is the maximum security parameter out of the security parameters of *all components* of the non-malleable commitment for $\mathsf{tag} \in [t]$.) We also assume the existence of two-message SPSS ZK for delayed-input statements, such that $T_{\mathsf{zk}} \gg T \gg T_{\mathsf{sim}}$. Finally, we require the underlying non-malleable commitment scheme $\mathsf{com}_{1,\mathsf{tag}}, \mathsf{com}_{2,\mathsf{tag}}(m;r)$ for tags in $[t]$ to be $\ell(n)$-concurrent non-malleable against adversaries running in time T_{sim}. In particular, this also means that the ZK arguments used in the input non-malleable commitment scheme $\mathsf{com}_{1,\mathsf{tag}}, \mathsf{com}_{2,\mathsf{tag}}(m;r)$ are sound against adversaries running in time T_{sim}.

Then the compiler in Figure 6 gives a two round scheme that is $\ell(n)$-concurrent non-malleable for tags in $[\mathcal{T}]$, where $\mathcal{T} = \binom{t}{t/2}$. Just like Section 6.4, the compiler can be applied iteratively $O(\log^* n)$ times, starting with a scheme for 4 tags, to obtain a scheme for $\mathsf{tag} \in [2^n]$. The resulting scheme can easily be made to have polynomial running time and polynomial communication complexity (via a slight modification of the statement for the ZK argument, as already described in Section 6.4).

The statistical binding (by the end of the first round) and computational hiding properties of the commitment scheme are obvious by inspection. Because we only rely on uniform simulation for proof of Theorem 6, the same proof as Section 6.4 goes through. Thus, we have the following theorem for tag amplification.

Theorem 7. *Assuming the existence of (sub-exponentially secure) two-round SPSS ZK for delayed-input statements, there exists a compiler that compiles a (sub-exponentially secure) bounded-concurrent non-malleable commitment scheme for* $\mathsf{tag} \in [4]$*, into a bounded-concurrent non-malleable commitment scheme for* $\mathsf{tag} \in [2^n]$.

Language L: We define $L = \{\{c_i, \mathsf{com}_{s_i}\}_{i \in [t/2]} : \exists M, r_i : c_i = \mathsf{com}_{s_i}(M; r_i)\}$.

Committer Input: Message $M \in \{0,1\}^p$, tag $\mathsf{tag} \in [1, T]$, where $T = \binom{t}{t/2}$.

Receiver Input: Tag tag.

Commit Stage:

1. Let $\mathbb{T}$ denote the ordered set of all possible subsets of $[t]$, of size $t/2$. Pick the i^{th} element in set $\mathbb{T}$, for $i = \mathsf{tag}$. Let this element be denoted by $(s_1, \ldots s_{t/2})$.

2. **First Round.**
   Committer Message. For $i \in [t/2]$, sample randomness $r_i \overset{\$}{\leftarrow} \{0,1\}^*$ and send $c_{1,C,i} = \mathsf{com}_{1,C,s_i}(M; r_i)$ to $\mathcal{R}$.

   Receiver Message. Send Π_1 as the first message of Π for language L, and com_{1,R,s_i} for $i \in [t/2]$ as the first messages of the non-malleable commitment for small tags.

3. **Second Round: Committer Message.** For $i \in [t/2]$, send $c_{2,i} = \mathsf{com}_{2,s_i}(M; r_i)$ to $\mathcal{R}$. Send Π_2 proving that:
$$\{c_i, \mathsf{com}_{s_i}\}_{i \in [t/2]} \in L$$

4. The receiver accepts the commitment if Π verifies and all $t/2$ commitments are accepting.

Reveal Stage: The committer reveals randomness $r_1, r_2, \ldots r_{t/2}$ to the receiver. The receiver verifies that all the commitments were correctly decommitted.

Figure 9: Round-Preserving Tag Amplification

We note that for all our schemes, security parameters can be suitably increased so that the resulting scheme is non-malleable against sub-exponential time adversaries, subject to the total number of sub-exponential levels remaining bounded by $O(\log n / \log\log n)$.

7.2 One-Round Non-Malleable Commitments with Simultaneous Messages

In this section, we prove the following main theorem.

Theorem 8. *Given a two-round commitment scheme in the simultaneous exchange model that is $(\ell(n))$ non-malleable with respect to commitment against subexponential man-in-the-middle adversaries, there exists a one-round non-malleable commitment scheme in the simultaneous exchange model that is $(\ell(n))$-concurrent non-malleable with respect to opening, according to Definition 9, against all PPT man-in-the-middle adversaries.*

Proof. In Figure 10, we describe a compiler that given any two-round non-malleable commitment scheme w.r.t. commitment, $\mathsf{NM-Com}$, in the simultaneous exchange model, compiles it using SPSS-ZK into a one-round non-malleable commitment scheme w.r.t. opening.

Let $\Pi = (\Pi_1, \Pi_2)$ denote a two message SPSS ZK argument with a T_{Sim}-time simulator, that is zero-knowledge against T_{zk}-time adversaries. Let $\mathsf{com} = \mathsf{com}_{1,C}(M; r), \mathsf{com}_{1,R}(\cdot), \mathsf{com}_{2,C}(M; r)$ denote a two-message $\ell(n)$-concurrent non-malleable commitment scheme with respect to commitment in the simultaneous exchange model, against subexponential man-in-the-middle adversaries running in time T_{nm}, such that all parameters in the non-malleable commitment are breakable in time T_{com}. The parameters are leveraged such that $T_{\mathsf{Sim}} << T_{\mathsf{nm}}$, and $T_{\mathsf{zk}} >> T_{\mathsf{com}}$.

Language L: We define $L = \{\{c, \mathsf{com}, M\} : \exists r : c = \mathsf{com}(M; r)\}$.
Committer Input: Message $M \in \{0, 1\}^p$, tag $\mathsf{tag} \in [2^n]$.
Receiver Input: Tag tag.

Commit Stage:
In one simultaneous exchange round, the committer and receiver send the following messages:

- **Committer Message.** Send $\mathsf{com}_{1,C}(M; r)$ to the receiver.
- **Receiver Message.** Send $\mathsf{com}_{1,R}$ to the committer, together with Π_1.

Reveal Stage:

- The committer sends $\mathsf{com}_{2,C}(M; r)$ to the receiver. It also reveals message M to the receiver, and proves via Π_2 that $\{c, \mathsf{com}, M\} \in L$, where c denotes $(\mathsf{com}_{1,C}(M; r), \mathsf{com}_{1,R}(\cdot), \mathsf{com}_{2,C}(M; r))$.
- The receiver accepts if Π verifies.

Figure 10: One-Round Non-Malleable Commitments with respect to Opening

Non-malleability with respect to opening of the scheme in Figure 10, can be proven via the following sequence of hybrid experiments.

Hybrid_0 : This hybrid denotes the joint distribution of the view of the MIM (in the commitment and opening phases) together with the value committed (corresponding to statistical binding mode) during the commitment phase, when the MIM interacts with an honest committer committing to some value value. This corresponds to the distribution $\mathsf{MIM}_{\langle C,R\rangle,\mathsf{open}}(\mathsf{value}, z)$.

Hybrid_1 : This hybrid is the same as Hybrid_0, except that the SPSS ZK argument Π is simulated (in time T_{Sim}). Since $T_{\mathsf{com}} << T_{\mathsf{zk}}$, the joint distribution of the view and value committed by the MIM remains indistinguishable – since the value committed can be extracted via brute force in time T_{com}, and if the joint distribution of view and value becomes distinguishable, this can be used to violate the zero-knowledge property of the commitment scheme against T_{zk}-time adversaries.

Hybrid_2 : This hybrid is the same as Hybrid_1, except that the commitment com is generated as a commitment to 0. On the other hand, the opening is still to message M, and the SPSS ZK proof is still simulated, the same way as Hybrid_1. Again, the challenger runs in time T_{Sim}. By hiding of the commitment scheme com against T_{Sim}-time adversaries, the fraction of executions where the MIM fails to provide a valid opening, as well as the joint distribution of the view and value committed (defined to be $\perp$) for these executions remains indistinguishable between both hybrids. Moreover, by non-malleability of the commitment scheme com against T_{Sim}-time adversaries, conditioned on the MIM completing a valid opening (in particular, this means the MIM sent a valid second message for com) the joint distribution of the view (including the opening) and the value committed by the MIM remains indistinguishable between Hybrid_1 and Hybrid_2.

Suppose there exists a distinguisher $\mathcal{D}$ that distinguishes the joint distribution of the view and value committed between Hybrid_1 and Hybrid_2 (conditioned on executions where the MIM did not abort), then there exists a reduction $\mathcal{R}$ against the non-malleability of com. The reduction does the following: It externally obtains com as either a commitment to 0 and 1, and sends this as the honest commitment. If the MIM does not complete the commitment phase, $\mathcal{R}$ outputs 0. Else, if the MIM completes the commitment phase, the reduction $\mathcal{R}$ obtains the value committed by the MIM, and then runs the distinguisher $\mathcal{D}$ on the joint distribution of the view and value. Then, we have that $|\Pr[\mathcal{R} = 1|\mathsf{com}\text{ was a commitment to }0] - \Pr[\mathcal{R} = 1|\mathsf{com}\text{ was a commitment to }M]| \geq \frac{1}{\mathsf{poly}(n)}$, which is a contradiction. Thus, we have that the joint distribution of the view and committed value remains indistinguishable between Hybrid_2 and Hybrid_3.

This hybrid corresponds to the simulated distribution, $\mathsf{Sim}_{\langle C,R\rangle,\mathsf{open}}(1^n, z)$, completing our proof of non-malleability. □

Combining Theorem 8 with ($\ell(n)$-concurrent) two-round non-malleable commitment with simultaneous messages described in the previous section, we obtain our main result, that is, ($\ell(n)$-concurrent) non-malleable commitments with respect to opening, according to Definition 8, against all PPT man-in-the-middle adversaries. Combining Theorem 8 with fully concurrent two-round non-malleable commitments with simultaneous messages described in Appendix A, we obtain fully concurrent non-malleable commitments with respect to opening, according to Definition 9, against all PPT man-in-the-middle adversaries.

7.2.1 Non-Interactive Non-Malleable Commitments in a Special Setting

Our protocol implies non-interactive non-malleable commitments with respect to opening, in a setting where parties are determined a-priori and have access to a broadcast channel. Moreover, each party is aware of every other party in the system.

In this setting, the protocol of Figure 10 can be compressed further, so that party P_i while sending its own *commit* message (say C_i), is also required to send *receive* messages $R_{i,j}$ corresponding to all other parties in the system (note that the receiver message is a random string and therefore does not require knowledge of the tags of other parties) in the same non-interactive message.

While opening, commit message C_j, party P_j is required to provide openings with respect to all receive messages $\{R_{i,j}\}_{i\in[\mathcal{N}]}$, that other parties have sent so far, corresponding to party P_j. Here $\mathcal{N}$ denotes an upper bound on the number of parties in the system. If a party chooses not to commit before another party opens, we no longer need to guarantee non-malleability with respect to the opened commitment.

7.3 Two Round Multi-party Coin-Tossing

In this section, we describe a two round multi-party coin-tossing scheme that is *simulatable* via super-polynomial time simulation, according to Definition 13. In particular, this also implies two-round multi-party pseudo-random coin tossing according to Definition 14.

The scheme (for $\mathcal{N}$ parties) is described in Figure 11, and consists of each party P_i sampling random coins r_{ij} and sending to every other party P_j, a one-round $\mathcal{N}^2$-bounded concurrent non-malleable commitment to r_{ij}. In the second round, all parties open their commitments, and output $\bigoplus_{i,j\in[\mathcal{N}]^2} r_{ij}$.

Theorem 9. *Assuming ($\ell(n)$-)concurrent non-malleability of the underlying non-malleable commitment scheme* $\mathsf{NM}-\mathsf{Com}$*, the protocol in Figure 11 is a two-round multi-party coin tossing protocol with a super-polynomial time simulator, according to Definition 14, secure against $\ell(n)$ corruptions.*

Proof. We first describe the simulator $\mathcal{S}$ that forces an external (uniform random) output. $\mathcal{S}$ picks an honest party in the set and simulates it (while honestly playing on behalf of all other parties), Let $\mathsf{NM}-\mathsf{Com} = \{\mathsf{NM}-\mathsf{Com}_{1,C}, \mathsf{NM}-\mathsf{Com}_{1,R}, \mathsf{NM}-\mathsf{Com}_{\mathsf{open}}\}$ denote a one-round non-malleable commitment scheme, where $\mathsf{NM}-\mathsf{Com}_{1,C}$ and $\mathsf{NM}-\mathsf{Com}_{1,R}$ denote the simultaneous committer and receiver messages during the commitment round, and $\mathsf{NM}-\mathsf{Com}_{\mathsf{open}}$ denotes the opening message.

- **Round I:**

  1. Each party P_i samples randomness $r_{ij}, \widetilde{r}_{ij} \xleftarrow{\$} \{0,1\}^*$, and sends $c_{ij} = \mathsf{NM}-\mathsf{Com}_{1,C}(r_{ij}; \widetilde{r}_{ij})$ to party P_j for all $j \in [\mathcal{N}] \setminus \{i\}$.
  2. Each party P_i also samples randomness $\hat{r}_{ij}$ and sends $\mathsf{NM}-\mathsf{Com}_{1,R}(\hat{r}_{ij})$ to party P_j for all $j \in [\mathcal{N}] \setminus \{i\}$.

- **Round II:**

  1. Each party P_i outputs $\mathsf{NM}-\mathsf{Com}_{\mathsf{open}}(r_{ij})$ for all $j \in [\mathcal{N}]$.

- **Output:** At the end of this round, parties output $\bigoplus_{i,j\in[\mathcal{N}]^2} r_{ij}$.

Figure 11: Two Round Multi-Party Coin Tossing

whereas the adversary may be corrupting upto $n-1$ parties. For simplicity, we describe the simulation in the setting where there is a single honest party, this directly extends to simulating any general number of honest parties, by simulating one party and using honest strategy on behalf of all other parties.

$\mathcal{S}$ runs in time $T_{\mathsf{Sim,nmc}} \cdot \mathsf{poly}(1/\delta(n))$ (where $\delta(n)$ denotes the simulation error, which can be set to any neligible value, and $T_{\mathsf{Sim,nmc}}$ denotes the running time of the simulator for the non-malleable commitments). It runs the simulation strategy $\mathcal{S}_{\mathsf{nmc}}$ for the non-malleable commitment protocol, sending to $\mathcal{S}_{\mathsf{nmc}}$, a random string $r_1 \leftarrow \{0,1\}^n$ in the opening phase. If the adversarial parties open, $\mathcal{S}$ records the opened value. If not, $\mathcal{S}$ runs the opening phase of $\mathcal{S}_{\mathsf{nmc}}$ again with a different uniformly random chosen string $r_2 \leftarrow \{0,1\}^*$. It repeats $\mathsf{poly}(1/\delta(n))$ times trying with independent uniform random r_i for $i \in [\mathsf{poly}(1/\delta(n)]$. If the adversarial parties abort in all executions, $\mathcal{S}$ outputs $\perp$. (By a simple probabilistic argument, together with the non-malleability w.r.t. opening of the commitment scheme, this also implies that with overwhelming probability, the adversary aborts in the real execution over the randomness of the coins of honest parties.)

Else $\mathcal{S}$ obtains the value v opened by the adversarial set of parties, which equals the value committed by the adversarial set of parties during the simulated experiment (this is because the MIM's commitment remains computationally binding even during the simulated experiment, refer Definition 8). On obtaining this value, $\mathcal{S}$ obtains external coins ext, computes $r = \mathsf{ext} \oplus v$, and repeats the opening phase, now sending r to $\mathcal{S}_{\mathsf{nmc}}$ for the honest opening. $\mathcal{S}$ then outputs the resulting transcript of the execution with $\mathcal{S}_{\mathsf{nmc}}$ opening to r.

By the computational binding property of the non-malleable (with respect to opening) scheme in the simulated experiment, the adversary's output is either $\perp$ or v, in which case the simulator has successfully forced the output to $\mathsf{ext} = r \oplus v$.

Because ext is chosen uniformly at random, the view of the adversary remains indistinguishable between the real and simulated worlds. Thus, non-malleability of com guarantees that the joint distribution of the view of the MIM and value committed, is indistinguishable, from the joint distribution of the view and value committed in a real execution where the honest parties commit and open coins chosen uniformly at random. This completes the proof of simulatable coin tossing with a super-polynomial simulator. □

Remark 7. *In spite of requiring super-polynomial time simulation, we observe that our two-round coin-tossing protocol can be a useful component of protocols achieving standard polynomial time simulation. For instance, we observe that our two-round coin-tossing protocol can be used to generate a CRS for the two-round semi-malicious MPC protocol of Mukherjee and Wichs [MW16], for which the semi-malicious simulation strategy does not itself require programmability of the CRS.*

We claim that the resulting four-round protocol is secure against adversaries behaving maliciously in the first two rounds, and semi-maliciously in the last two rounds, with only polynomial simulation. The simulator for the protocol honestly generates the common random string in the first two rounds, and then runs the semi-malicious simulator of [MW16] for the last two rounds.

To argue indistinguishability, consider a series of hybrid experiments, where in the first experiment, the simulator forces the output of the coin tossing to an external CRS. Next, it switches from behaving honestly in the last two rounds, to using the semi-malicious simulation strategy (while still forcing the output of the coin toss, allowing the proof of semi-malicious security to go through). Finally, it switches back the output of the coin toss to being generated honestly (while still using the semi-malicious simulation strategy which can work with any external non-programmable CRS). Assuming appropriate sub-exponential hardness of the two-round semi-malicious protocol of [MW16], this yields a four-round hybrid protocol with polynomial time simulation.

More generally, to compile from the resulting hybrid protocol to full malicious security, it should suffice to apply techniques similar to those in [GMPP16], especially in the sub-exponential hardness regime (that is, using non-malleable commitments, strong delayed-input witness indistinguishable arguments of knowledge and four round delayed-input zero-knowledge arguments for input extraction and enforcing correct output).

References

[Bar02] Boaz Barak. Constant-Round Coin-Tossing with a Man in the Middle or Realizing the Shared Random String Model. In *FOCS 2002*, pages 345–355, 2002. 2

[BGI$^+$17] Saikrishna Badrinarayanan, Sanjam Garg, Yuval Ishai, Amit Sahai, and Akshay Wadia. Two-message witness indistinguishability and secure computation in the plain model. 2017. 18

[Blu81] Manuel Blum. Coin flipping by telephone. In *Advances in Cryptology: A Report on CRYPTO 81, CRYPTO 81, IEEE Workshop on Communications Security, Santa Barbara, California, USA, August 24-26, 1981.*, pages 11–15, 1981. 4

[CIO98] Giovanni Di Crescenzo, Yuval Ishai, and Rafail Ostrovsky. Non-interactive and non-malleable commitment. In Jeffrey Scott Vitter, editor, *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*, pages 141–150. ACM, 1998. 3

[COSV16a] Michele Ciampi, Rafail Ostrovsky, Luisa Siniscalchi, and Ivan Visconti. 4-round concurrent non-malleable commitments from one-way functions. ePrint Report 2016/621, 2016. 2

[COSV16b] Michele Ciampi, Rafail Ostrovsky, Luisa Siniscalchi, and Ivan Visconti. Concurrent non-malleable commitments (and more) in 3 rounds. ePrint Report 2016/566, 2016. 2

[DDN91] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-Malleable Cryptography (Extended Abstract). In *STOC 1991*, 1991. 1, 2, 9, 13, 18

[GK90] Oded Goldreich and Hugo Krawczyk. On the composition of zero-knowledge proof systems. *SIAM Journal on Computing*, 25:169–192, 1990. 5

[GKS16] Vipul Goyal, Dakshita Khurana, and Amit Sahai. Breaking the three round barrier for non-malleable commitments. In *FOCS*, 2016. 2, 3, 4, 15, 21

[GLOV12] Vipul Goyal, Chen-Kuei Lee, Rafail Ostrovsky, and Ivan Visconti. Constructing non-malleable commitments: A black-box approach. In *FOCS*, 2012. 2

[GMPP16] Sanjam Garg, Pratyay Mukherjee, Omkant Pandey, and Antigoni Polychroniadou. The exact round complexity of secure computation. In *EUROCRYPT 2016*, pages 448–476, 2016. 4, 50

[Goy11] Vipul Goyal. Constant Round Non-malleable Protocols Using One-way Functions. In *STOC 2011*, pages 695–704. ACM, 2011. 2, 18

[GPR15] Vipul Goyal, Omkant Pandey, and Silas Richelson. Textbook non-malleable commitments. *IACR Cryptology ePrint Archive*, 2015:1178, 2015. 2

[GRRV14] Vipul Goyal, Silas Richelson, Alon Rosen, and Margarita Vald. An algebraic approach to non-malleability. In *FOCS 2014*, pages 41–50, 2014. 2

[HK12] Shai Halevi and Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. *J. Cryptology*, 25(1):158–193, 2012. 18

[JKKR17] Abhishek Jain, Yael Kalai, Dakshita Khurana, and Ron Rothblum. Distinguisher-dependent simulation in two rounds and its applications. 2017. http://eprint.iacr.org/2017/330. 8, 14, 15, 18, 29

[KOS03] Jonathan Katz, Rafail Ostrovsky, and Adam Smith. Round Efficiency of Multi-party Computation with a Dishonest Majority. In *Advances in Cryptology — EUROCRYPT '03*, volume 2656 of *Lecture Notes in Computer Science*, pages 578–595. Springer, 2003. 23

[KS17] Dakshita Khurana and Amit Sahai. Birthday simulation from exponential hardness: 2 round non-malleable commitments and 3 round gap zk. 2017. 5, 6, 7, 9, 13

[LP] Huijia Lin and Rafael Pass. Constant-round Non-malleable Commitments from Any One-way Function. In *STOC 2011*, pages 705–714. 2, 13

[LPS17] Huijia Lin, Rafael Pass, and Pratik Soni. Two-round and non-interactive concurrent non-malleable commitments from time-lock puzzles. Cryptology ePrint Archive, Report 2017/273, 2017. http://eprint.iacr.org/2017/273. 2, 5, 6

[LPV] Huijia Lin, Rafael Pass, and Muthuramakrishnan Venkitasubramaniam. Concurrent Non-malleable Commitments from Any One-Way Function. In *TCC 2008*, pages 571–588. 12, 18, 21

[LPV09] Huijia Lin, Rafael Pass, and Muthuramakrishnan Venkitasubramaniam. A Unified Framework for Concurrent Security: Universal Composability from Stand-alone Non-malleability. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, STOC '09, pages 179–188, 2009. 16, 53

[MW16] Pratyay Mukherjee and Daniel Wichs. Two round multiparty computation via multi-key FHE. In *EUROCRYPT 2016*, pages 735–763, 2016. 50

[Nao03] Moni Naor. On cryptographic assumptions and challenges. In *CRYPTO 2003*, pages 96–109, 2003. 5

[NP01] Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In *Proceedings of the Twelfth Annual Symposium on Discrete Algorithms, January 7-9, 2001, Washington, DC, USA.*, pages 448–457, 2001. 18

[OPV09] Rafail Ostrovsky, Giuseppe Persiano, and Ivan Visconti. *Simulation-Based Concurrent Non-malleable Commitments and Decommitments*, pages 91–108. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. 21

[Pas03] Rafael Pass. Simulation in quasi-polynomial time, and its application to protocol composition. In *EUROCRYPT 2003*, pages 160–176, 2003. 14

[Pas04] Rafael Pass. Bounded-Concurrent Secure Multi-Party Computation with a Dishonest Majority. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, STOC '04, pages 232–241, 2004. 17

[Pas13] Rafael Pass. Unprovable security of perfect NIZK and non-interactive non-malleable commitments. In *TCC*, pages 334–354, 2013. 1, 3, 5, 9

[Pas16] Rafael Pass. Unprovable security of perfect NIZK and non-interactive non-malleable commitments. *Computational Complexity*, 25(3):607–666, 2016. 2, 6

[PPV08] Omkant Pandey, Rafael Pass, and Vinod Vaikuntanathan. Adaptive One-Way Functions and Applications. In *Advances in Cryptology — CRYPTO '08*, pages 57–74, 2008. 5

[PR05a] Rafael Pass and Alon Rosen. Concurrent Non-Malleable Commitments. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of ComputerScience*, FOCS '05, pages 563–572, 2005. 21

[PR05b] Rafael Pass and Alon Rosen. New and improved constructions of non-malleable cryptographic protocols. In *STOC 2005*, pages 533–542, 2005. 2, 12, 18, 21, 54

[PW] Rafael Pass and Hoeteck Wee. Black-Box Constructions of Two-Party Protocols from One-Way Functions. In *TCC 2009*. 9, 10, 14, 40

[PW10] Rafael Pass and Hoeteck Wee. Constant-round non-malleable commitments from subexponential one-way functions. In *EUROCRYPT 2010*, pages 638–655, 2010. 2

[RSW96] Ronald L Rivest, Adi Shamir, and David A Wagner. Time-lock puzzles and timed-release crypto. 1996. 5

[Wee10] Hoeteck Wee. Black-box, round-efficient secure computation via non-malleability amplification. In *FOCS 2010*, pages 531–540, 2010. 2, 13, 53

A Two Round Fully-Concurrent Non-Malleable Commitments with Simultaneous Messages

In this section, we construct two round fully-concurrent non-malleable commitments with respect to commitment, with simultaneous messages, against synchronous adversaries. Our construction follows via non-malleability amplification techniques developed in previous work, including [LPV09, Wee10]. We first construct a simulation-sound variant of our SPSS ZK protocol, and then use this to obtain concurrent non-malleable commitments in the simultaneous exchange model.

A.1 Simulation-Sound SPSS ZK

In this section, we construct SPSS ZK that satisfies a variant of simulation soundness. We consider a MIM that interacts with an honest prover in the left execution and generates its own argument in the right execution for some possibly related instance: and we require that there exist a (super-polynomial time) simulator-extractor that extracts the witness being used by the MIM to generate the MIM's arguments, without knowing the witness for the honest interaction.

The construction of simulation-sound SPSS ZK is described in Figure 12. This is obtained by substituting the extractable commitments in SPSS ZK (Section 5.2) with non-malleable commitments. We assume the existence of a two-round non-malleable commitment scheme, that is at least $(T_{\mathsf{hid}}, \delta_{\mathsf{hid}})$ hiding (that is, hiding such that T_{hid}-time adversaries have advantage at most δ_{hid} in the hiding game), and at most $(T_{\mathsf{Ext}}, T'_{\mathsf{Ext}}, \delta_{\mathsf{Ext}})$-extractable, where $T_{\mathsf{Ext}} \ll T_{\mathsf{hid}} \ll T'_{\mathsf{Ext}}$, and $\delta_{\mathsf{Ext}} = \mathsf{negl}(n)$. This scheme is exactly the one-one version of the scheme in Section 6, except that in the basic scheme for 4 tags, we use extractable commitments for all tags (instead of using a non-interactive commitment for small tag $= 4$). We note that this scheme is non-malleable against adversaries running in time $\tilde{T} \gg T_{\mathsf{Ext}}$.

We also assume the existence of a two-round witness-indistinguishable proof, denoted by zap such that adversaries running in time T_{wi} have advantage at most δ_{wi}. We assume that com is hard to invert by adversaries running in time $T_{\mathsf{hid-com}}$, and can be broken (via brute-force) in time T_{com}, where $T_{\mathsf{hid-com}} \ll T_{\mathsf{com}}$. We will set parameters so that: $T_{\mathsf{Ext}} \ll T_{\mathsf{hid-com}}, T_{\mathsf{com}} \ll \tilde{T}$.

Looking ahead, the soundness parameter of the resulting simulation-sound SPSS ZK will be such that $T_{\Pi} \cdot T_{\mathsf{Ext}} \ll T_{\mathsf{hid-com}}$, the zero-knowledge parameters will be such that: $T_{\mathsf{zk}} \ll T_{\mathsf{wi}}$, $\delta_{\mathsf{hid}} \geq \delta_{\mathsf{zk}}, \delta_{\mathsf{zap}} \geq \delta_{\mathsf{zk}}$, and the witness extraction parameter will be equal to T_{Ext}.

Lemma 7. *The protocol in Figure 12 is a one-one simulation-sound SPSS zero knowledge argument.*

Proof Sketch. One-one simulation soundness follows from the one-one non-malleability of the $\mathsf{NM-Com}$ against T_{com}-time adversaries, such that when the simulator, in time T_{Sim}, breaks com and changes the commitments e_2 and e'_2 for the honest execution, the distribution of the MIM's view and corresponding committed value (witness) in e'_2 doesn't change.

A.2 Concurrent Two-Round Non-Malleable Commitments w.r.t. Commitment

We will now use one-one simulation-sound ZK to construct two-message non-malleable commitments with respect to commitment, with simultaneous messages. Let $\Pi_{1,\mathsf{tag}}, \Pi_{2,\mathsf{tag}}$ denote both messages of the protocol, which is simulatable in time T_{Sim}, zero-knowledge against adversaries running in time T_{zk}, sound against adversaries running in time T_{Π} and a witness can be extracted from SPSS ZK *by brute-force* in time T_{wext}, where $T_{\Pi} \ll T_{\mathsf{Sim}} \ll T_{\mathsf{zk}} \ll T_{\mathsf{wext}}$. Furthermore, we require that when a simulator in time T_{Sim} simulates the argument in the honest interaction, the MIM continues using the right witnesses to generate his SPSS ZK.

Let $\mathsf{NM}-\mathsf{Com} = (\mathsf{NM}-\mathsf{Com}_1, \mathsf{NM}-\mathsf{Com}_2)$ denote the messages of a two-round non-malleable commitment scheme, and com denote the non-interactive commitment scheme.
Prover Input: Instance x, witness w such that $R(x, w) = 1$, tag tag.
Verifier Input: Instance x.

1. Verifier V sends $e_1 = \mathsf{NM}-\mathsf{Com}_{1,\mathsf{tag}}, \mathsf{NM}-\mathsf{Com}'_{1,\mathsf{tag}}$ to V, together with $c = \mathsf{com}(s; r)$ for $s \xleftarrow{\$} \{0,1\}^n, r \xleftarrow{\$} \{0,1\}^*$ and zap_1.

2. Verifier V picks $s' \xleftarrow{\$} \{0,1\}^n, (r', \widetilde{s}) \xleftarrow{\$} \{0,1\}^*$, computes $e_2 = \mathsf{NM}-\mathsf{Com}_{2,\mathsf{tag}}(s'; r'), e'_2 = \mathsf{NM}-\mathsf{Com}'_{2,\mathsf{tag}}(w; r'')$. It also computes zap_2 proving:
$$\exists w, r'' \text{ such that } w \text{ is a witness for } x \in L \ \wedge\ e'_2 = \mathsf{NM}-\mathsf{Com}'_{2,\mathsf{tag}}(w; r'')\mathsf{OR}$$
$$\exists (s', r', r) \text{ such that } e_2 = \mathsf{NM}-\mathsf{Com}_{2,\mathsf{tag}}(s'; r') \ \wedge\ c = \mathsf{com}(s'; r).$$
It then sends (e_2, zap_2) to P.

3. **Verification.** The verifier accepts (outputs 1) if and only if zap verifies.

Figure 12: Simulation-Sound SPSS ZK

We let com denote a non-interactive statistically binding commitment scheme that is hiding against adversaries running in time T_{hid}, where T_{hid} is larger than brute-force extraction time T_{wext} of the simulation sound SPSS ZK. Then, our construction of concurrent two-round non-malleable commitments w.r.t. commitment is described in Figure 13.

Lemma 8. *The protocol in Figure 13 is a fully concurrent non-malleable commitment with simultaneous messages.*

Proof Sketch. As before, we will only consider one-many non-malleability (for an unbounded number of MIM sessions), and full concurrent non-malleability will directly follows [PR05b]. Furthermore, the synchronous scheduling (with a rushing adversary), and will consider the following sequence of hybrid experiments. Hybrid_0 corresponds to the real execution where the honest committer commits to message M. We let $\mathsf{MIM}_{\langle C,R \rangle}(\mathsf{value}, z)$ denote the joint distribution of the view and all the values committed by the MIM in all right executions.

In Hybrid_1, the challenger continues to commit to message M, but in the second round, starts simulating the SPSS ZK proof for the honest execution. Since the commitment phase already occured, conditioned on the fixed commit phase (and therefore the fixed committed values), the joint distribution of the MIM's views remains indistinguishable. Thus, in this hybrid, the joint distribution of the values committed by the MIM and the views in all right executions remains indistinguishable from Hybrid_0.

Furthermore, by the one-one simulation soundness property of SPSS ZK, even when the simulator runs in time T_{Sim} to simulate the honest proof, the MIM continues to use the actual witness (for the commitment) in each of his executions.

In Hybrid_2, the challenger changes the commitment to message M to a commitment to 0, while still simulating the SPSS ZK proof. Next, the challenger runs in time $T_{\mathsf{wext}} \ll T_{\mathsf{hid}}$ to extract the witness from the MIM's SPSS ZK arguments, and use this witness to extract the values committed by the MIM (in polynomial time). If the MIM stops using correct witnesses or if the joint distribution of witnesses changes, this breaks hiding of the commitment scheme. Thus, we have that

Language L: We define $L = \{\{c, \mathsf{com}\} : \exists M, r : c = \mathsf{com}(M; r)\}$.
Committer Input: Message $M \in \{0,1\}^p$, tag $\mathsf{tag} \in [2^n]$.
Receiver Input: Tag tag.

Commit Stage:

1. **First Round.**
   Committer Message. Sample randomness $R \xleftarrow{\$} \{0,1\}^*$ and send $\mathsf{com}(M; R)$ to the receiver.

   Receiver Message. Send $\Pi_{1,\mathsf{tag}}$ as the first message of Π for language L and tag tag.

2. **Second Round: Committer Message.** Send $\Pi_{2,\mathsf{tag}}$ proving that:

$$\{c, \mathsf{com}\} \in L$$

3. The receiver accepts the commitment if Π verifies.

Reveal Stage: The committer reveals randomness R to the receiver. The receiver verifies that the commitment was correctly decommitted.

Figure 13: Two Round Fully Concurrent Non-Malleable Commitments with Simultaneous Messages

the joint distribution of the view and values committed by the MIM in all right sessions, remains indistinguishable between Hybrid_1 and Hybrid_2, otherwise this would contradict the hiding of com.

Thus, the joint distribution of the view and values committed by the MIM in all his right interactions, remains indistinguishable between the real and simulated executions. The simulator can also generate the SPSS ZK arguments honestly, and indistinguishability of the joint distribution follows via the same argument as indistinguishability between Hybrid_0 and Hybrid_1. We note that all these arguments go through against a rushing adversary that obtains all honest messages for some round before generating his own message for the same round. This completes the sketch of the proof.